

Patent No. 11,558,317
Expert Declaration in Support of *Inter Partes* Review

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

WHATFIX INC.,
Petitioner,

v.

WALKME LTD.,
Patent Owner.

Patent No. 11,558,317

Inter Partes Review No. IPR2024-00065

DECLARATION OF DR. HOMAYOON BEIGI

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

Exhibit List

Exhibit Description	Exhibit #
U.S. Patent No. 11,558,317	1001
File History of U.S. Patent No. 11,558,317	1002
Expert Declaration of Dr. Homayoon Beigi	1003
S. Symons, “When Chatbots meet RPA bots,” https://medium.com/roborana/when-chatbots-meet-rpa-bots-c8b22579f49a , Medium (Oct. 19, 2017)	1004
Screenshots of S. Symons, “RPA meets chatbot: Automatically generate e-mail signature,” https://vimeo.com/238927879 , Vimeo (Oct. 19, 2017)	1005
English Translation of Screenshots of S. Symons, “RPA meets chatbot: Automatically generate e-mail signature,” Vimeo (Oct. 19, 2017)	1006
Declaration of Mathijs Mekes	1007
Declaration of Senne Symons	1008
W. Baetens, “When Chatbots meet RPA bots,” https://chatbotslife.com/when-chatbots-meet-rpa-bots-c5f71e462e6 , Medium (Oct. 20, 2017)	1009
Declaration of Wouter Baetens	1010
D. Williams et al., “Using artificial intelligence to optimize the value of robotic process automation,” https://www.ibm.com/downloads/cas/KDKAAK29 , IBM White paper (2017)	1011
Declaration of Ira Allen	1012

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

Exhibit Description	Exhibit #
IT World Canada, “Using artificial intelligence to optimize the value of robotic process automation,” https://www.itworldcanada.com/assets/using-artificial-intelligence-to-optimize-the-value-of-robotic-process-automation , IT World Canada (2017)	1013
U.S. Patent No. 7,983,997 to Allen <i>et al.</i>	1014
[RESERVED]	1015
U.S. Patent No. 10,839,798 to Krishnan <i>et al.</i>	1016
U.S. App. No. 62/581,550	1017
U.S. Patent No. 10,503,627 to Radhakrishnan <i>et al.</i>	1018
[RESERVED]	1019
[RESERVED]	1020
[RESERVED]	1021
[RESERVED]	1022
[RESERVED]	1023
[RESERVED]	1024
[RESERVED]	1025
C. Meléndez, “Chatbots and RPA: Digital Heaven,” https://www.wovenware.com/blog/2016/10/chatbots-rpa-digital-heaven/ , Wovenware (Oct. 14, 2016)	1026
Declaration of Nathaniel Frank-White	1027
P. Ferscht <i>et al.</i> , “Robotic Automation Emerges as a Threat to	1028

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

Exhibit Description	Exhibit #
Traditional Low-cost Outsourcing,” https://www.horsesforsources.com/wp-content/uploads/2016/06/RS-1210_Robotic-automation-emerges-as-a-threat-060516.pdf , Horses for Sources (2012)	
M. Lacity <i>et al.</i> , “What Knowledge Workers Stand to Gain from Automation,” https://hbr.org/2015/06/what-knowledge-workers-stand-to-gain-from-automation , Harvard Business Review (June 19, 2015)	1029
P. Bornet, “Intelligent automation is about creating synergies between RPA, cognitive, chatbots and AI,” https://www.linkedin.com/pulse/from-robotic-process-automation-rpa-artificial-ai-journey-bornet , LinkedIn (Apr. 22, 2017)	1030
C. Le Clair, “The Forrester Wave™: Robotic Process Automation, Q1 2017,” Forrester Wave (Feb. 13, 2017)	1031
M. Arya, “A brief history of Chatbots,” https://chatbotlife.com/a-brief-history-of-chatbots-d5a8689cf52f , Medium (Mar. 11, 2019)	1032
S. Symons, “RPA meets chatbot: Automatically generate e-mail signature,” https://vimeo.com/238927879 , Vimeo (Oct. 19, 2017)	1033
[RESERVED]	1034
[RESERVED]	1035

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

I, Homayoon Beigi, make this declaration in connection with the proceeding identified above.

I. INTRODUCTION

1. I have been retained as an expert witness on behalf of Whatfix Inc. (“Whatfix” or “Petitioner”) to provide expert opinions on the patentability of United States Patent No. 11,558,317 (“the ’317 patent”).

2. I am being compensated at my normal rate, plus reimbursement for expenses, for my analysis. My compensation does not depend on the content of my opinions or the outcome of this proceeding.

II. EXPERIENCE AND QUALIFICATIONS

3. In formulating my opinions, I have relied upon my knowledge, training, and experience in the relevant art. My qualifications are stated more fully in my curriculum vitae, which has been provided as Appendix A. Here, I provide a brief summary of my qualifications.

4. As set forth in my resume, I have over 33 years of experience in the field of interactive computing devices, initially with International Business Machines (IBM) during development of its earliest handwriting, speaker, and speech recognition engines, and later with Recognition Technologies to develop proprietary and advanced recognition engines for performing speaker, speech, and

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

face recognition. As detailed below, I have worked on many projects and technologies relevant to the subject matter of the '317 patent.

5. My academic background provides a technical foundation for my work in speech, biometric, and pattern recognition research, as well as my work in the design and implementation of various transcription, access, and biometric systems. I earned a Bachelor of Science Degree in Mechanical Engineering from Columbia University in 1984. I earned a Master of Science in Mechanical Engineering from Columbia University in 1985, with a focus on Kinematics, Dynamics, and Control Systems. I earned my Doctor of Engineering Science in Mechanical Engineering from Columbia University in 1991. My thesis was titled “Neural Network Learning and Learning Control Through Optimization Techniques.”

6. While at Columbia, I was a research assistant for multiple different professors. In Columbia’s Center for Telecommunication Research, I assisted in the research of digital image coding and image data compression and transmission. In the Department of Mechanical Engineering, I assisted with researching several different scientific areas, including learning-adaptive control and neural network learning, fault detection of mechanical systems and machine health prognosis,

generalization theory on kinematic analysis of commonly used linkages, and digital image processing applied to fluid mechanics for the analysis of lubricant behavior in zero gravity.

7. I have 28 years of experience, teaching several graduate-level courses as an adjunct Professor in the Computer Science, Mechanical Engineering, and Electrical Engineering departments at Columbia University, including courses on speech recognition, speaker recognition, handwriting recognition, digital control systems, applied signal recognition, and mathematics of machine learning. In the spring of 2023, I was chosen as one of 11 finalists for the Columbia University Presidential Awards in Teaching Excellence, across the faculty body of over 3000 throughout the University. I have also been advising several Master's and Ph.D. students and have been a member of many Ph.D. defense committees at Columbia University.

8. From 1991 to 2001, I was a postdoctoral research fellow and later a Research Staff member at IBM T.J. Watson Research Center. While at IBM, I engaged in numerous research activities directly related to handwriting and speech processing. For example, from 1991-1997, I developed online handwriting algorithms using a Hidden Markov Model (HMM) framework. And from

1997-2001, I conducted research in speaker recognition (verification and identification) and speech recognition as part of the development of the IBM Research Speaker Verification and Identification Engine, which is still being used in an array of different products at IBM. In this capacity, I worked directly on speech, speaker, image, and face recognition engines and their fusion for handling full diarization of multimedia segments, including full transcription and searching of content.

9. Beginning in 2000, I became fully involved in Internet Server Connections, Inc., of which I am the Vice President and Chief Technology Officer. In 2003, I began a second company, Recognition Technologies, Inc. I am the Chief Executive Officer and President of Recognition Technologies. During my time with Recognition Technologies, and extending before its inception, I developed the algorithms and code for an interactive voice response system and several recognition engines for pattern recognition, culminating over 5 Millions of lines of C++ code. This collection of engines, called RecoMadeEasy® includes full speech transcription capabilities in several languages, speaker segmentation and recognition, emotion detection, and face detection and recognition. All engines are capable of client/server as well as standalone operation. These engines

provide a websocket interface, a streaming interface, a full C++ API and provide services on servers, standalone computers, embedded Linux and Android devices, as well as a command-line access.

10. I have been awarded 13 patents as an inventor, many of which relate to pattern recognition involving speaker, face, and speech recognition. I also have over 100 publications and 3 best paper awards. In addition, I am the author of the first and only textbook on speaker recognition (1000+ pages of fundamentals), titled “Fundamentals of Speaker Recognition,” published by Springer-Verlag, which is the primary textbook used for my courses on “Fundamentals of Speech Recognition” and “Mathematics of Machine Learning and Signal Recognition.” It is also used by other researchers in the field, with over 80,000 electronic copies sold, plus many hard copies.

11. I was also voted into and served as the biometrics and speaker recognition expert for the “Voice Identification Policy Group,” which was formed by the U.S. Embassy in Mexico to help provide policy for surveillance efforts of the 600,000+ Mexican Federal Police. In that role, my responsibilities primarily centered around speaker biometrics.

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

12. I also served as the keynote speaker for the Mexican government's "Primera Reunión Nacional de Usuarios AVIS," which was a conference for the users of the Federal police voice surveillance system, and was held in November of 2011 in Mexicali, Baja California, Mexico.

13. In 2009, I served as an Interoperability Panelist in the U.S. Government Interagency Symposium for Investigatory Biometrics, and from 2009 to 2010, I was an Interoperability Committee Member for the Federal Bureau of Investigation.

14. Finally, I served as an Active Liaison in the ISO/SC 37 JTC WG3 group formed to create International Biometrics standards for the "Common Biometric Exchange Format (CBEF)." I also served as an active liaison and driving force for the speaker recognition standards development, from 2003 to 2010, in the ANSI/INCITS standards development for Biometric Data Interchange Formats in which I defined an ANSI standard for media interchange. I also had a similar role in the VoiceXML 3.0 standards development effort.

15. Over the course of my academic and professional career, I have received several awards relating to my work in pattern, speech and speaker recognition, which are detailed in my attached Curriculum Vitae (Appendix A).

III. MATERIALS CONSIDERED

16. In preparing this declaration, I have reviewed, among other things, the '317 patent and file history. I have also considered each of the documents cited in this Declaration and the documents specifically identified below and in the Exhibit List. In formulating my opinions, I have further relied upon my extensive experience in the relevant fields.

IV. OVERVIEW OF NATURAL LANGUAGE PROCESSING AND PROCESS AUTOMATION

17. Natural language processing (NLP) refers to the interaction between a computer and a human using human “natural” language. Human “natural” language is ordinary language used to communicate every day among humans. One example application for NLP is a customer service chatbot¹, one that provides customer service on, *e.g.*, websites via text chat. By using NLP, human users are able to converse with the chatbots using every day, natural language (instead of using computer or programming code, which is not a natural form of communicating). Some businesses have replaced human customer service agents

¹Chatbots are as old as artificial intelligence itself, dating back to 1950 when Alan Turing devised his so called “Turing Test” to test the intelligence of such systems. However, the phrase Chatbot seems to have been first used in 1994 and 1995 for Eliza and Alice (two such programs). (EX1032.)

with chatbots. As Melendez explains, “[b]usinesses have jumped on the chatbot bandwagon, using them as agents in the first line of customer-service response when we call for information, or to provide information and answer questions when we visit their websites. These assistants not only help companies gain efficiencies by automating some of the routine communication tasks—but even more importantly—as this technology continues to mature, they can elevate the customer experience by helping customers quickly and accurately find what they need.” (EX1026, 2.)

18. Process automation refers to automation of certain tasks, for example, transactional, repetitive tasks (automated using, *e.g.*, robotic process automation (RPA) bots). Process automation has been known and in use by at least October 2012. One organization “spent some time examining a new technology called robotic automation, pioneered by UK-based vendor Blue Prism, that enables non-engineers to automate certain business processes quickly and cheaply.” (EX1028, 1.) “Knowledge workers consistently tell us they want to be liberated from such highly structured, routine, and dreary tasks to focus on more interesting work. Some are actually getting that wish, thanks to a new approach known as Robotic Process Automation (RPA).” (EX1029, 2.)

19. As Melendez explains, “companies are turning to other types of automation, such as Robotic Process Automation (RPA), or software robots, to automate high-volume, repeatable tasks that were previously handled manually. This technology has taken hold in a variety of industries and has proven most useful for back-office functions. The insurance industry, for example, uses RPA to efficiently and accurately process claims; while security professionals are looking to apply it to the complex processes associated with ID and access management when people leave or change positions. Mortgage and banking professionals use these software robots to conduct credit checks, employment verifications and other tasks so that they can process mortgages more quickly.” (EX1026, 2.)

20. Both NLP and process automation offer several benefits such as lower operational costs, high volume processing, faster turnaround time, and improvements in user experience, quality, and accuracy. (EX1030, 2.) In 2017, a chatbot could have an “intelligent conversation[] with a human,” but may have been limited to providing pure raw information (*e.g.*, the weather for the week, answers from a search engine). (*Id.*, 6.) An RPA bot may be able to automatically perform more complex tasks that require, *e.g.*, analysis or executing actions. However, in 2017, the “limits of RPA lie in the absence of capacity to . . . interact

using natural language and to handle judgement activities.” (*Id.*, 3.) In other words, while a chatbot could interact with a human, it may not have been able to handle complex tasks, and while an RPA bot can handle complex tasks, it may not be good at interacting with humans. The combination could have been in different forms, one being what some called intelligent chatbots. The “intelligent chatbots enable interaction with users (*e.g.*, internal or external customers). Their benefits are mainly qualitative, focusing on customer experience improvement.” (*Id.*, 4.) An RPA could perform more complex tasks that a chatbot may not be able to. For example, an RPA could adapt to “applications on desktops or the web [that] change frequently,” such as the location of user interface (UI) elements (*e.g.*, the “Submit” button) on a displayed UI. (EX1031, 4.) “[T]his harmonious alliance [of a chatbot combined with an RPA bot] is revolutionizing business operations, enhancing customer experiences, and driving efficiency gains.” (EX1026, 1.) “[E]ndless possibilities [] arise when chatbots and RPA come together.” (*Id.*, 1-2.)

21. The idea of combining NLP and process automation was known by at least November 5, 2017 (earliest priority date for the ’317 patent). In April 2017, Bornet commented: “[t]o create a maximum value out of this journey, effective interactions between the generations of robots need to be implemented. For

example, if intelligent chatbots are not connected with RPA, although chatbots create intelligent conversations with humans, they are not able to deliver more than pure raw information (*e.g.*, the price of a stock, or the temperature currently in London). We call this the ‘empty shell effect.’ Let us take a look at a concrete illustration of interaction between intelligent chatbot and RPA. During a conversation, intelligent chatbot understands that a finance manager needs some clarity on the evolution of the margins of one of its business divisions. Hence, chatbot asks the finance manager whether he wants to receive the usual margin analysis report. The finance manager confirms. The chatbot triggers RPA, which produces the report computing information from different systems, and delivers to the chatbot. The chatbot then provides the requested report to the finance manager. . . . This example demonstrates a powerful synergy where chatbots improve the link between the humans and the RPA platform, while RPA enriches the content and the reach of the chatbots’ interactions with humans.” (EX1030, 6-7.) By February 2017, EdgeVerve implemented a “chatbot for a technology company that will launch RPA scripts based on user interactions. Several vendors had integrated a voice or chatbot interface into their RPA platforms” (EX1031, 4.)

V. SUMMARY OF THE '317 PATENT

A. Filing Date and Priority Dates

22. The '317 patent was filed on January 18, 2022, and issued on January 17, 2023. The '317 patent is a continuation of U.S. Patent No. 11,258,732 and U.S. Patent No. 10,819,664, and claims priority to Provisional Application 62/581,714, which was filed on November 5, 2017.

B. Specification

23. The '317 patent is titled “Invoking an Automatic Process in a Web-based Target System Using a Chat-bot.” The '317 patent discloses “[a] method, apparatus and product for chat-based application interface for automation.” (EX1001, Abstract.) An allegedly “user-friendly manner of defining functionality of chat bots” is disclosed, where the “chat bot may collect information to be used as part of the automation process.” (*Id.*, 7:67-8:1, 9:51-54, 10:15-17, 14:21-25.) The automation process “interact[s] with a Graphical User Interface (GUI) of [a] web-based target system and simulat[es] user input to the GUI of the web-based target system[.]” (*Id.*, Claim 1.) The automation process is performed with respect to a third-party web-based target system and without relying on an Application Programming Interface (API) exposed by the web-based target system. (*Id.*)

24. Fig. 2A (below) shows a flowchart involving a chat bot and automation process. “On Step 210, a chatbot is launched.” (*Id.*, 13:32.) “On Step 220, a conversation of the chat bot is selected. . . . On Step 230, the conversation may be implemented.” (*Id.*, 13:41-52.) An automation process is selected and implemented based on the user’s interactions with the chatbot. (*Id.*, 13:66-14:25.) “On Step 240, an automation process may be selected. . . . On Step 250, the automation process may be executed.” (*Id.*, 14:12-25.)

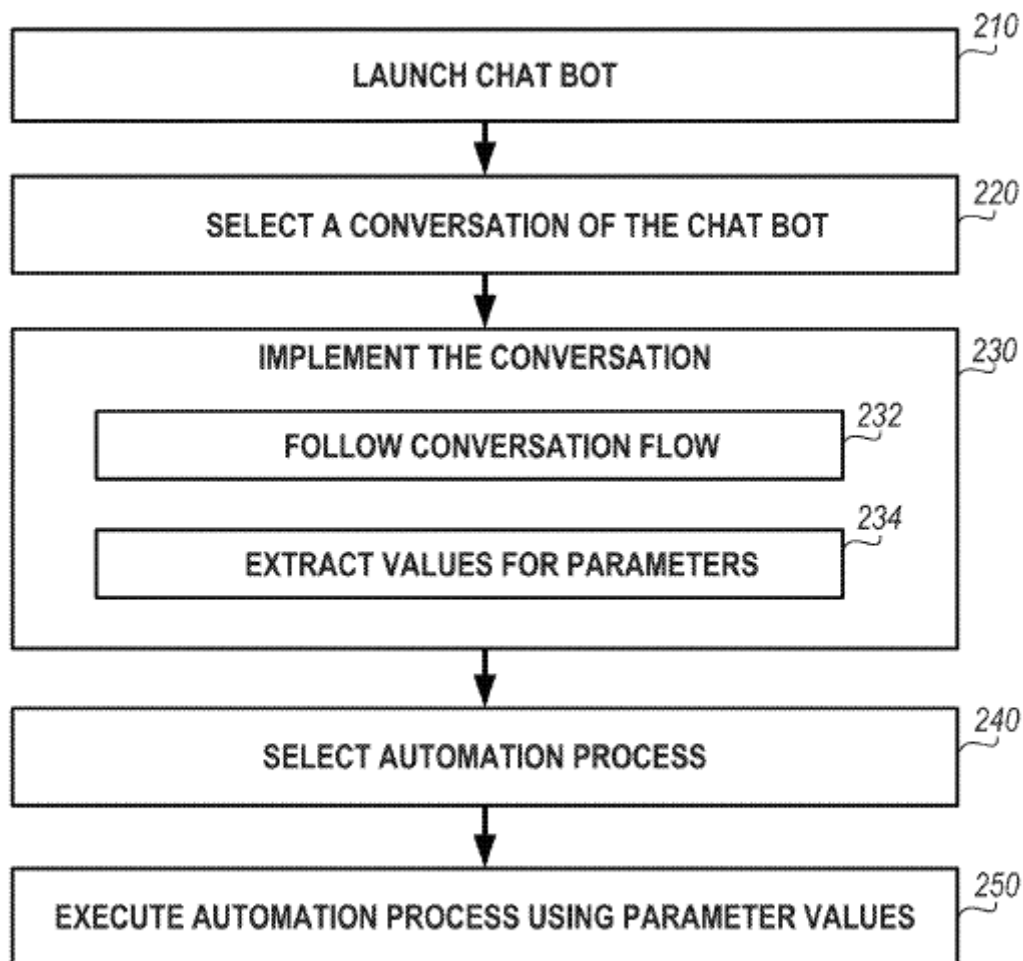


FIG. 2A

C. Claims

25. I was asked to evaluate whether claims 1-20 of the '317 patent are patentable.

26. Claim 1 is directed to chat-bot program product comprising a non-transitory computer readable medium retaining program instructions that

cause the processor to receive a user instruction from a user, determine to implement an automation process on a web-based target system, and collect information used as part of the automation process. The automation process interacts with a graphical user interface (GUI) of the web-based target system. The automation process is performed without relying on an application programming interface (API). The automation process automatically and without user input, fills in a text input field and interacts with a widget in the GUI. Claims 9 and 16 are directed to a method and a computerized apparatus, respectively, that perform the same.

27. Claims 2, 10, and 17 require the automation process by implemented by a client device used by the user, and claims 3 and 11 require the automation process be implemented by a browser.

28. Claims 4, 12, and 18 require the chat-bot program product converse with a user using a vocal interface.

29. Claims 5, 13, and 19 are directed to an online form.

30. Claims 6 and 14 require information of the user, personalizing the automation process. Claims 7 requires utilizing credentials of the user.

31. Claims 8, 15, and 20 are directed to automatically identifying the widget and the text input field in the GUI.

D. Prosecution History

32. I understand that prosecution of the '317 patent began with a single Office Action, on November 10, 2022, that rejected claims 1-20 based on obvious-type double patenting over the parent U.S. Patent No. 10,819,664 (the '664 patent). (EX1002, 111-117.)

33. On November 19, 2022, Patent Owner filed an electronic terminal disclaimer. (*Id.*, 164-166.)

34. Subsequently, on December 9, 2022, the Examiner issued a Notice of Allowance without stating specific reasons why the claims were allowed. (*Id.*, 167-173.)

VI. RELEVANT LEGAL STANDARDS AND DEFINITIONS

A. Anticipation

35. It is my understanding that the claims of a patent are anticipated by a prior art reference if each and every element of the claim is found either explicitly or inherently in a single prior art reference or system. I understand that inherency requires a showing that the missing descriptive matter in the claim is necessarily or implicitly present in the allegedly anticipating reference and that it would have

been so recognized by a POSITA. In addition, I understand that an enabling disclosure is a disclosure that allows a POSITA to make the invention without undue experimentation. Although anticipation typically involves the analysis of a single prior art reference, I understand that additional references may be used to show that the primary reference has enabling disclosure, to explain the meaning of a term used in the primary reference, and/or to show that a characteristic is inherent in the primary reference.

36. I understand that if the reference is a device or system, the public need not have access to the inner workings of a device for it to be considered in public use or used by others.

B. Obviousness

37. I understand that a claim is unpatentable if the differences between the claimed subject matter and the prior art are such that the subject matter as a whole would have been obvious at the time the claimed subject matter was made to a POSITA to which the subject matter pertains. I understand that a patent claim may be obvious to a POSITA in view of the prior art teachings of a single reference or a combination of references.

38. To assess obviousness, I understand that I am to consider the scope and content of the prior art, the differences between the prior art and the claim, the level of ordinary skill in the art, and any secondary considerations to the extent they exist.

39. It is also my understanding that there are principles that may be used as further guidance in obviousness analysis (especially when considering combinations of references), which include considering whether:

- the claimed subject matter is simply a combination of prior art elements according to known methods to yield predictable results;
- the claimed subject matter is a simple substitution of one known element for another to obtain predictable results;
- the claimed subject matter uses known techniques to improve similar devices or methods in the same way;
- the claimed subject matter applies a known technique to a known device or method that is ready for improvement to yield predictable results;

- the claimed subject matter would have been “obvious to try” choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- there is known work in one field of endeavor that may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to a POSITA;
- there existed at the time of conception and reduction to practice a known problem for which there was an obvious solution encompassed by the patent’s claims; and
- there is some teaching, suggestion, or motivation in the prior art that would have led a POSITA to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed subject matter.

40. I understand that there are secondary considerations that may tend to show that a claimed invention would not have been obvious to a POSITA. These include, for example, commercial success, long-felt but unsolved needs, failure of others, and unexpected results. I further understand that there must be a

relationship, or nexus, between any secondary considerations and the novelty of the claimed invention.

C. Prior Art

41. I am informed and understand that prior art includes patents and printed publications that existed before the effective filing date of the claimed invention. The effective filing date is the earliest claimed priority date or the earliest filing date of the application that describes the subject matter. A patent or published patent application is prior art if it was filed before the earliest filing date of the claimed invention. I understand that a patent or published patent application is entitled to have its effective filing date as the filing date of its provisional application if the disclosure of the provisional application provides support for the claim(s) of the patent or published patent application.

42. I understand that a printed publication is prior art if it was publicly available before the effective filing date. I am informed and understand that a printed publication qualifies as publicly available as of the date it was disseminated or otherwise made available such that any member of the public interested in the subject matter could locate it through the exercise of reasonable diligence. I am

informed and understand that videos can constitute printed publications in *inter partes* review proceedings.

VII. LEVEL OF ORDINARY SKILL IN THE ART FOR THE '317 PATENT

43. I am informed and understand that various factors can be considered in determining a person of ordinary skill in the art (POSITA). I am informed and understand that those factors include: (1) the educational level of the inventors; (2) the type of problems encountered in the art; (3) prior art solutions to those problems; (4) the rapidity with which innovations are made; (5) sophistication of the technology; and (6) education level of active workers in the field.

44. I have been informed that Petitioner proposed in the Petition that a POSITA in the field of the '317 patent in 2017 would have bachelor's degree in engineering or computer science and two years of experience working on NLP or automation processes. Alternatively, a POSITA would have five to six years of experience working on NLP or automation processes.

45. I agree with this level of ordinary skill for the '317 patent. Based on my experiences, I understand and know of the capabilities of a person of this skill level as of the time the '317 patent priority date in 2017. I am familiar with how a

POSITA would have understood and used the terminology found in the '317 patent at the time of its filing, and with the state of the art at that time.

VIII. CLAIM CONSTRUCTION

46. I have been informed that claim construction requires consideration of the words of the claims themselves, the specification, the prosecution history, and extrinsic evidence concerning relevant scientific principles, the meaning of technical terms, and the state of the art. I have also been informed that the specification is usually definitive, as it is the single best guide to the meaning of a disputed term.

47. I have been informed that, in the context of this proceeding, Petitioner did not propose any terms for construction. Instead, Petitioner believes each of the terms of the '317 patent carries its ordinary and customary meaning, which is the meaning that the term would have to a POSITA. I agree with Petitioner and apply the ordinary and customary meaning of the terms in my analysis below.

IX. PRIOR ART

48. I understand that each of the references discussed herein qualifies as prior art against the '317 patent:

49. S. Symons, "When Chatbots meet RPA bots," Medium, (2017)
("Symons Article," EX1004) was published on October 19, 2017;

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

50. S. Symons, “RPA meets chatbot: Automatically generates e-mail signature,” Vimeo, (2017) (EX1005, EX1033) was published on October 19, 2017 (an English translation of EX1005 (“Symons Video”) and certification are provided as EX1006 and EX1007, respectively);

51. D. Williams et al., “Using artificial intelligence to optimize the value of robotic process automation,” IBM White paper, (2017) (“Williams,” EX1011) was published in September 2017;

52. U.S. Patent No. 7,983,997 (“Allen,” EX1014) was filed on November 2, 2007, published on May 7, 2009, and issued on July 19, 2011;

53. U.S. Patent No. 10,839,798 (“Krishnan,” EX1016) was filed on January 30, 2018, claims priority to a provisional application filed on November 3, 2017², published May 9, 2019, and issued on November 17, 2020;

54. U.S. Patent No. 10,503,627 (“Radhakrishnan,” EX1018) was filed on October 30, 2017, published on May 2, 2019, and issued on December 10, 2019.

² I reviewed and compared the provisional patent application filed on November 3, 2017 to Krishnan. (EX1017.) I confirm that Krishnan’s provisional application provides support for the claims of Krishnan (discussed in more detail below).

X. ANALYSIS AND OPINIONS

A. Claims 1-20 are Obvious Over Symons and Williams

1. Summary of Symons

55. The Symons Article, posted on Medium, and the Symons Video, posted on Vimeo, (collectively “Symons”) are authentic and were publicly available to a POSITA before the earliest priority date for the ’317 patent. Along the way in my experience as a scientist for the past 33 years, I have become personally familiar with Medium and Vimeo. Medium is a well-known Internet-based social journalism and weblogging (blogging) platform that is used by journalists and scientists to disseminate information about ideas, products, and technologies to interested members of the public, including POSITAs. Vimeo is a video sharing platform, similar to YouTube, accessible to anyone in the public. A POSITA would know that companies like RoboRana and Arinti would have used platforms such as Medium and Vimeo, which were readily accessible at that time through public search engines, and scientific citations and references. To the best of my knowledge, Medium and Vimeo are public (*e.g.*, no access restrictions), searchable, and intended to reach a wide audience. Both Medium and Vimeo are freely accessible, so a POSITA would not need a subscription or account to access the Symons Article or the Symons Video. A POSITA interested in the technology

would have known about Medium and Vimeo and would have readily located the information posted on these platforms with reasonable diligence.

56. The Medium page indicates that the Symons Article was posted on October 19, 2017. (EX1004, 1.) In my experience, members in the scientific and technical communities rely on the date posted on the front page.

57. The Symons Article includes the Symons Video. I compared the text of the Symons Article to the contents of the Symons Video, and I confirm that the text of the Symons Article agrees with the contents of the Symons Video.

(EX1004; EX1006.) The Symons Article indicates that it was posted by Senne Symons. (EX1004.) I confirm that Mr. Symons' name appears in the Symons Video. (EX1006, 17.)

58. The Baetens Article, posted on Medium, also includes a link to the Symons Article and a link to the Symons Video. (EX1009.) The Baetens Article is authentic and was publicly available to a POSITA before the earliest priority date for the '317 patent. The Medium page indicates that the Baetens Article was posted on October 20, 2017. (*Id.*) The date the article was posted is reflected on the front page. (*Id.*) In my experience, members in the scientific and technical communities rely on the date posted on the front page. I compared the text of the

Baetens Article to the contents of the Symons Article and the Symons Video, and I confirm that the text of the Baetens Article agrees with the contents of the Symons Article and the Symons Video. (EX1009; EX1004; EX1006.) Mr. Symons and Mr. Baetens both verified that they worked together on the development of the chatbot-RPA bot system described in the Symons Article, Symons Video, and Baetens Article. (EX1008; EX1010.) Thus, the Symons Article (which includes the Symons Video) was publicly available by at least October 20, 2017.

59. Symons discloses a computer program running on a computer system that “automate[s] the process of creating an email signature.” (EX1004, 2.) “First, the chatbot collects all the information in a friendly, interactive manner.” (*Id.*) The user provides a user instruction (“I want a new signature”), and then the chatbot asks a series of questions of the user to collect the user’s name, job role, email address, and mobile phone number. (EX1006, 2-6.) “What is your name?” (*Id.*, 5.) “What is your role?” (*Id.*) “What is your email address?” (*Id.*) “What is your cell phone number?” (*Id.*)

60. The chatbot then triggers the RPA bot. (*Id.*, 10.) The collected “information gets passed on to the RPA robot, which will then automatically generate the signature, save it, and email it back to the requester.” (EX1004, 2.)

The RPA bot automatically inputs the collected user information in text input fields to generate the email signature, and clicks on UI buttons to email the signature to the user. (EX1006, 10-17.)

61. As Symons explains, “[c]hatbots enable organizations to offer a more intuitive, natural way to interact with their customers. By handling and structuring problems and questions quicker, it greatly reduces waiting times and improves customer satisfaction.” (EX1004, 2.) “Robotics Process Automation (RPA) allows organizations to execute manual processes much faster, more accurate and reliable. A[n] RPA bot can mimic the users actions and thus enables organizations to automate repetitive tasks and ensures that those tasks are always executed with the same consistency and precision. It reduces costs and human efforts by running automated processes 24/7/365 and by adding a virtual workforce that work alongside the process owners.” (*Id.*) “Easy and smooth integration proves a great future for both technologies.” (*Id.*, 3.)

2. Summary of Williams

62. The Williams white paper is authentic and was publicly available to a POSITA before the earliest priority date for the '317 patent. Through my experience as a scientist for 33 years, and having worked for IBM Research for a

decade, I have become personally familiar with IBM and its website. IBM is a well-known technology company that disseminates information about its products and technological research advancements to interested members of the public, including POSITAs, on its website. To the best of my knowledge, IBM's website is public (*e.g.*, no access restrictions), searchable, and intended to reach a wide audience. A POSITA would be able to access IBM's website, including its white papers freely, without needing a subscription or account. A POSITA interested in the technology would have known about IBM's website and would have readily located the information with reasonable diligence.

63. The first and last pages of Williams indicates that it was available to the public no later than the September 2017 (date time stamps on the first and last pages). In my experience, members in the scientific and technical communities rely on the dates posted on the first and last pages.

64. Williams was also publicly available on IT World Canada's website to a POSITA before the earliest priority date for the '317 patent. IT World Canada is an online website that companies like IBM use to disseminate information about products to interested members of the public, including POSITAs. This website

was known, public (e.g., no access restrictions), searchable, and intended to reach a wide audience.

65. The IT World Canada page indicates that Williams was posted by at least September 16, 2017. The date the white paper was made publicly available is reflected on the front page. (EX1013.) In my experience, members in the scientific and technical communities rely on the date posted on the front page. A POSITA interested in the technology would have known about IT World Canada and would have readily located Williams on this website with reasonable diligence. Mr. Ira Allen verified that he worked on and made publicly available Williams. (EX1012.) Thus, Williams was publicly available by at least September 16, 2017.

66. Williams discloses a system and method that combines “two complementary forms of automation”: artificial intelligence and RPA. (EX1011, 1.) As Williams explains, RPA offers a significant reduction in operational costs, “[i]t’s not uncommon to find that we are able to drive 30% to 50% reductions in operational costs of transactional activities within shared services operations through use of RPA technologies.” (EX1011, 2.) As is common with most technologies, there are deployment challenges. (*Id.*, 2.) Williams proposes using cognitive computing to address the challenges of RPA. (*Id.*, 2.)

67. Williams' intelligent automation system comprises chatbots that engage in natural language dialog with a user to determine the user's intent. (*Id.*, 3.) "For example, chatbots or digital assistants are deployed within helpdesk functions to interact with an end user, replacing the human agent in a traditional call center. These chatbots can engage in natural language dialog with the caller and discern, through a series of interactions, the nature of the call." (*Id.*) The chatbot "capture[s] information required to further handle the request." (*Id.*)

68. The chatbot then triggers an RPA bot, passes the user information to the RPA bot, and the RPA bot completes the request. (*Id.*)

69. Fig. 1 (below) shows Williams' intelligent automation continuum. As part of robotic process automation (RPA), "[t]he 'robot' is taught to drive applications like a human does through the graphical user interface (GUI), following predetermined pathways." (*Id.*, Fig. 1.) The predetermined computer pathways may involve "conduct[ing] complex calculations, perform[ing] actions, and trigger[ing] downstream activities." (*Id.*) Intelligent process automation (IPA) adds to RPA with the robot "interact[ing] with humans through a combination of advanced algorithms, natural language processing, machine learning, and other forms of AI." (*Id.*)

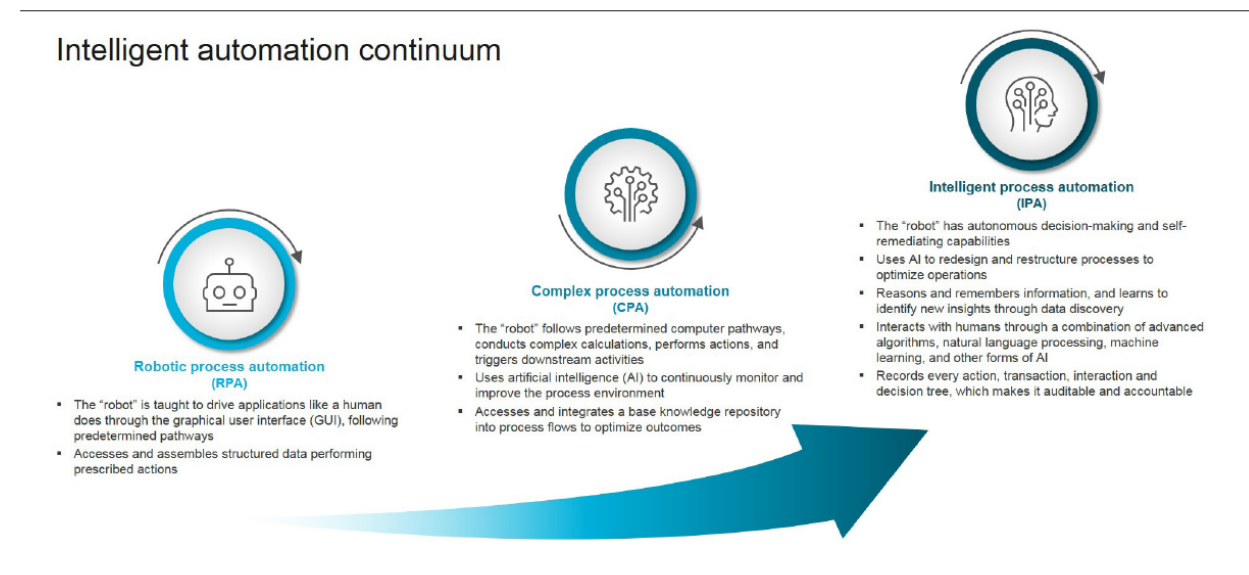


Figure 1. This diagram shows IBM's intelligent automation continuum.

3. Claim 1

- a. **Preamble: "A chat-bot program product, wherein the chat-bot program product comprises non-transitory computer readable medium retaining program instructions, which program instructions when read by a processor, cause the processor to:"**

70. Symons discloses the preamble. Symons discloses a chatbot-RPA computer system that "automate[s] the process of creating an email signature." (EX1004, 2.) A computer system comprises a non-transitory computer readable medium retaining program, such as a hard drive, that stores (retains) software (program instructions). A personal computer has a processor that accesses the software (read by a processor) and then performs the steps of the software.

b. 1[a]: “receive a user instruction from a user using a natural language interface;”

71. The screen below shows the conversation between Symon’s chatbot and the user, where the chatbot receives a user instruction from a user: “I want a new signature.”



72. According to the '317 patent, “[a] Chat bot is a computer program implementing natural language interface to conduct a conversation via auditory or textual methods with the user.” (EX1001, 1:28-30.)

73. As Symons describes, “[f]irst, the chatbot collects all the necessary information in a friendly, interactive manner. This information gets passed on to the RPA robot, which will then automatically generate the signature, save it, and email it back to the requester.” (EX1004, 2.) “The chatbot triggers the RPA bot.”

(EX1006, 10.) Symons discloses a chatbot-RPA bot system, which comprises a chatbot. As Symons explains, “[c]hatbots enable organizations to offer a more intuitive, natural way to interact with their customers. By handling and structuring problems and questions quicker, it greatly reduces waiting times and improves customer satisfaction. Eventually, a chatbot powered by AI will become indistinguishable from a human, leading to reduced costs and higher productivity.”

(EX1004, 2.) Interacting naturally with customers (humans) and becoming indistinguishable from a human means the chatbot communicates using natural language. Hence, Symons’ chatbot implements a “natural language interface.”

74. The conversation of the user is with a natural language interface. The chatbot asks a series of questions of the user to collect the user’s name, job role, email address, and mobile phone number. (EX1006, 3-7.) “What is your name?” (*Id.*, 5.) “What is your role?” (*Id.*) “What is your email address?” (*Id.*) “What is your cell phone number?” (*Id.*) The user responds with “Vincent De Bie,” “Engineer,” “vincent.debie@cronos.be,” which resemble language spoken by a human. (*Id.*) The questions and responses do not resemble, *e.g.*, programming code. Hence, Symons discloses element 1[a] (“receive a user instruction from a user using a natural language interface”).

c. 1[b]: “determine to implement an automation process on a web-based target system in response to the user instruction;”

75. Symons “automate[s] the process of creating an email signature” for the user. (EX1004, 2.) In response to the user instruction (“I want a new signature”), Symons determines to implement the automation process.

76. If the user does not provide a user instruction, it would have been obvious to a POSITA at the time of the ’317 priority date for Symons’ chatbot-RPA bot system to wait for the user to provide the instructions. Otherwise, if Symons’ chatbot-RPA bot system proceeds with creating an email signature without waiting for the user instruction, it may end up creating an email signature that is blank or has the wrong user information.

77. If the user provides a user instruction, then Symons’ chatbot-RPA bot system proceeds with the automatic email signature generation, determining to implement an automation process in response to the user instruction, where “the chatbot collects all the necessary information in a friendly, interactive manner. This information gets passed on to the RPA robot, which will then automatically generate the signature, save it, and email it back to the requester.” (EX1004, 2.) Then, “[t]he chatbot triggers the RPA bot.” (EX1006, 10.) Hence, Symons

discloses “determin[ing] to implement an automation process in response to the user instruction.”

78. If, for some reason, Symons does not determine to implement an automation process, it would have been obvious to a POSITA at the time of the ’317 priority date. Williams discloses a system that uses both chatbots and RPA bots. Chatbots are used to interface with a human, where it is common for humans to have more than one intent. “[C]hatbots or digital assistants [] deployed within helpdesk functions to interact with an end user, replacing the human agent in a traditional call center. These chatbots can engage in natural language dialog with the caller and discern, through a series of interactions, the nature of the call and then capture information required to further handle the request. The chatbot will then trigger a robot, passing the required information to the robot to complete the request.” (EX1011, 3.) Hence, the chatbot is able to determine what the human is requesting.

79. A POSITA would have been motivated to have a chatbot that discerns the user’s intent (the nature of the call) so that the RPA bot can handle multiple requests. RPA bots are useful for “repetitive, labor-intensive and transactional business processes.” (*Id.*, 2.) “Automation through RPA can offer a return on

investment (ROI) often ranging between 30 and 50 percent according to IBM internal analyses.” (*Id.*) “It’s not uncommon to find that we are able to drive 30% to 50% reductions in operational costs of transactional activities within shared services operations through use of RPA technologies.” (*Id.*) That is, RPA bots improve the costs and time for, *e.g.*, operations of transactional activities. This 30-50 percent ROI is achievable when the RPA bots are able to handle multiple requests. An RPA bot that is only able to handle a single request would be limited.

80. For example, Williams uses “chatbots for vendor inquiry helpdesk where it replaces a human agent to interface with suppliers who want to understand the status of their payments. Using chatbots, we have replaced the call center agent and now use a virtual agent to interact directly with the supplier. Through several scripted questions, the virtual agent can surmise that the supplier is asking for status on their payment and, thus, trigger an RPA robot to access the appropriate enterprise resource planning (ERP) backend system, validate the caller’s details, retrieve necessary information, and provide status back in real-time to the supplier. The chatbot then closes the call and updates the ticket management system used to track call center activities.” (*Id.*, 3.)

81. As discussed above and throughout Williams, RPA robots are designed to handle multiple requests, executing multiple different types of automation processes. Symons and Williams both disclose chatbots-RPA bot systems, where Williams teaches handling multiple requests. Combining Symons and Williams would expand the capabilities, making use of the advantages of RPA. The combination of Symons and Williams results in a chatbot-RPA bot system capable of handling more than a user request for automated generation of an email signature. The combination of Symons and Williams would result in a chatbot-RPA bot system that can handle other requests, such as automated generation of an electronic letterhead, automated generation of an electronic invoice, automated generation of an electronic flyer, automated generation of electronic business cards, etc.

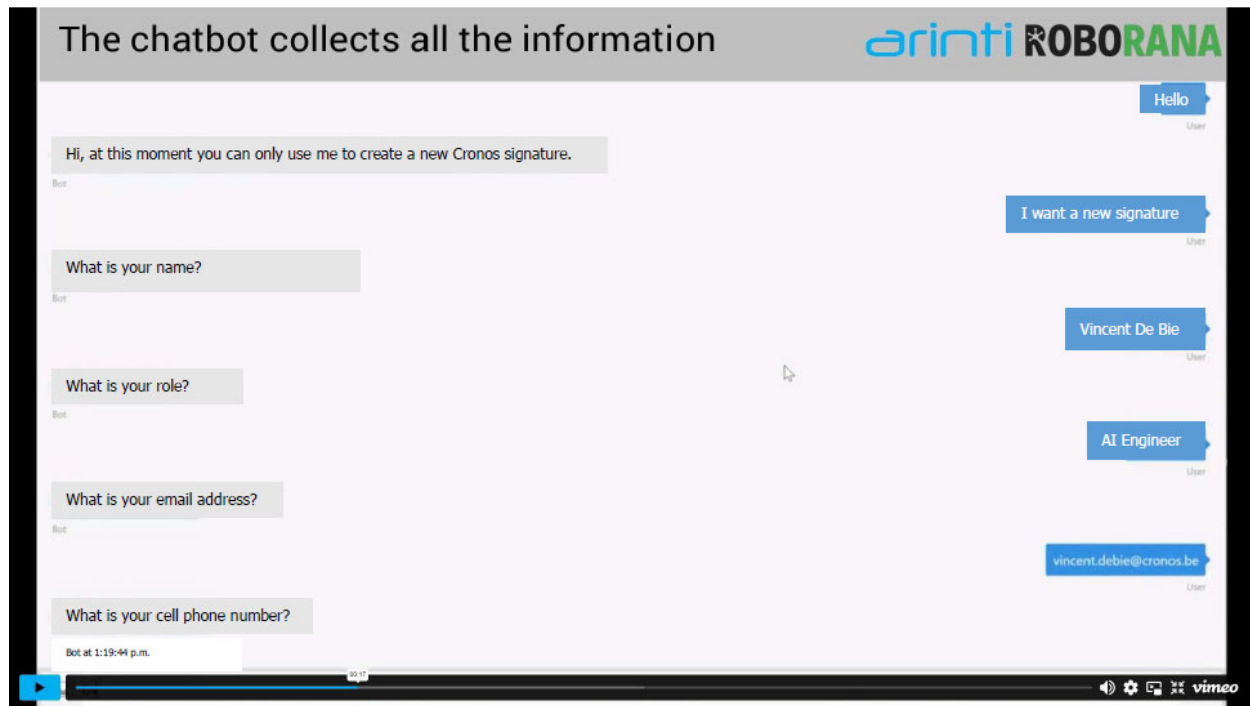
82. As Williams explains, the capabilities of a chatbot-RPA bot system may be expanded to handle multiple requests with a reasonable expectation of success. “[T]he ability to address these technical challenges will unlock significant additional value within RPA-enabled business operations.” (*Id.*, 3.) “In a real-life example within HR operations, the virtual agent has helped our contact centers improve in over 50 percent of our first contact resolutions, obtaining travel and

expense and payment status, updating our backend systems, and closing the inquiry in record time. Chatbots can be used in a variety of other processes and across industries such as mortgage, banking, insurance, healthcare and others to help reinvent call center operations.” (*Id.*) Accordingly, it would have been obvious to combine Symons and Williams to a POSITA at the time of the ’317 priority date.

83. Additionally, Williams discloses implementing the automation process on a web-based target system. “Although robots can be configured to interface directly with applications through pre-existing application programming interfaces (APIs), these platforms are equally able to work through systems’ graphical interfaces, including web browsers and Citrix terminal sessions.” (EX1011, 2.) Thus, Symons and Williams render obvious element 1[b].

d. 1[c]: “collect information to be used as part of the automation process; and”

84. Symons’ “chatbot collects all the information in a friendly, interactive manner,” as shown in the screenshot below. (EX1004, 2.)



85. “The chatbot collects all the information” by receiving input from the user. In the screenshot shown above, the chatbot asks the user for a name (“What is your name?”), job role (“What is your role?”), email address (“What is your email address?”), and mobile phone number (“What is your cell phone number?”). (EX1006, 5.) The user responds by using a keyboard to input data in the input field. (*Id.*) The chatbot’s questions and the user’s responses, which are displayed on a GUI and imitate human-to-human conversations, are part of a natural language conversation between the natural language interface and the user. (*Id.*)

86. The collected information is used as part of the automation process. After the chatbot collects all the information (“Vincent De Bie,” “Engineer,”

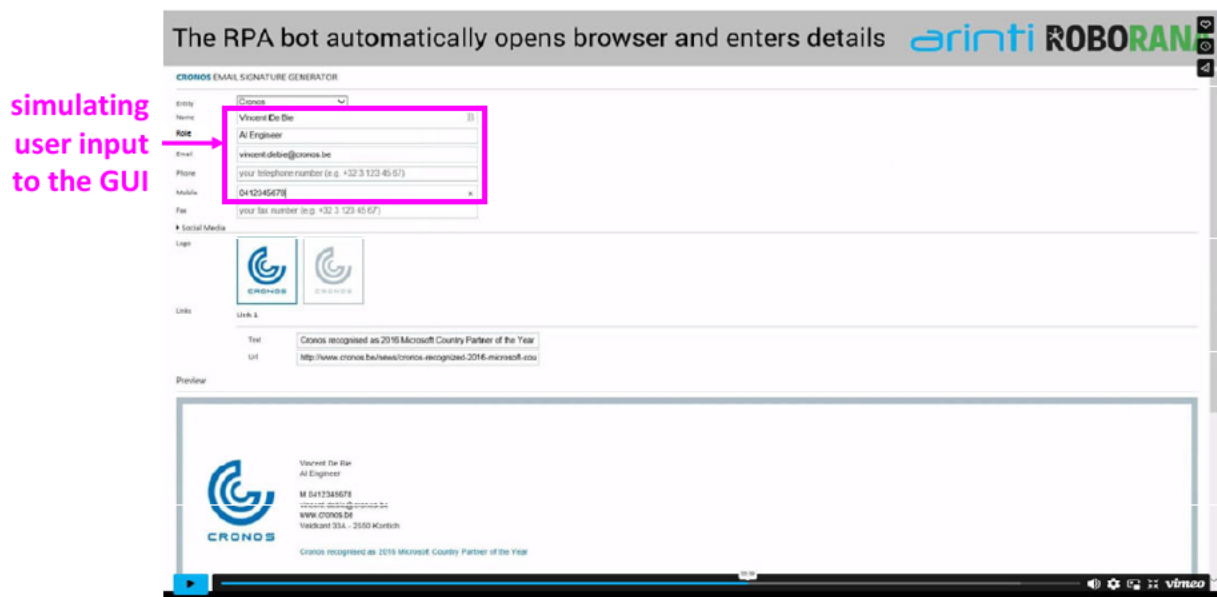
“vincent.debie@cronos.be,” “0412345678”), it utilizes the collected information in the automation process of creating an email signature. It utilizes “Vincent De Bie” as input in the name input field in the email signature, “Engineer” as input in the job role input field in the email signature, “vincent.debie@cronos.be” as input in the email input field in the email signature, and “0412345678” as input in the mobile number input field in the email signature. Hence, Symons discloses element 1[c] (“collect information to be used as part of the automation process”).

- e. **1[d]: “implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”**

87. As Symons explains, “Robotic Process Automation (RPA) allows organizations to execute manual processes must faster, more accurate and reliable. RPA bot can mimic the user’s actions and thus enables organizations to automate repetitive tasks. . . .” (EX1004, 2.) For generating an email signature, a graphical user interface GUI is displayed to the user, the user interacts with the GUI: inputting data into input fields on the GUI and clicking on certain GUI buttons. Symons’ RPA bot mimicking the user’s actions being part of an automation process means it implements the automation process by interacting with a GUI.

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

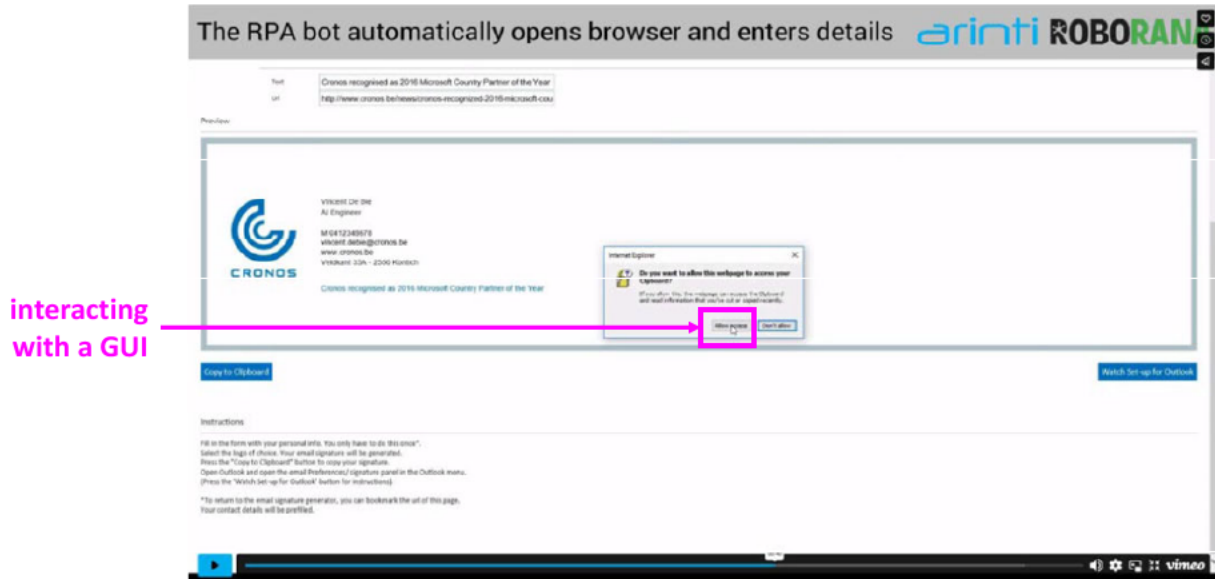
88. “The RPA bot automatically opens browser and enters details.”
(EX1006, 11.) As shown in the screen (annotated below), the RPA bot enters the user’s name “Vincent De Bie” in the name input field of the GUI and “Engineer” in the job role input field (annotated below). (*Id.*, 10). The RPA bot also enters the user’s email address “vincent.debie@cronos.be” in the email address input field and mobile number “0412345678” in the mobile number input field. (*Id.*, 12.)



89. Additionally, the RPA bot automatically clicks on certain GUI buttons. As shown in the annotated screen below, the RPA bot clicks on the “Allow access” GUI button when asked whether to allow the webpage to access

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

the Clipboard. This GUI button click is an interaction with the GUI via GUI button clicks (annotated below). (*Id.*, 13.)



90. As discussed for element 1[b], Symons and Williams render obvious implementing the automation process on a web-based target system. Thus, Symons-Williams renders obvious element 1[d] (“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system”).

- f. **1[e]: “wherein the automation process is performed with respect to third-party web-based target system and”**

91. Williams discloses “[a]lthough robots can be configured to interface directly with applications through pre-existing application programming interfaces (APIs), these platforms are equally able to work through systems’ graphical interfaces, including web browsers and Citrix terminal sessions.” (EX1011, 2.) Working through a web browser means interfacing with a third-party web-based target system.

92. As discussed for element 1[b], Symons and Williams render obvious performing the automation process on a web-based target system. Thus, Symons-Williams renders obvious element 1[e] (“wherein the automation process is performed with respect to third-party web-based target system”).

g. 1[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

93. As Symons confirms in his declaration, Symons’ RPA bot implements an automation process without relying on an Application Programming Interface (API). (EX1011.)

94. If, for some reason, Symons does not disclose implementing the automation process without relying on an API, it would have been obvious to a POSITA at the time of the ’317 priority date. Williams discloses “RPA enlists

software robots to perform complex, nested routines that cut across multiple applications and interact with these systems without the need to build complex and rigid system-to-system interfaces. Although robots can be configured to interface directly with applications through pre-existing application programming interfaces (APIs), these platforms are equally able to work through systems' graphical interfaces, including web browsers and Citrix terminal sessions.” (*Id.*, 2.)

95. Williams explains the benefits, specifically shorter times for deployment, lower upfront costs, and lower operational costs, to implementing the automation process without relying on an API. “This means that robotic automation can be deployed in a relatively short period of time measured in days and weeks, instead of more heavy, traditional IT programming efforts that can take months and require a larger upfront investment. This makes software robots an attractive way to drive operational cost reduction” (*Id.*, 2.)

96. It would have been obvious to combine Symons and Williams to a POSITA at the time of the '317 priority date. Symons interfaces with multiple platforms: email signature platform, Microsoft Word, and Microsoft Outlook. As Williams explains, RPA bots interface across multiple platforms: “[r]obotic process automation (RPA) is the application of technology that allows computer

software or a ‘robot’ to capture and interpret existing applications for processing a transaction, manipulating data, triggering responses and communicating with other digital systems.” (*Id.*) Hence, RPA allows interfacing with multiple platforms, where a POSITA would be motivated to combine because RPA bots may further reduce costs, time, and efforts when implemented without relying on an API. Traditional IT programming efforts are not needed, and as Williams explains, the prior art has demonstrated a reasonable expectation of success.

97. As discussed for element 1[b], Symons-Williams renders obvious implementing the automation process on a web-based target system. The interaction among systems without the need to build complex and rigid system-to-system interfaces that Williams teaches means there is no exposure by a web-based system. Thus, Symons-Williams renders obvious element 1[f].

h. 1[g]: “wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

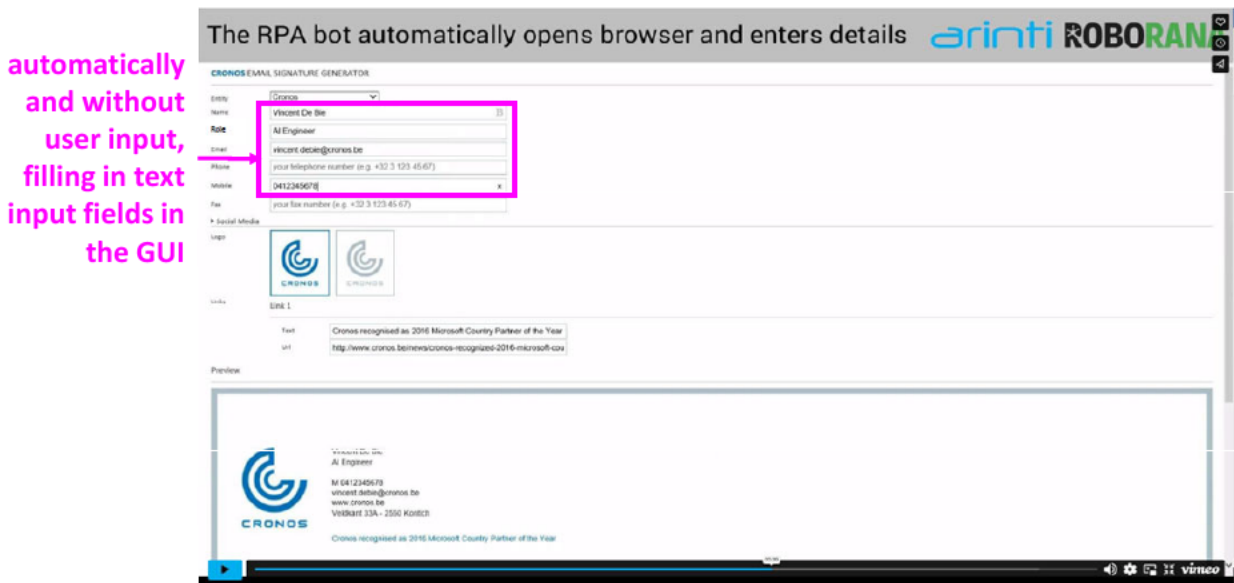
98. Symons discloses its “RPA bot automatically opens browser and enters details.” (EX1006, 11.) Symons’ sequence of interactions involves filling in the collected information (entering details) in text input fields in the GUI. In Symons, the text input fields are the name, job role, email address, and mobile

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

number text input fields (as shown in the annotated screen below). (EX1006, 12.)

The RPA bot fills in text input fields in the GUI with “Vincent De Bie,”

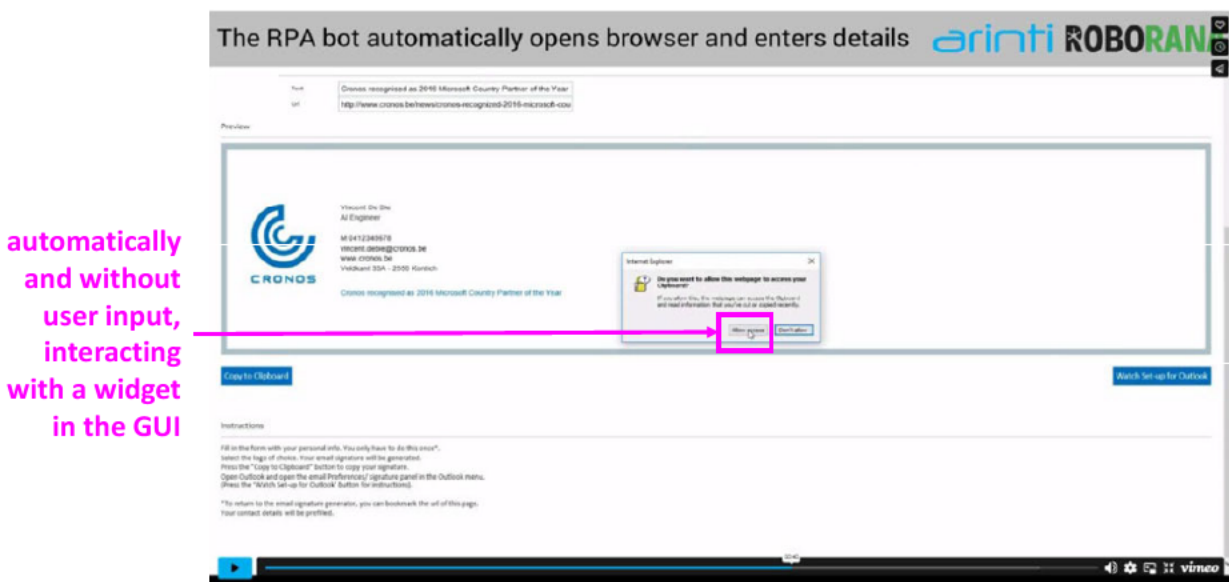
“Engineer,” “vincent.debie@cronos.be,” and “0412345678” automatically and without user input. (*Id.*)



99. As discussed for element 1[b], the GUI is of a web-based target system. Hence, Symons-Williams renders obvious element 1[g] (“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system”).

- i. **1[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

100. Symons' RPA bot automatically and without user input interacts with a widget in the GUI. As shown in the screen (annotated below), the RPA bot simulates user GUI button clicks that copies the email signature onto the clipboard. (EX1006, 13.) According to the '317 patent, a widget is a GUI element. (EX1001, 10:19-21). Hence, Symon's GUI button is a widget in a GUI.

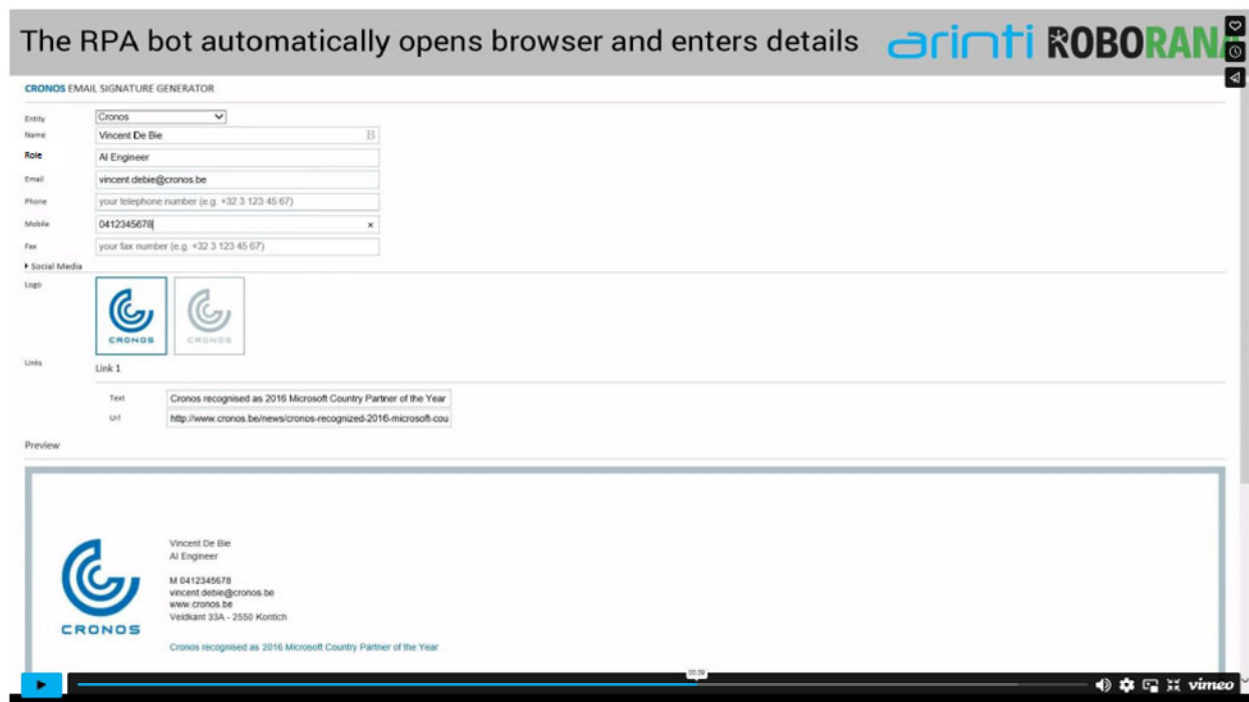


101. Copying the email signature onto the clipboard involves performing a predefined functionality. As discussed for element 1[b], the GUI is of a web-based target system. Hence, Symons-Williams renders obvious element 1[h] ("automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality").

4. Claim 2: "The chat-bot program product of claim 1, wherein the chat-bot program product is configured to

implement the automation process by a client device used by the user.”

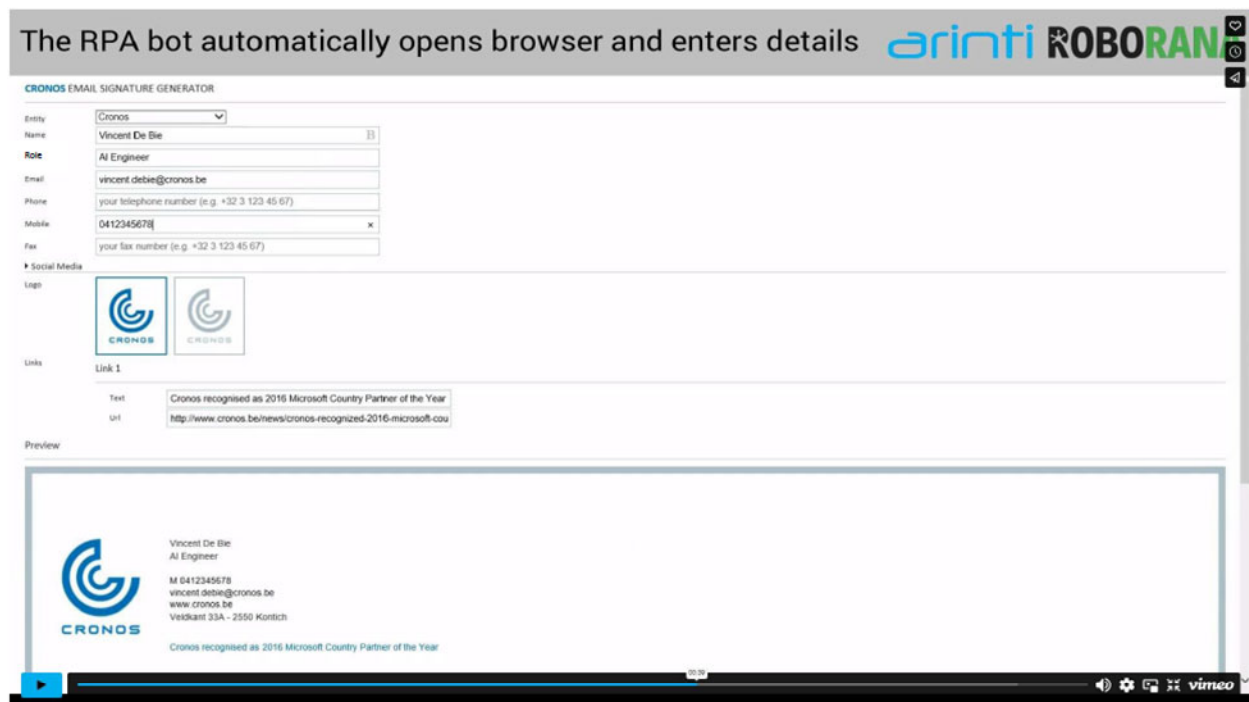
102. Symons’ chatbot-RPA bot implements an automation process of creating an email signature, as shown in the screen below. The screen is a computer screen of a personal computer, where the “RPA bot automatically opens [a computer program] browser and enters details.” (EX1006, 11.) A personal computer is a client device used by a user. Thus, Symons discloses claim 2 (“wherein the chat-bot program product is configured to implement the automation process by a client device used by the user”).



5. Claim 3: “The chat-bot program product of claim 2, wherein the chat-bot program product is configured to

implement the automation process by a browser that is executed by the client device.”

103. As discussed for element 1[b], Symons’ chatbot-RPA bot implements an automation process. The “RPA bot automatically opens [a computer program] browser and enters details.” (EX1006, 11.) The screen below is a computer screen of a personal computer, showing a browser that is executed by a client device. Thus, Symons discloses claim 3 (“wherein the chat-bot program product is configured to implement the automation process by a browser that is executed by the client device”).



6. Claim 4: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to

converse with the user in natural language using a vocal interface.”

104. As discussed for element 1[a], Symons’s chatbot is a chat-bot program product configured to converse with the user in a natural language. As Symons describes, “[f]irst, the chatbot collects all the necessary information in a friendly, interactive manner. This information gets passed on to the RPA robot, which will then automatically generate the signature, save it, and email it back to the requester.” (EX1004, 2.) “The chatbot triggers the RPA bot.” (EX1006, 10.)

105. The chatbot asks a series of questions of the user to collect the user’s name, job role, email address, and mobile phone number. (*Id.*, 3-7.) “What is your name?” (*Id.*, 5.) “What is your role?” (*Id.*) “What is your email address?” (*Id.*) “What is your cell phone number?” (*Id.*) The user responds with “Vincent De Bie,” “Engineer,” “vincent.debie@cronos.be,” which resemble language spoken by a human. (*Id.*) The questions and responses do not resemble, *e.g.*, programming code.

106. As discussed for claims 2 and 3, Symons’ “RPA bot automatically opens [a computer program] browser and enters details.” (EX1006, 11.) Hence, Symons’ system is executed on a personal computer.

107. One input component commonly implemented on a personal computer in 2017 is a microphone that receives vocal commands from a human user. Vocal commands may be a more convenient way for a human user to input information for, *e.g.*, the conversation with the natural language interface, and thus, it would have been obvious to receive vocal commands. Williams discloses such voice-based service: “[f]or example, chatbots or digital assistants are deployed within helpdesk functions to interact with an end user, ***replacing the human agent in a traditional call center.***” (EX1011, 3 (emphasis added).) Williams further discloses: “[t]hese chatbots can engage in natural language dialog with the caller and discern, through a series of interactions, the nature of the call.” (*Id.*) Thus, Symons-Williams renders obvious claim 4 (“wherein the chat-bot program product is configured to converse with the user in natural language using a vocal interface”).

7. Claim 5

- a. 5[a]: “The chat-bot program product of claim 1, wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field,”**

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

108. As discussed for element 1[g], the automation process of Symons' chatbot-RPA bot is configured to automatically fill in a text input field in a GUI, and it would have been obvious for the GUI to be of a web-based target system.

109. The screen (below) shows the GUI including a form comprising text input fields that the chatbot-RPA bot automatically fills information into. A form of a GUI of a web-based target system is an online form. Thus, Symons-Williams renders obvious element 5[a] ("wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field").

automatically
fill information
in a form,
wherein the
form comprises
the text input
field

The RPA bot automatically opens browser and enters details

CRONOS EMAIL SIGNATURE GENERATOR

Copy

Name: Vincent De Bie

Role: AI Engineer

Email: vincent.debie@cronos.be

Phone: your telephone number (e.g. +32 3 123 45 67)

Mobile: 04 123 45 678

Social Media: your social media (e.g. Facebook, LinkedIn)

Logos: CRONOS, MICROSOFT

Link 1: CRONOS recognized as 2016 Microsoft Country Partner of the Year

Text: CRONOS recognized as 2016 Microsoft Country Partner of the Year

URL: <http://www.cronos.be/news/cronos-recognized-2016-microsoft-country-partner>

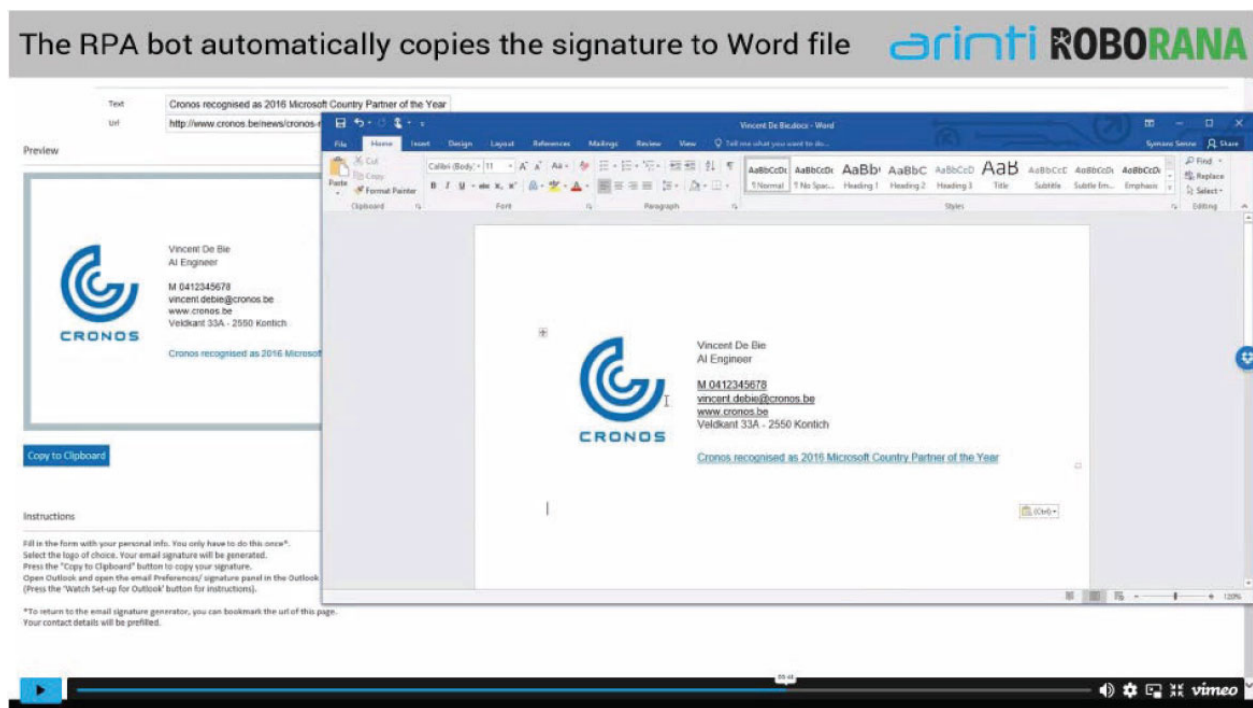
Preview: Vincent De Bie, AI Engineer, M. 04 123 45 678, vincent.debie@cronos.be, www.cronos.be, Boulevard 11A, 1160A Willebroek, CRONOS recognized as 2016 Microsoft Country Partner of the Year

- b. 5[b]: “wherein the widget is configured to invoke submission of the online form,”

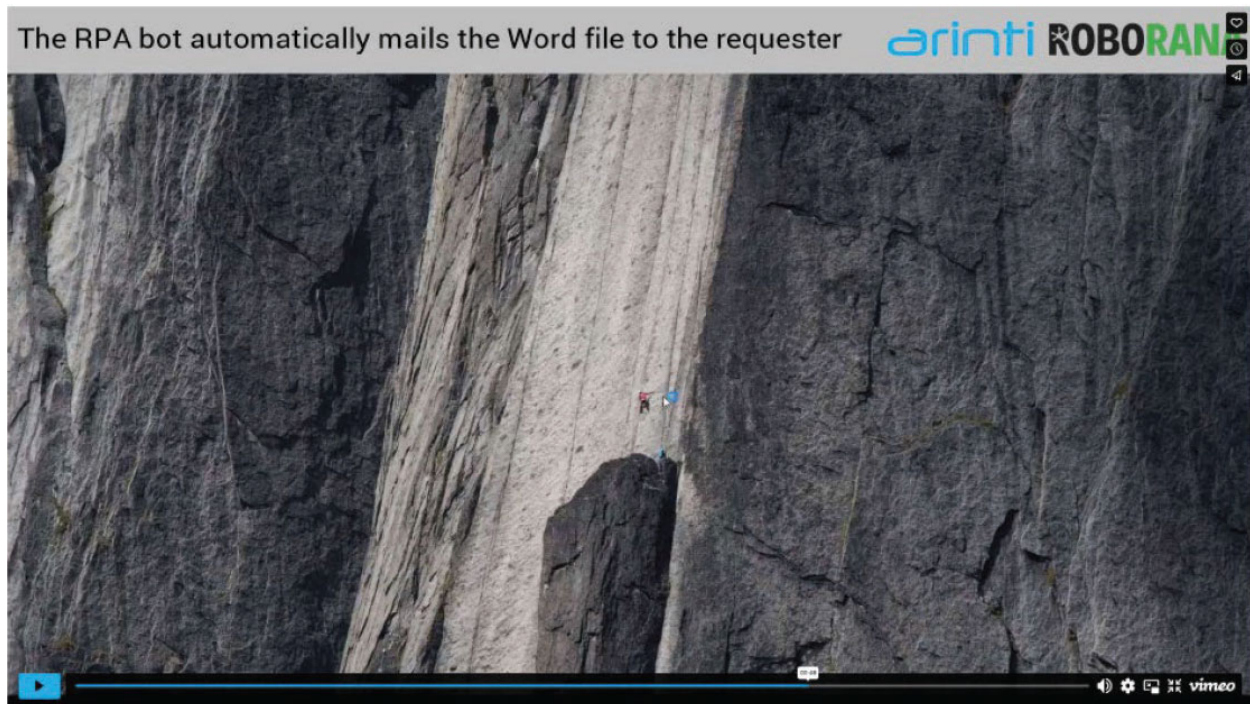
Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

110. As shown for element 1[h], Symons discloses a GUI button that when clicked copies the email signature onto the clipboard. (EX1006, 13.) The GUI button is a widget. (EX1001, 10:19-21)

111. After copying the signature to the clipboard, the “RPA bot automatically copies the signature to a Word file” (screen (below)). (EX1006, 15.)



112. Then, as shown in the screen below, the “RPA bot automatically emails the Word file to the requester.” (*Id.*, 16.)



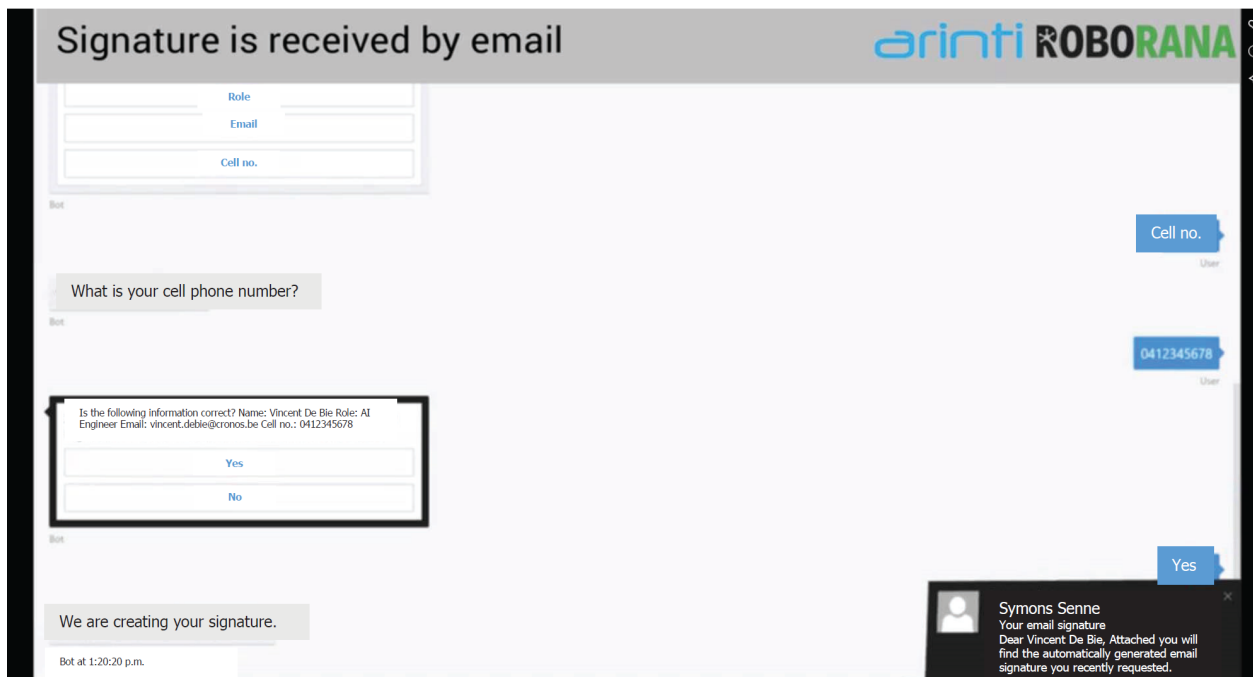
113. When the GUI button is clicked, the email signature is generated. The widget must have invoked submission of the form because otherwise the email signature would not have been generated. Additionally, the widget invoked submission of the email signature to the requester. Thus, Symons discloses element 5[b] (“wherein the widget is configured to invoke submission of the online form”).

- c. **5[c]: “wherein the chat-bot program product is configured to request user validation from the user before submitting the online form.”**

114. As shown in element 5[b], Symons’ chatbot-RPA bot submits the form before generating the email signature.

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

115. Before submitting the form and before inputting the collected information in the text inputs fields of the form, Symons' chatbot asks the user to verify the user's information: "Is the following information correct? Name: Vincent De Bie Role: AI Engineer Email: vincent.debie@cronos.be Cell no.: 04123456" (screen below (annotated)). Asking the user to verify user information is requesting user validation from the user.



116. As discussed for element 5[a], it would have been obvious for Symons' form to be an online form. Thus, Symons renders obvious element 5[c] ("wherein the chat-bot program product is configured to request user validation from the user before submitting the online form").

8. Claim 6: “The chat-bot program product of claim 1, wherein the information comprises information of the user, whereby personalizing the automation process.”

117. Symons’ chatbot asks a series of questions of the user to collect information of the user: the user’s name, job role, email address, and mobile phone number. (EX1006, 3-7.) “What is your full name?” (*Id.*, 5.) “What is your role?” (*Id.*) “What is your email address?” (*Id.*) “What is your cell phone number?” (*Id.*) The user responds with “Vincent De Bie,” “Engineer,” “vincent.debie@cronos.be,” and “0412345678.” (*Id.*)

118. “[T]he chatbot collects all the necessary information in a friendly, interactive manner. This information gets passed on to the RPA robot, which will then automatically generate the signature, save it, and email it back to the requester.” (EX1004, 2.) Then, “[t]he chatbot triggers the RPA bot.” (EX1006, 10.) The collected information is personalized information used as part of the automation process, here, as part of the personalized email signature that is generated. Thus, Symons discloses claim 6 (“wherein the information comprises information of the user, whereby personalizing the automation process”).

9. Claim 7: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to utilize credentials of the user, whereby the chat-bot is privy to user-specific information.”

119. As discussed for claim 6, Symons' chatbot-RPA bot utilizes the user's full name ("Vincent De Bie"), role ("Engineer"), email address ("vincent.debie@cronos.be"), and mobile phone number ("0412345678) in generating the email signature. (EX1006, 3-7, 11-13.)

120. According to the '317 patent, "the user may associate parameters available to the chat bot, such as based on the user credentials[.]" (EX1001, 20:14-16.) The '317 patent provides example user credentials: Employee ID, First Name, and Last Name. (*Id.*, 20:17-27.)

121. Hence, in Symons, the user's full name, role, email address, and mobile phone number are credentials of the user. The user's full name, role, email address, and mobile phone number are user-specific information that is provided to the chat-bot, so the chat-bot is privy to the user-specific information. Thus, Symons discloses claim 7.

10. Claim 8: "The chat-bot program product of claim 1, wherein implementation of the automation process is performed by automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI."

122. As shown for element 1[h], Symons' system automatically interacts with a widget in the GUI. Symons' system also automatically performs the "[t]he

chatbot trigger[ing] the RPA bot,” “[t]he RPA bot automatically opens browser and enters details,” “[t]he RPA bot automatically copies the signature to Word file,” and “[t]he RPA bot automatically mail the Word file to the requester” (among other steps). (EX1006, 10-16.)

123. The system automatically identifies the GUI button (widget) in the GUI. (*Id.*, 13; EX1033.) The system also automatically identifies the text input fields (name text input field, role text input field, email text input field, mobile text input field) in the GUI. (EX1006, 11, 12; EX1033.) Thus, Symons discloses claim 8 (“wherein implementation of the automation process is performed by automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI”).

11. Claim 9

a. Preamble: “A method comprising:”

124. Symons discloses the preamble. Symons discloses “automat[ing] the process of creating an email signature.” (EX1004, 2.) A process is a method.

b. 9[a]: “receiving a user instruction from a user using a natural language interface of a chat-bot;”

125. Element 9[a] is similar to element 1[a], but requires the natural language interface be “of a chat-bot.” The table below highlights the differences.

Element 1[a]	Element 9[a]
“receive a user instruction from a user using a natural language interface.”	“receiving a user instruction from a user using a natural language interface <i>of a chat-bot</i> ”

126. As shown for the preamble of claim 1, Symons discloses a chat-bot.

Symons discloses element 1[a], and thus, element 9[a].

c. 9[b]: “determining to implement an automation process on a web-based target system in response to the user instruction;”

127. Element 9[b] mirrors element 1[b]. The table below shows the similarities. Symons-Williams renders obvious element 1[b], and thus, element 9[b].

Element 1[b]	Element 9[b]
“determine to implement an automation process on a web-based target system in response to the user instruction;”	“determining to implement an automation process on a web-based target system in response to the user instruction;”

d. 9[c]: “collecting information to be used as part of the automation process; and”

128. Element 9[c] mirrors element 1[c]. The table below shows the similarities. Symons discloses element 1[c], and thus, element 9[c].

Element 1[c]	Element 9[c]
“collect information to be used as part of the automation process; and”	“collecting information to be used as part of the automation process; and;”

- e. **9[d]: “implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”**

129. Element 9[d] mirrors element 1[d]. The table below shows the similarities. Symons-Williams renders obvious element 1[d], and thus, element 9[d].

Element 1[d]	Element 9[d]
“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”	“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

- f. **9[e]: “wherein the automation process is performed with respect to third-party web-based target system and”**

130. Element 9[e] mirrors element 1[e]. The table below shows the similarities. Symons-Williams renders obvious element 1[e], and thus, element 9[e].

Element 1[e]	Element 9[e]
“wherein the automation process is performed with respect to third-party web-based target system and”	“wherein the automation process is performed with respect to third-party web-based target system and”

g. 9[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

131. Element 9[f] mirrors element 1[f]. The table below shows the similarities. Symons-Williams renders obvious element 1[f], and thus, element 9[f].

Element 1[f]	Element 9[f]
“without relying on an Application Programming Interface (API) exposed by the web-based target system,”	“without relying on an Application Programming Interface (API) exposed by the web-based target system,”

h. 9[g]: “wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

132. Element 9[g] mirrors element 1[g]. The table below shows the similarities. Symons-Williams renders obvious element 1[g], and thus, element 9[g].

Element 1[g]	Element 9[g]
“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”	“wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

- i. **9[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

133. Element 9[h] mirrors element 1[h]. The table below shows the similarities. Symons-Williams renders obvious element 1[h], and thus, element 9[h].

Element 1[h]	Element 9[h]
“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”	“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

- 12. Claim 10: “The method of claim 9, wherein said implementing is performed by a client device used by the user.”**

134. Claim 10 recites limitations similar to claim 2, but does not require a “chat-bot program product [be] configured to” implement the automation process.

The table below highlights the differences. Symons discloses claim 2, and thus, claim 10.

Claim 2	Claim 10
“wherein <i>the chat-bot program product is configured to implement the automation process</i> by a client device used by the user.”	“wherein <i>said implementing is performed</i> by a client device used by the user.”

13. Claim 11: “The method of claim 10, wherein said implementing is performed by a browser that is executed by the client device.”

135. Claim 11 recites limitations similar to claim 3, but does not require a “chat-bot program product [be] configured to” implement the automation process. The table below highlights the differences. Symons discloses claim 3, and thus, claim 11.

Claim 3	Claim 11
“wherein <i>the chat-bot program product is configured to implement the automation process</i> by a browser that is executed by the client device.”	“wherein <i>said implementing is performed</i> by a browser that is executed by the client device.”

14. Claim 12: “The method of claim 9, wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

136. Claim 12 recites limitations similar to claim 4, but requires a “chat-bot” instead of a “chat-bot program product” be configured to converse with

the user in natural language using a vocal interface. A chat-bot program product is a chat-bot. Symons, alone or in combination with Williams, renders obvious claim 4, and thus, claim 12.

Claim 4	Claim 12
“wherein the chat-bot program product is configured to converse with the user in natural language using a vocal interface.”	“wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

15. Claim 13

- a. 13[a]: “The method of claim 9, wherein the automation process is configured to automatically fill information in an online form in the GUI, wherein the online form comprises the text input field,”

137. Element 13[a] recites limitations similar to element 5[a], but requires filling information in an online form “in the GUI.” The table below highlights the differences. As shown for element 5[a], Symons-Williams renders obvious filling information in a form of a GUI of a web-based target system, which is filling information in an online form in the GUI.

Element 5[a]	Element 13[a]
“wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field,”	“wherein the automation process is configured to automatically fill information in an online form in the GUI , wherein the online form comprises the text input field,”

b. 13[b]: “wherein the widget is configured to invoke submission of the online form,”

138. Element 13[b] mirrors element 5[b]. The table below shows the similarities. Symons discloses element 5[b], and thus, element 13[b].

Element 5[b]	Element 13[b]
“wherein the widget is configured to invoke submission of the online form,”	“wherein the widget is configured to invoke submission of the online form,”

c. 13[c]: “wherein the method further comprises requesting user validation from the user before submitting the online form.”

139. Element 13[c] mirrors element 5[c], but does not require the “chat-bot program product [be] configured” to request user validation. The table below highlights the differences. Symons renders obvious element 5[c], and thus, element 13[c].

Element 5[c]	Element 13[c]
“wherein <i>the chat-bot program product is configured to</i> request user validation from the user before submitting the online form.”	“wherein the method further comprises requesting user validation from the user before submitting the online form.”

16. Claim 14: “The method of claim 9, wherein the information comprises information of the user, whereby personalizing the automation process.”

140. Claim 14 mirrors claim 6. The table below shows the similarities.

Symons discloses claim 6, and thus, claim 14.

Claim 6	Claim 14
“wherein the information comprises information of the user, whereby personalizing the automation process.”	“wherein the information comprises information of the user, whereby personalizing the automation process.”

17. Claim 15: “The method of claim 9, wherein said implementing the automation process comprises: automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”

141. Claim 15 recites limitations similar to claim 8. The table below shows the similarities. Symons discloses claim 8, and thus, claim 15.

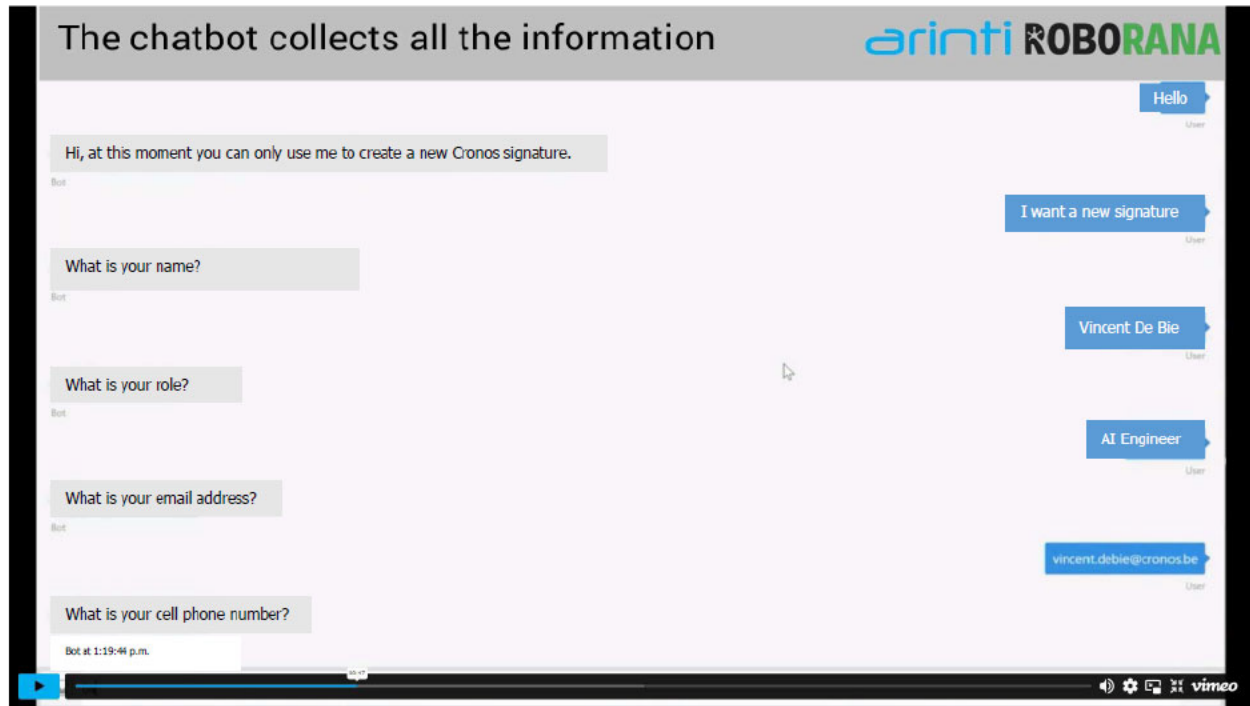
Claim 8	Claim 15
“wherein implementation of the automation process is performed by automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI.”	“wherein said implementing the automation process comprises: automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”

18. Claim 16

a. Preamble: “A computerized apparatus comprising a processor and coupled memory, the processor being adapted to perform:”

142. Symons discloses the preamble. Symons’ chatbot-RPA system is a computer system (computer screen shown below). The computer system runs the

chatbot-RPA program, Microsoft Word, and Microsoft Outlook. (EX1006, 15, 19-21.) A computer system is a computerized apparatus comprising a processor and coupled memory.



b. 16[a]: “receiving a user instruction from a user using a natural language interface of a chat-bot;”

143. Element 16[a] is similar to element 1[a], but requires the natural language interface be “of a chat-bot.” The table below highlights the differences.

Element 1[a]	Element 16[a]
“receive a user instruction from a user using a natural language interface.”	“receiving a user instruction from a user using a natural language interface <i>of a chat-bot</i> ”

144. As shown for the preamble of claim 1, Symons discloses a chat-bot. Symons discloses element 1[a], and thus, element 16[a].

c. 16[b]: “determining to implement an automation process on a web-based target system in response to the user instruction;”

145. Element 16[b] mirrors element 1[b]. The table below shows the similarities. Symons-Williams renders obvious element 1[b], and thus, element 16[b].

Element 1[b]	Element 16[b]
“determine to implement an automation process on a web-based target system in response to the user instruction;”	“determining to implement an automation process on a web-based target system in response to the user instruction;”

d. 16[c]: “collecting information to be used as part of the automation process; and”

146. Element 16[c] mirrors element 1[c]. The table below shows the similarities. Symons discloses element 1[c], and thus, element 16[c].

Element 1[c]	Element 16[c]
“collect information to be used as part of the automation process; and”	“collecting information to be used as part of the automation process; and;”

e. 16[d]: “implementing the automation process by interacting with a Graphical User Interface (GUI) of

the web-based target system and simulating user input to the GUI of the web-based target system,”

147. Element 16[d] mirrors element 1[d]. The table below shows the similarities. Symons-Williams renders obvious element 1[d], and thus, element 16[d].

Element 1[d]	Element 16[d]
“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”	“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

f. 16[e]: “wherein the automation process is performed with respect to third-party web-based target system and”

148. Element 16[e] mirrors element 1[e]. The table below shows the similarities. Symons-Williams renders obvious element 1[e], and thus, element 16[e].

Element 1[e]	Element 16[e]
“wherein the automation process is performed with respect to third-party web-based target system and”	“wherein the automation process is performed with respect to third-party web-based target system and”

g. 16[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

149. Element 16[f] mirrors element 1[f]. The table below shows the similarities. Symons-Williams renders obvious element 1[f], and thus, element 16[f].

Element 1[f]	Element 16[f]
“without relying on an Application Programming Interface (API) exposed by the web-based target system,”	“without relying on an Application Programming Interface (API) exposed by the web-based target system,”

- h. 16[g]: “wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”**

150. Element 16[g] mirrors element 1[g]. The table below shows the similarities. Symons-Williams renders obvious element 1[g], and thus, element 16[g].

Element 1[g]	Element 16[g]
“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”	“wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

- i. 16[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

151. Element 16[h] mirrors element 1[h]. The table below shows the similarities. Symons-Williams renders obvious element 1[h], and thus, element 16[h].

Element 1[h]	Element 16[h]
“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”	“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

19. Claim 17: “The computerized apparatus of claim 16, wherein the computerized apparatus is a client device used by the user, wherein said implementing is performed by a browser that is executed by the client device.”

152. A portion of claim 17 recites limitations similar to claim 2 and claim 3, but requires a “computerized apparatus” be a client device and be configured to implement the automation process. The table below shows the differences.

Claim 2	Claim 3	Claim 17
“wherein <i>the chat-bot program</i> product is configured to implement the automation process by a client device used by the user.”	“wherein <i>the chat-bot program product</i> is configured to implement the automation process <i>by a browser that is executed by</i> the client device.”	“wherein the <i>computerized apparatus</i> is a client device used by the user, wherein said implementing is <i>performed by a browser that is executed by</i> the client device.”

153. A chat-bot program product is a computerized apparatus. Symons discloses claim 2 and claim 3, and thus, claim 17.

20. Claim 18: “The computerized apparatus of claim 16, wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

154. Claim 18 recites limitations similar to claim 4, but requires a “chat-bot” instead of a “chat-bot program product” be configured to converse with the user in natural language using a vocal interface. A chat-bot program product is a chat-bot. Symons, alone or in combination with Williams, renders obvious claim 4, and thus, claim 18.

Claim 4	Claim 18
“wherein the chat-bot <i>program product</i> is configured to converse with the user in natural language using a vocal interface.”	“wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

21. Claim 19

- a. 19[a]: “The computerized apparatus of claim 16, wherein the automation process is configured to automatically fill information in an online form in the GUI, wherein the online form comprises the text input field,”**

155. Element 19[a] recites limitations similar to element 5[a], but requires filling information in an online form “in the GUI.” The table below highlights the

differences. As shown for element 5[a], Symons-Williams renders obvious filling information in a form of a GUI of a web-based target system, which is filling information in an online form in the GUI.

Element 5[a]	Element 19[a]
“wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field,”	“wherein the automation process is configured to automatically fill information in an online form <i>in the GUI</i> , wherein the online form comprises the text input field,”

b. 19[b]: “wherein the widget is configured to invoke submission of the online form,”

156. Element 19[b] mirrors element 5[b]. The table below shows the similarities. Symons discloses element 5[b], and thus, element 19[b].

Element 5[b]	Element 19[b]
“wherein the widget is configured to invoke submission of the online form,”	“wherein the widget is configured to invoke submission of the online form,”

c. 19[c]: “wherein the method further comprises requesting user validation from the user before submitting the online form.”

157. Element 19[c] mirrors element 5[c], but does not require the “chat-bot program product [be] configured” to request user validation. The table below

highlights the differences. Symons renders obvious element 5[c], and thus, element 19[c].

Element 5[c]	Element 19[c]
“wherein <i>the chat-bot program product is configured to</i> request user validation from the user before submitting the online form.”	“wherein the method further comprises requesting user validation from the user before submitting the online form.”

22. **Claim 20: “The computerized apparatus of claim 16, wherein said implementing the automation process comprises: automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”**

158. Claim 20 recites limitations similar to claim 8. The table below shows the similarities. Symons discloses claim 8, and thus, claim 20.

Claim 8	Claim 20
“wherein implementation of the automation process is performed by automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI.”	“wherein said implementing the automation process comprises: automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”

B. Claims 1-20 are Obvious Over Allen

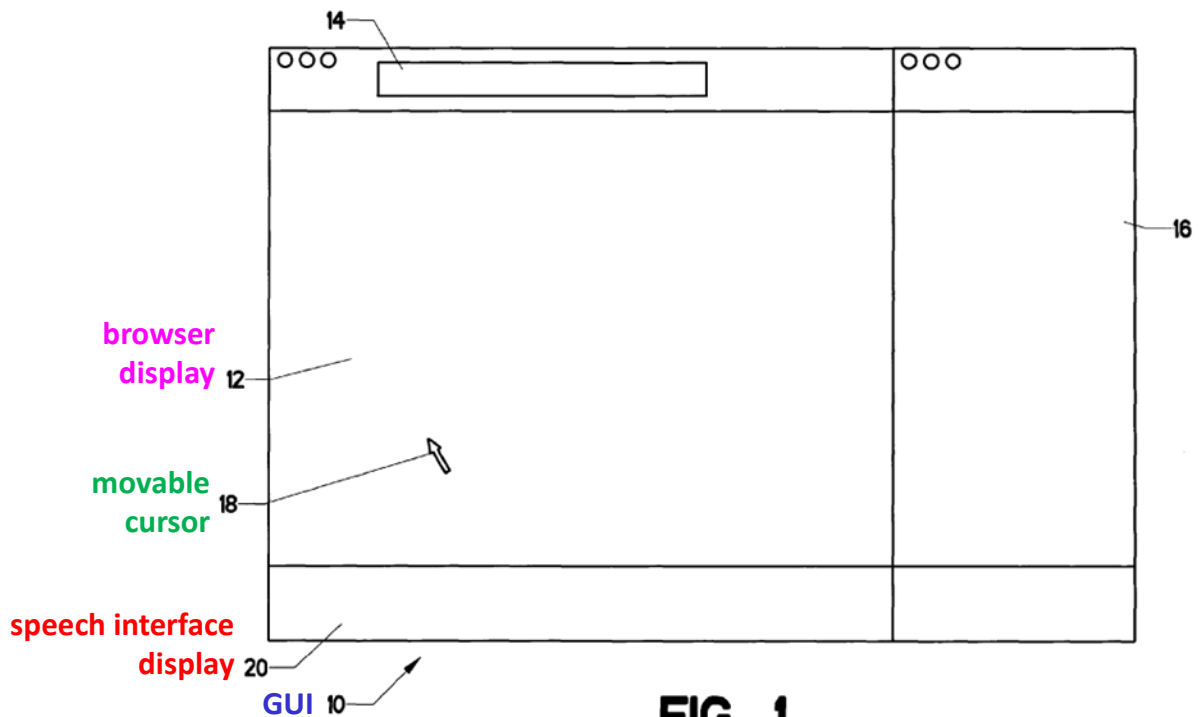
1. Summary of Allen

159. Allen discloses a “system which allows a user to teach a computational device how to perform complex, repetitive tasks that the user

usually would perform using the device's graphical user interface (GUI) often but not limited to being a web browser. (EX1014, Abstract.) "The user 'teaches' task steps by inputting natural language and demonstrating actions with the GUI." (*Id.*)

160. FIG. 1 (annotated below) shows the GUI 10, which includes a "[s]peech interface display 20 [] presented across the bottom of the display area. These displays allow the user to monitor his or her interaction with [the system] PLOW" (*Id.*, 5:17-20.)

161. The GUI 10 also includes a "[b]rowser display 12 [] presented on the left side of the display area." (*Id.*, 5:7-8.) The browser display 12 displays a web page corresponding to the website displayed in "URI entry box 14 [] displayed near the windows top." (*Id.*, 5:9-12.) The movable cursor 18 is also shown in GUI 10. (*Id.*, 5:12-13.) "After a complete task has been taught, the system is able to automatically execute the task in new situations." (*Id.*, Abstract.) "The system [] execute[s] the steps it previously learned" from the user. (*Id.*, 3:45-46.)



1. Claim 1

- a. **Preamble: “A chat-bot program product, wherein the chat-bot program product comprises non-transitory computer readable medium retaining program instructions, which program instructions when read by a processor, cause the processor to:”**

162. Allen discloses the preamble. Allen discloses a “system [that] includes software running on a user’s computational device.” (EX1014, Abstract.) The software is referred to as “Procedural Learning on the Web” (PLOW). (*Id.*, 4:64-66.) The “software includes a natural dialog-based interface whereby said

user can communicate with said software using natural dialog-based language.”

(*Id.*, Claim 1.)

163. According to the ’317 patent, “[a] Chat bot is a computer program implementing natural language interface to conduct a conversation via auditory or textual methods with the user.” (EX1001, 1:28-30.)

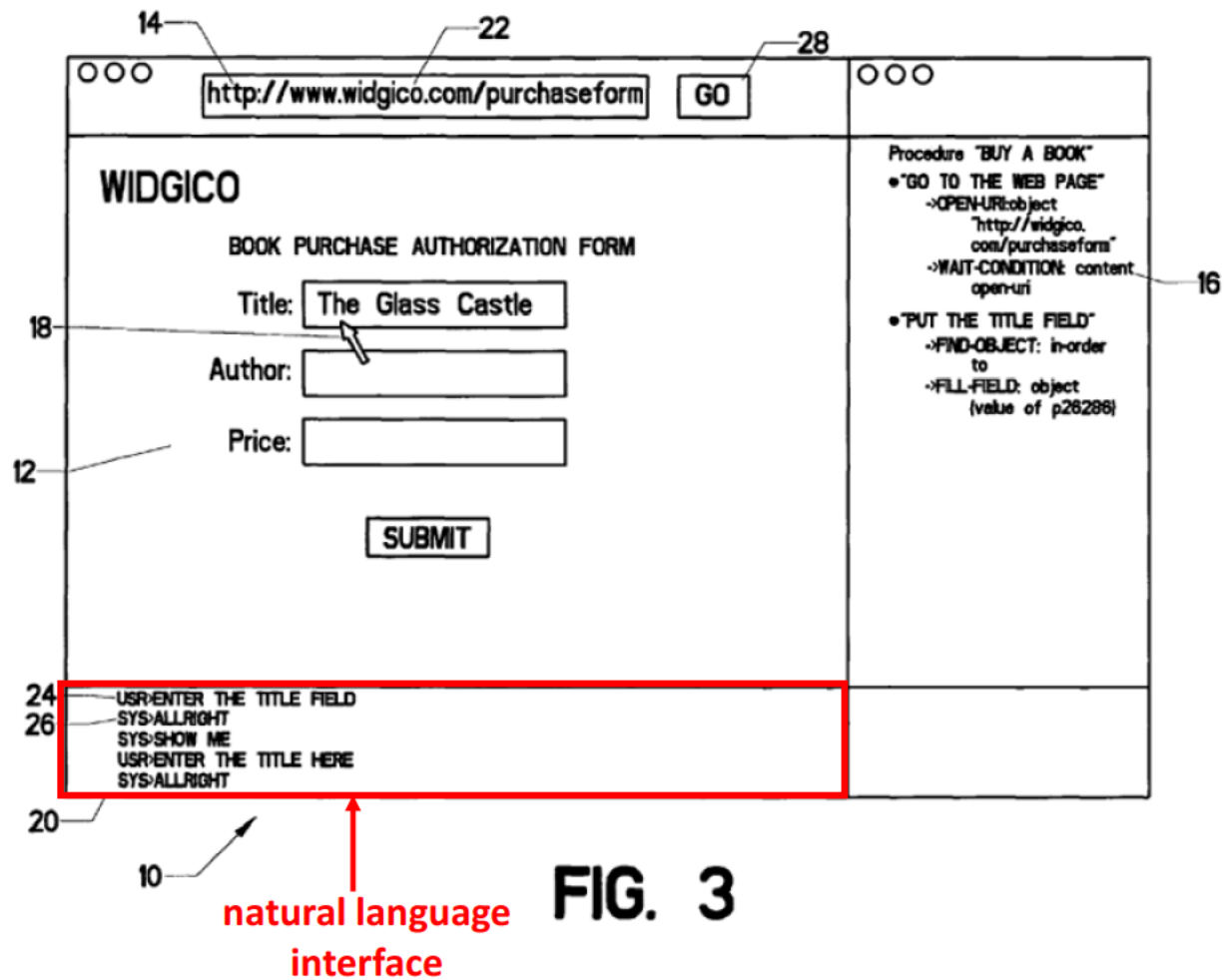
164. A computational device comprising software that includes a natural dialog-based interface that communicates with a user is a chat-bot program product.

165. Software comprises program instructions. A computational device comprises a non-transitory computer readable medium, such as a hard drive, that stores (retains) software (program instructions). A personal computer has a processor that accesses the software (read by a processor) and then performs the steps of the software.

b. 1[a]: “receive a user instruction from a user using a natural language interface;”

166. Allen’s Fig. 3 (annotated below) shows the speech interface display 20 showing the user’s speech 24 (input from a user) and the system’s speech 26 (part of the natural language conversation). The “software includes a natural

dialog-based interface whereby said user can communicate with said software using natural dialog-based language.” (EX1014, Claim 1.)



167. Allen discloses “[o]nce a task has been successfully taught to the system, a user can ask the system to perform that task. The system will then execute the steps it previously learned.” (*Id.*, 3:44-46.) For example, the user may

say, “I want to buy a book,” which means the system receives a user instruction from a user. (*Id.*, 12:64-65.)

168. Allen’s GUI 10 comprises display areas, such as task log 16 (“presented on the right side of the display area. The task log shows the tasks that have been learned so far in this earning session”) and speech interface display 20 (“is presented across the bottom of the display area. These displays allow the user to monitor his or her interaction with [the system] PLOW”). (*Id.*, 5:15-20.) Thus, Allen discloses element 1[a] (“receive a user instruction from a user using a natural language interface”).

c. 1[b]: “determine to implement an automation process on a web-based target system in response to the user instruction;”

169. As discussed for element 1[a], Allen discloses a user instruction being, “I want to buy a book.” (EX1014, 12:64-65.) In response to the user instruction, system would then determine to implement a book buying task (automation process). “The system would then ask, ‘What is the title of the book?’ Once information is provided, [the system] PLOW would then perform the steps it was previously taught and complete the transaction.” (*Id.*, 12:66-13:1.)

170. The book buying task (automation process) is implemented on a web-based target system. As Allen explains, “FIG. 7 [(below)] shows the process continued. Browser display 12 shows the ‘AMAZON.COM’ website (For purposes of visual clarity it is shown in simplified form). . . . The task log also shows the identification of a new sub-task of going to a new website (shown as sub-task label 540).” (*Id.*, 12:31-37.)

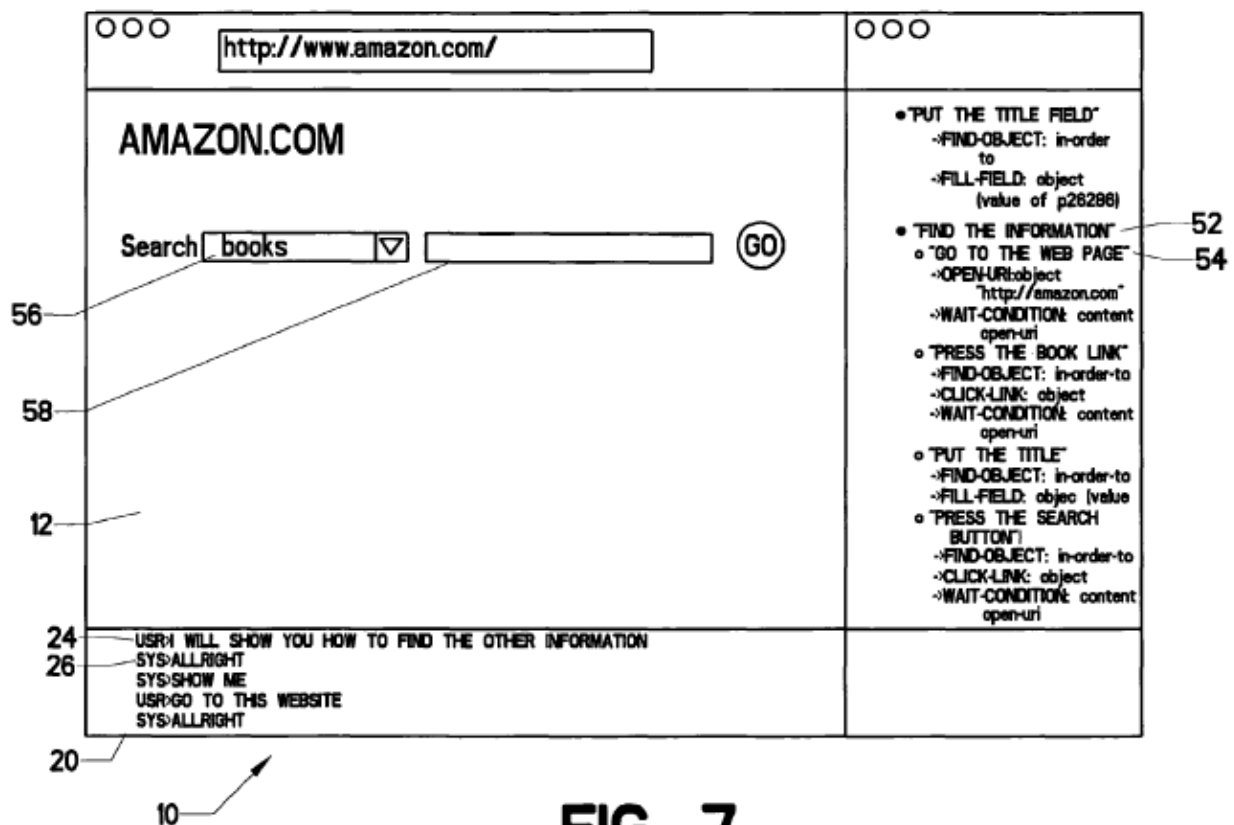


FIG. 7

171. In response to a different user instruction, the system would determine to implement a different automation process. For example, “a user conducts a search on the website known as ‘MAPQUEST.COM.’ Specifically, the user searches for hotels which are near the address ‘1018 E. De Soto St. Pensacola, Fla. 32501-3362.’” (*Id.*, 16:33-40.) The hotel searching task would be implemented in response to a user instruction such as the user requesting the system search for hotels, and the system automatically completing a transaction of searching for hotels. MAPQUEST.COM is an example web-based target system. Thus, Allen discloses element 1[b] (“determine to implement an automation process on a web-based target system in response to the user instruction”).

d. 1[c]: “collect information to be used as part of the automation process; and”

172. Allen discloses its system asking the user questions to collect information. For example, the user may provide a user instruction: “I want to buy a book.” (EX1014, 12:64-65.) The system asks questions to collect information such as “What is the title of the book?” (*Id.*, 12:65-66.) The user provides an answer, the title of the book, to the system’s question. The system may also “obtain price and delivery date information, then provide this information back to

the user and ask whether the user wishes to complete the transaction.” (*Id.*, 13:3-5.)

173. “When a user starts a new task, *the system first asks for the values of the task input parameter.*” (*Id.*, 17:64-65 (emphasis added).) “Once the values are provided, the system executes the task.” (*Id.*, 17:67-18:1.) “Once [all the] information is provided, [the system] PLOW would then perform the steps it was previously taught and complete the transaction.” (*Id.*, 12:66-13:1.) In the book buying task, the system uses the title of the book and optionally price and delivery date information to complete the transaction of buying the book (automation process). (*Id.*, 12:62-13:5.) Hence, Allen discloses element 1[c] (“collect information to be used as part of the automation process”).

- e. **1[d]: “implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”**

174. Allen’s system implements an automation process by interacting with a GUI of a web-based target system. As Allen explains, “[f]or each step a user teaches the system, the system attempts to recognize the user’s intent. The system then learns to automatically execute steps by observing the user’s actions and analyzing the verbal instructions the user provides.” (EX1014, 6:61-65.)

175. For the book buying task, the system learns how to execute the steps of buying a book by observing the user interacting with a GUI of a web-based target system. The “user [] instructed the system to ‘Enter the title here’ (referring to a book title to be entered in title entry box 30).” (*Id.*, 12:12-14.) The “system will utilize the processes previously described to search the HTML code and identify the text object corresponding to the book’s title. . . . The system then submits the title as entered by the user.” (*Id.*, 12:19-25.) Fig. 3 (below) illustrates the title entry box as located in the browser display 12, which means the GUI is of a web-based target system. The example web-based target system shown in Fig. 3 is www.widgico.com. (*Id.*, 6:6-12.)

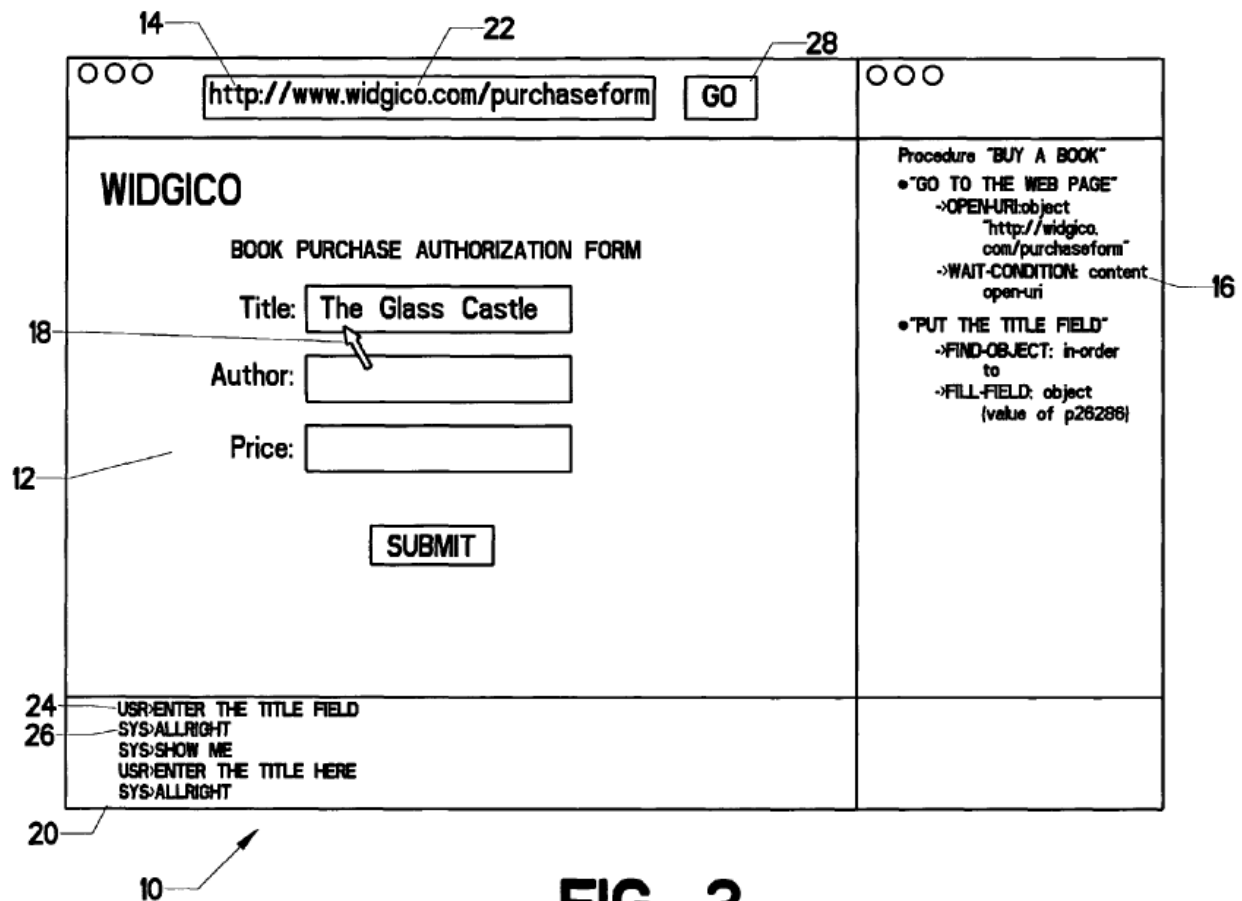


FIG. 3

176. The system also learns from the user (and thus, later executes) other user interactions with a GUI. The system learns “how to navigate to an online vendor site.” (*Id.*, 12:30-31.) The system also learns how to “pull[] down and select[] the ‘Books’ category in search category box 56,” “picking text entry box 58,” and “actuat[ing] the “GO” button.” (*Id.*, 12:38-46.) Additionally, the “user would then teach the system to open a specific book page and identify the price.

The price would then be stored and entered back on the book purchase form.” (*Id.*, 12:56-59.)

177. The system learns by “observing the user’s actions.” (*Id.*, 6:61-65.) These user actions are interactions with a GUI of the web-based target system, including input to the GUI, as discussed above. As a result, the system performing the steps it was previously taught means it simulates user input to the GUI of the web-based target system. The system performs the steps in browser display 12, which is part of the GUI 10 (displayed UI) (“[b]rowser display 12 [] presented on the left side of the display area.”), as shown in Fig. 1 (below). (*Id.*, 5:7-8.)

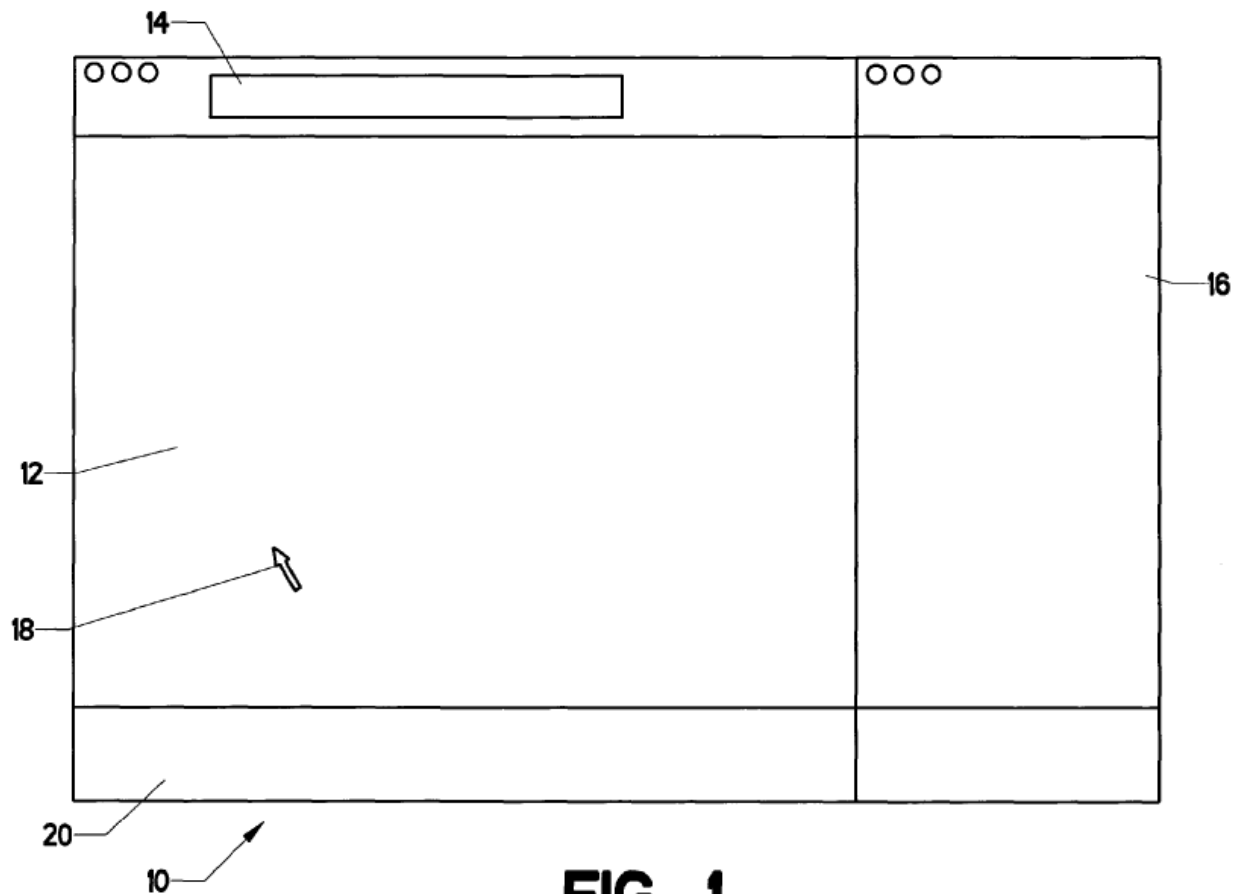


FIG. 1

178. If, for some reason, Allen does not disclose implementing the automation process by interacting with a Graphical User Interface (GUI), then it would have been obvious to a POSITA at the time of the '317 priority date.

Allen's GUI 10 comprises display areas, such as task log 16 ("presented on the right side of the display area. The task log shows the tasks that have been learned so far in this earning session") and speech interface display 20 ("is presented across the bottom of the display area. These displays allow the user to monitor his

or her interaction with [the system] PLOW”). (*Id.*, 5:15-20.) A “graphical user interface (‘GUI’) is preferably provided so that the user can easily monitor [the system’s] actions.” (*Id.*, 4:66-5:1.) It would have been obvious that in order for the user to easily monitor the system’s actions, the system’s actions would have to involve interaction with a GUI. As discussed above, the system’s actions are performed in browser display 12 of a GUI.

179. Thus, Allen discloses or renders obvious element 1[d] (“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system”).

f. 1[e]: “wherein the automation process is performed with respect to third-party web-based target system and”

180. Allen discloses the automation process “involv[ing] web-based operations. As stated previously, the inventive system can be configured to run within a web browser.” (EX1014, 4:61-63.) As shown for element 1[d], the automation process is implemented by interacting with the browser display 12 of GUI 10 (Fig. 2 (below)). For example, “[t]he user [] speaks the phrase ‘Go to this web page.’ [The system] PLOW then displays the message ‘USR>GO TO THE

WEB PAGE' in the speech interface display 20. PLOW analyzes the speech and recognizes it as a step in the procedure it is currently learning." (*Id.*, 5:63-67.)

The system "can monitor many actions, including a pointing device location and mouse button 'clicks.' As an example, it can note the fact that the user has selected a particular link within an open web page." (*Id.*, 7:29-32.)

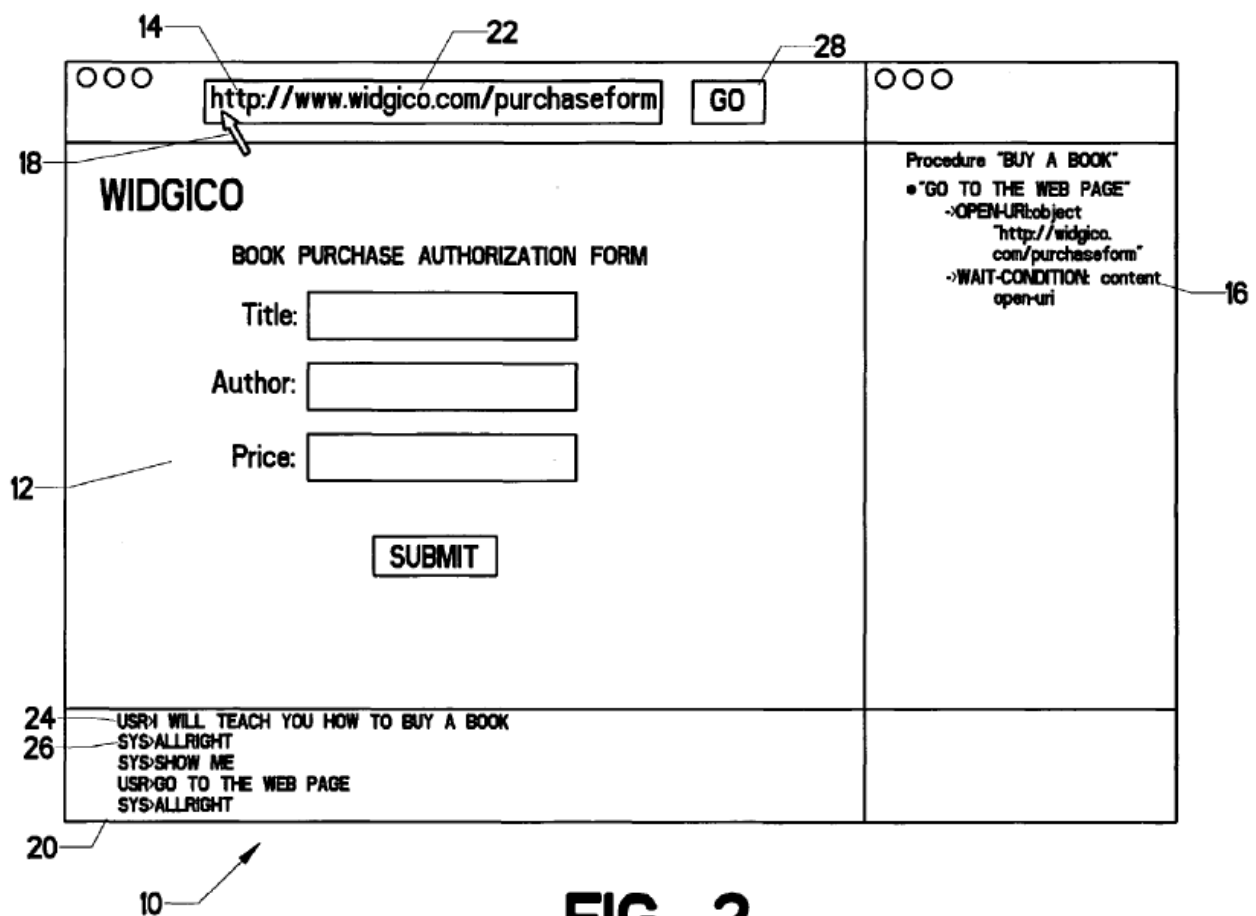


FIG. 2

181. Allen discusses example web-based target systems:

`www.widgico.com`, `www.amazon.com`, `www.mapquest.com`. (*Id.*, 6:6-10,

6:14-16, 6:24-26, 11:44-50, 12:31-33, 16:35-38, Fig. 3, Fig. 7, Fig. 12.) Widgico, Amazon, and Mapquest are third-party web-based target systems. Thus, Allen discloses element 1[e] (“wherein the automation process is performed with respect to third-party web-based target system”)

g. 1[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

182. Allen discloses “the system can be configured to function in a web-based environment in which information is transferred using HTML, XHTML, CSS, and/or scripts.” (EX1014, 3:26-31.) The “user ‘teaches’ task steps by inputting natural language and demonstrating actions with the GUI.” (*Id.*, Abstract.) Thus, Allen discloses “without relying on an Application Programming Interface (API) exposed by the web-based target system.” The automation process being implemented without relying on an API exposed by the web-based target system means the automation process may implement a functionality without knowledge of a predefined API, namely, the automation process can deduce the execution process from other means, such as observing the user interaction. The fact that a non-programmer user can perform this interaction in the GUI without providing any line of code means that it does not rely on an API.

- h. 1[g]: “wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”**

183. Allen discloses a “system which allows a user to teach a computation device how to perform complex, repetitive tasks that the user usually would perform using the device’s graphical user interface (GUI)” (EX1014, Abstract.) The user demonstrates actions, and the system observes. (*Id.*, 6:5-6, 6:61-65.) The user “presses transmit button 28, which causes the web browser to ‘navigate’ to the URI,” “select[s] the input box” (clicks a GUI button using a mouse), “mov[es] the mouse to a text input box” (moves a mouse cursor), (pulls down a drop down box), and “typ[es] in a title” (types into an input field). (*Id.*, 6:5-7, 6:10-11, 6:20-22, 7:43-45, 12:38-41, 12:45-46.) Hence, Allen discloses teaching the system to fill in a text input field in the GUI.

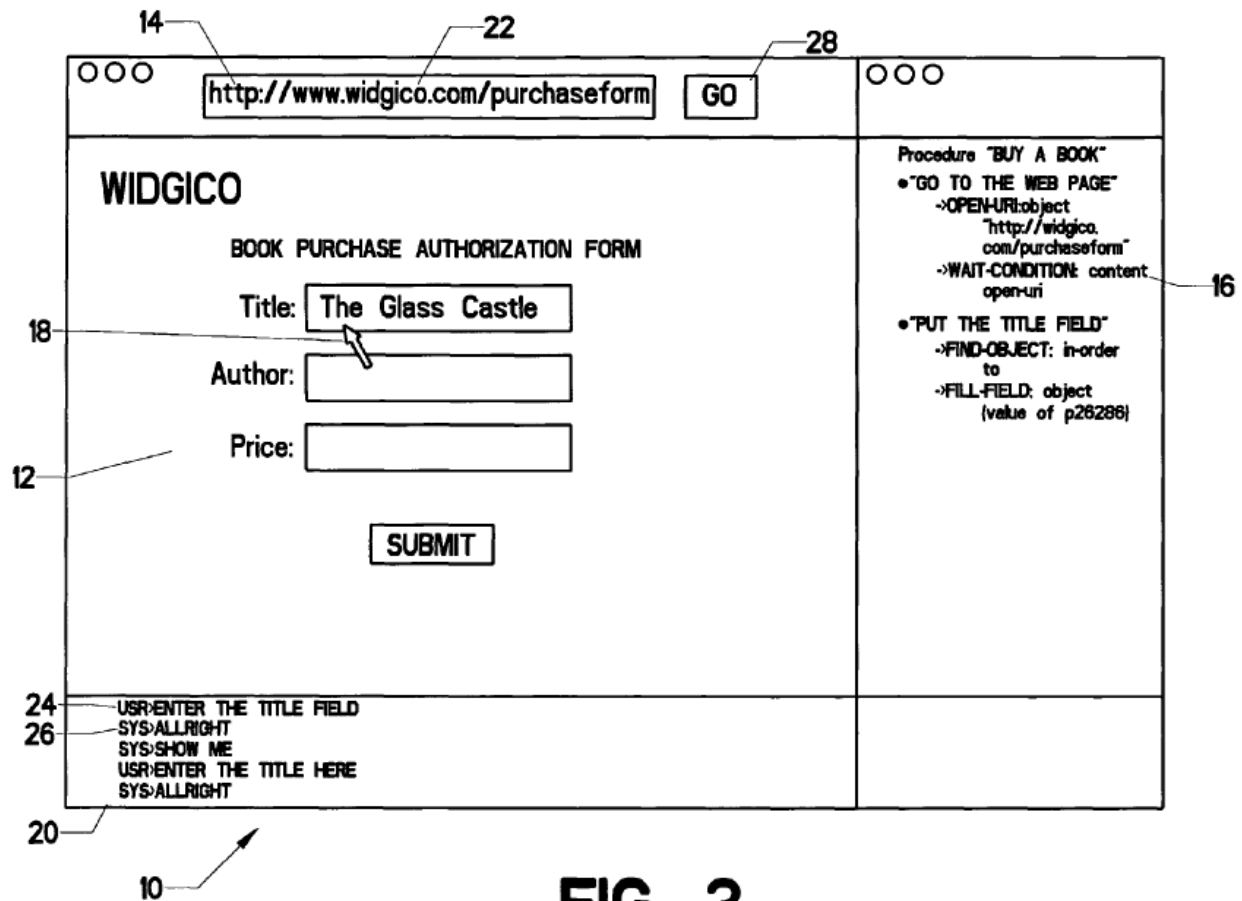
184. Allen discloses “[a]fter a complete task has been taught, the system is able to automatically execute the task, in new situations.” (*Id.*, Abstract.) “The ‘user ‘teaches’ task steps by inputting natural language and demonstrating actions with the GUI.” (*Id.*) Since the system is taught to fill in a text input field in the GUI, then the system automatically executing the task it was taught means the system automatically and without user input fills in a text input field in the GUI.

185. As discussed for element 1[b], Allen’s GUI is of a web-based target system. Thus, Allen discloses element 1[g] (“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system”).

- i. **1[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined-functionality.”**

186. Allen discloses a “system which allows a user to teach a computation device how to perform complex, repetitive tasks that the user usually would perform using the device’s graphical user interface (GUI)” (EX1014, Abstract.) The user demonstrates actions, and the system observes. (*Id.*, 6:5-7, 6:61-65.) The user “presses transmit button 28, which causes the web browser to ‘navigate’ to the URI” or “select[s] the input box” (clicks a GUI button using a mouse), “moves a computer mouse so that movable cursor 18 moves up to URI box 14” “mov[es] the mouse to a text input box” (moves a mouse cursor), (pulls down a drop down box to select a category in a search category box 56), and “typ[es] in a title” (types into an input field). (*Id.*, Abstract, 6:5-7, 6:10-11, 6:20-22, 7:43-45, 12:38-41, 12:45-46.) Fig. 3 (below) show the transmit button 28

and input box. Fig. 7 (below) shows a text entry box 58 (text input box) and drop down box.



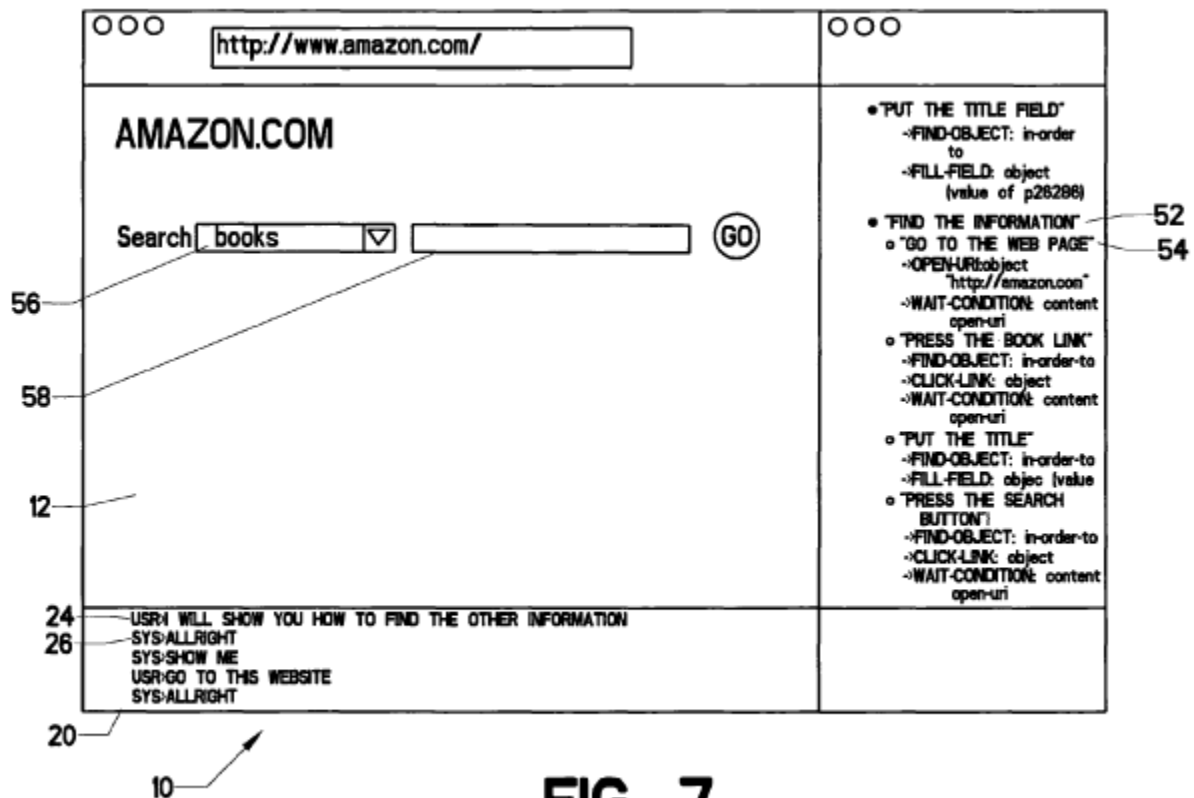


FIG. 7

187. According to the '317 patent, a widget is a GUI element. (EX1001, 10:19-21). Hence, Allen's transmit button 28, input box, drop down box, and input field are widgets. As discussed above, interacting with the transmit button 28 causes the web browser to navigate to a webpage, interacting with the input box and the input field cause the web browser to receive input in an input field, and interacting with a drop down box causes the web browser to display categories in a

search category box 56. Navigating to a webpage, receiving input in an input field, and displaying categories in a search category box are predefined functionalities.

188. The “system [] allows a user to teach a computation device how to perform complex, repetitive tasks that the user usually would perform using the device’s graphical user interface (GUI)” (EX1014, Abstract.) The user demonstrates actions, and the system observes and then “automatically execute[s] the task.” (*Id.*, Abstract, 6:5-6, 6:61-65.) Thus, Allen discloses element 1[h] (“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality”).

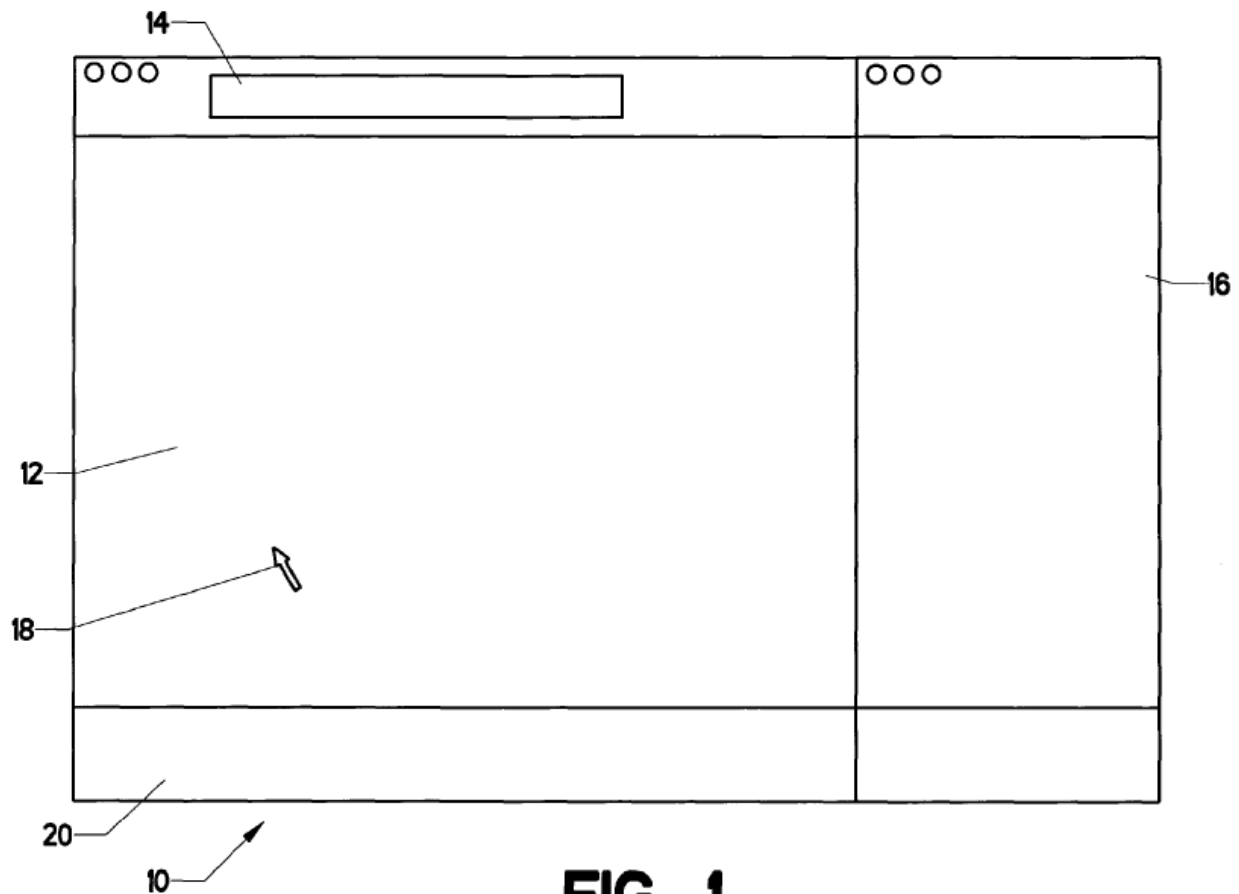
2. Claim 2: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to implement the automation process by a client device used by the user.”

189. As discussed for claim 1, Allen’s system comprises a chat-bot program product. The system implements an automation process of buying a book or searching for a hotel, “via software running on a computational device The system then learns to automatically execute steps by observing user’s actions and analyzing the verbal instructions the user provides.” (EX1014, 6:59-65.) Allen discloses “[t]he system would then include software running on a user’s computer.” (*Id.*, 4:63-64.) The user’s computer is a client device used by a user.

Thus, Allen discloses claim 2 (“wherein the chat-bot program product is configured to implement the automation process by a client device used by the user”).

3. Claim 3: “The chat-bot program product of claim 2, wherein the chat-bot program product is configured to implement the automation process by a browser that is executed by the client device.”

190. As discussed for element 1[b], Allen’s system implements an automation process using “software running on a user’s computer.” (EX1014, 4:63-64.) A user’s computer is a client device. Fig. 1 below is a computer screen of a browser that is executed by the computational device. Thus, Allen discloses claim 3 (“wherein the chat-bot program product is configured to implement the automation process by a browser that is executed by the client device”).



4. **Claim 4: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to converse with the user in natural language using a vocal interface.”**

191. Allen discloses the “user’s computer is equipped with a microphone to receive the user’s speech. PLOW use[s] a language recognition software module to decipher the sounds created by the user and transform them into text . . . However, speech recognition is preferred for its convenience.” (EX1014,

5:24-31.) The system conversing with the user via a microphone means a vocal interface is used. The user utters statements, such as “[e]nter the title field,” “[g]o to this web page,” “I want to buy a book,” etc. (*Id.*, 6:29-30, 8:5, 12:64-65.) The system “PLOW’s responses are presented as spoken words, using a voice synthesizer.” (*Id.*, 5:43-44.) Thus, Allen discloses claim 4 (“wherein the chat-bot program product is configured to converse with the user in natural language using a vocal interface”).

5. Claim 5

- a. 5[a]: “The chat-bot program product of claim 1, wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field,”**

192. As discussed for element 1[g], Allen’s automation process is configured to automatically fill in a text input field in a GUI of a web-based target system.

193. Fig. 3 (below) shows the GUI 10 including an online form (“book purchase form”) comprising text input fields. (EX1014, 6:27.) When the user is teaching the system, “[t]he user utters the statement ‘Enter the title field,’ which appears as deciphered text in speech interface display 20. PLOW then responds by issuing the voice-synthesized statement ‘All right. Show me.’ The user moves

movable cursor 18 to title entry box 30, while uttering the instruction ‘Enter the title here.’ . . . The user then types the desired book title (‘The Glass Castle’) into title entry box 30.” (*Id.*, 6:30-38.) The system then “automatically execute[s] the task.” (*Id.*, Abstract.) Thus, Allen discloses element 5[a] (‘wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field”).

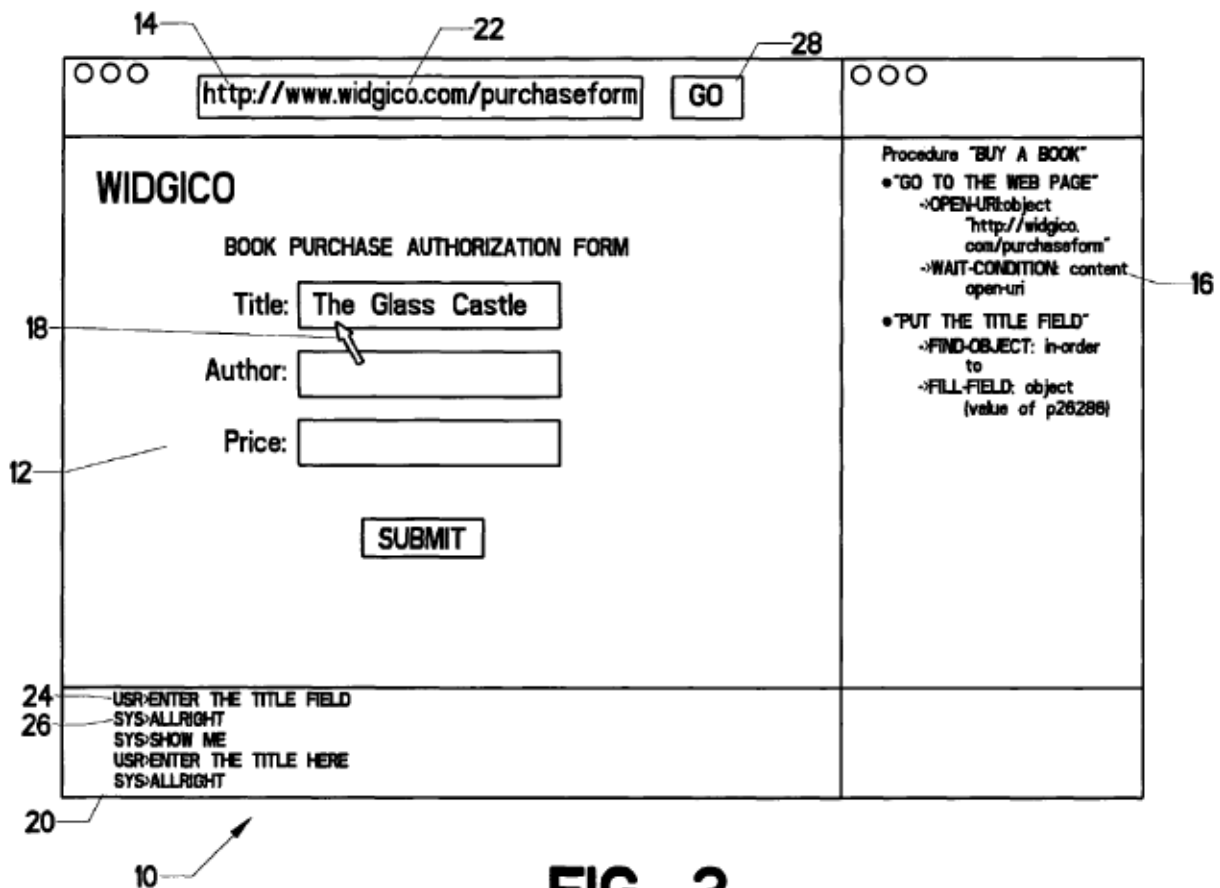


FIG. 3

b. 5[b]: “wherein the widget is configured to invoke submission of the online form,”

194. As shown for element 1[h], Allen’s system interacts with widgets in a GUI 10. Fig. 3 (below) shows the GUI 10 for a book purchase authorization form on the website www.widgico.com (online form). The GUI 10 comprises a Submit GUI button. Clicking on the Submit GUI button would submit the book purchase authorization form, which means it is configured to invoke submission of the online form. Thus, Allen discloses element 5[b].

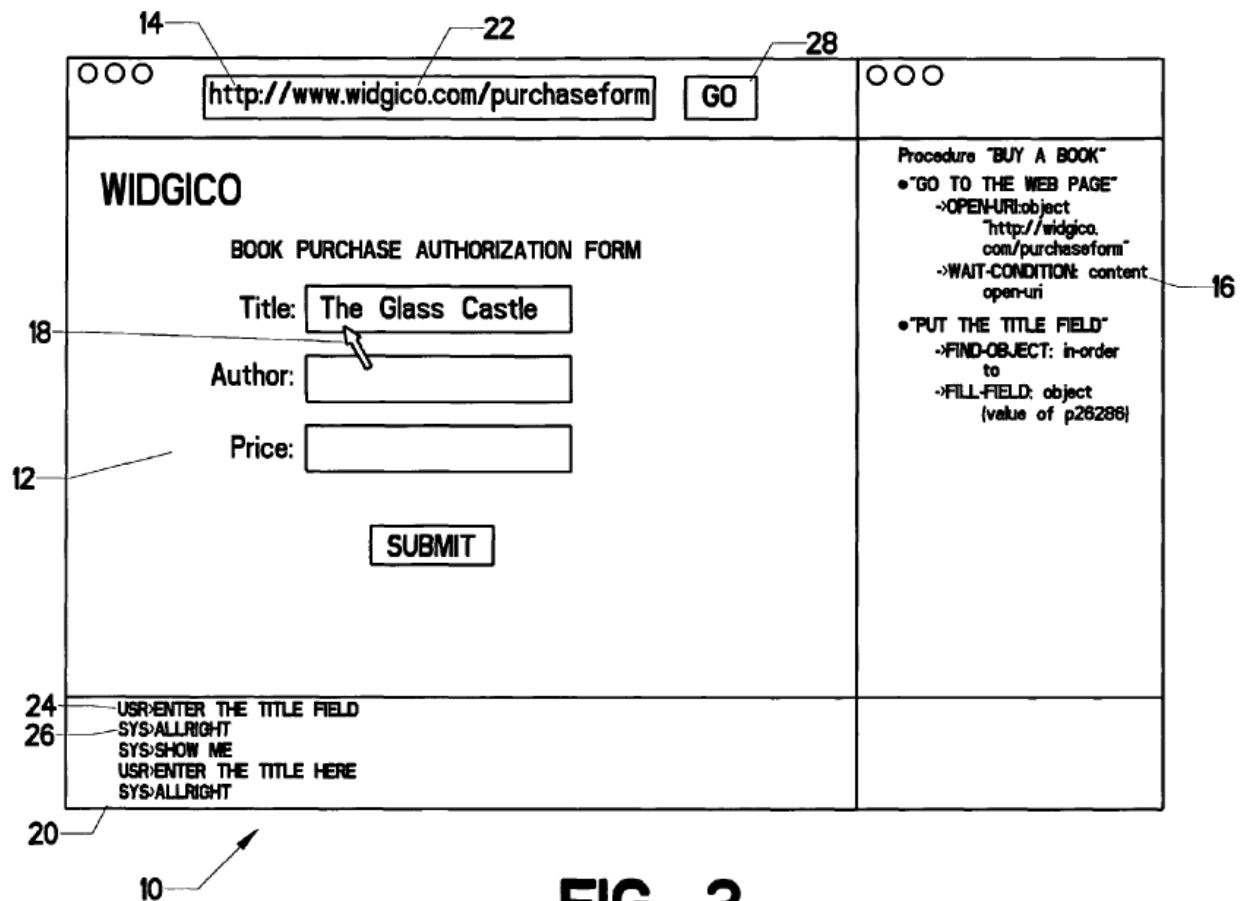


FIG. 3

- c. 5[c]: “wherein the chat-bot program product is configured to request user validation from the user before submitting the online form.”

195. For its book buying task, Allen discloses “[o]nce [all the] information is provided, [the system] PLOW would then perform the steps it was previously taught and complete the transaction. Of course, the user can build in ‘authorization’ steps. As an example, the user could teach the system to obtain price and delivery date information, then provide this information back to the user

and ask whether the user wishes to complete the transaction.” (EX1014, 12:66-13:5.) The system asking whether the user wishes to complete the transaction means the chat-bot program product is configured to request user validation from the user. It would have been obvious that the system would ask for this user validation before completing the transaction (submitting the online form) because the user’s response is needed to determine the system’s next step (whether to submit the online form and buy the book, or not). Thus, Allen renders obvious element 5[c] (“wherein the chat-bot program product is configured to request user validation from the user before submitting the online form”).

6. Claim 6: “The chat-bot program product of claim 1, wherein the information comprises information of the user, whereby personalizing the automation process.”

196. In Allen’s book buying task, the system asks the user for the title of the book that the user wants to buy. (EX1014, 12:65-66.) The title of the book is specific to the user and his/her request, and thus, the information collected by the system comprises information of the user, personalizing the automation process.

197. Allen also discloses “the user can build in ‘authorization’ steps. As an example, the user could teach the system to obtain price and delivery date information[.]” (*Id.*, 13:1-5.) It would have been obvious that in order to

determine the delivery date, the system would need to know the destination address for the delivery. The user's home or work address, information of the user that personalizes the automation process, is an example destination address for delivering the book to the user.

198. Allen also discloses a hotel searching task, where the "user searches for hotels which are near the address '1018 E. De Soto St. Pensacola, Fla. 32501-3362.'" (*Id.*, 16:35-38.) The address would be information of the user, such as the address where the user is going to for a vacation, whereby personalizing the automation process because the automation process is specific to the user. Thus, Allen discloses claim 6 ("wherein the information comprises information of the user, whereby personalizing the automation process").

7. Claim 7: "The chat-bot program product of claim 1, wherein the chat-bot program product is configured to utilize credentials of the user, whereby the chat-bot is privy to user-specific information."

199. As discussed for claim 6, Allen's system utilizes the user's book title and destination address for delivery for implementing the book buying task. (EX1014, 12:65-13:5.)

200. According to the '317 patent, "the user may associate parameters available to the chat bot, such as based on the user credentials." (EX1001,

20:14-16.) The '317 patent provides example user credentials: Employee ID, First Name, and Last Name. (*Id.*, 20:17-27.)

201. A POSITA would understand that there are other user credentials, such as the user's home or work address. The user's name and home or work address are credentials of the user that are provided to the system (e.g., needed to obtain delivery date information and to deliver the book), so the system is privy to the user-specific information. Thus, Allen discloses claim 7.

8. Claim 8: “The chat-bot program product of claim 1, wherein implementation of the automation process is performed by automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI.”

202. As shown for element 1[h], Allen's system automatically interacts with a widget in the GUI. The user “presses transmit button 28, which causes the web browser to ‘navigate’ to the URI” or “select[s] the input box” (clicks a GUI button using a mouse), “moves a computer mouse so that movable cursor 18 moves up to URI box 14” “mov[es] the mouse to a text input box” (moves a mouse cursor), (pulls down a drop down box to select a category in a search category box 56), and “typ[es] in a title” (types into an input field). The user

demonstrates actions, and the system observes and then “automatically execute[s] the task.” (EX1014, Abstract, 6:5-6, 6:61-65.)

203. Since the system observes the user identifying a widget in the GUI and identifying a text input field in a GUI and then automatically executes what it observes, the system implements the automation process by automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI. Thus, Allen discloses claim 8.

9. Claim 9

a. Preamble: “A method comprising:”

204. Allen discloses the preamble. Allen discloses an “interactive method for learning and executing executable tasks using [natural] language and demonstration inputs from a user.” (*Id.*, Claim 1.)

b. 9[a]: “receiving a user instruction from a user using a natural language interface of a chat-bot;”

205. Element 9[a] is similar to element 1[a], but requires the natural language interface be “of a chat-bot.” The table below highlights the differences.

Element 1[a]	Element 9[a]
“receive a user instruction from a user using a natural language interface.”	“receiving a user instruction from a user using a natural language interface <i>of a chat-bot</i> ”

206. As shown for the preamble of claim 1, Allen discloses a chat-bot.

Allen discloses element 1[a], and thus, element 9[a].

- c. **9[b]: “determining to implement an automation process on a web-based target system in response to the user instruction;”**

207. Element 9[b] mirrors element 1[b]. The table below shows the similarities. Allen discloses element 1[b], and thus, element 9[b].

Element 1[b]	Element 9[b]
“determine to implement an automation process on a web-based target system in response to the user instruction;”	“determining to implement an automation process on a web-based target system in response to the user instruction;”

- d. **9[c]: “collecting information to be used as part of the automation process; and”**

208. Element 9[c] mirrors element 1[c]. The table below shows the similarities. Allen discloses element 1[c], and thus, element 9[c].

Element 1[c]	Element 9[c]
“collect information to be used as part of the automation process; and”	“collecting information to be used as part of the automation process; and;”

- e. **9[d]: “implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”**

209. Element 9[d] mirrors element 1[d]. The table below shows the similarities. Allen renders obvious element 1[d], and thus, element 9[d].

Element 1[d]	Element 9[d]
“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”	“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

f. 9[e]: “wherein the automation process is performed with respect to third-party web-based target system and”

210. Element 9[e] mirrors element 1[e]. The table below shows the similarities. Allen discloses element 1[e], and thus, element 9[e].

Element 1[e]	Element 9[e]
“wherein the automation process is performed with respect to third-party web-based target system and”	“wherein the automation process is performed with respect to third-party web-based target system and”

g. 9[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

211. Element 9[f] mirrors element 1[f]. The table below shows the similarities. Allen discloses element 1[f], and thus, element 9[f].

Element 1[f]	Element 9[f]
“without relying on an Application Programming Interface (API) exposed by the web-based target system,”	“without relying on an Application Programming Interface (API) exposed by the web-based target system,”

- h. 9[g]: “wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”**

212. Element 9[g] mirrors element 1[g]. The table below shows the similarities. Allen discloses element 1[g], and thus, element 9[g].

Element 1[g]	Element 9[g]
“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”	“wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

- i. 9[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

213. Element 9[h] mirrors element 1[h]. The table below shows the similarities. Allen discloses element 1[h], and thus, element 9[h].

Element 1[h]	Element 9[h]
“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”	“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

10. Claim 10: “The method of claim 9, wherein said implementing is performed by a client device used by the user.”

214. Claim 10 recites limitations similar to claim 2, but does not require a “chat-bot program product [be] configured to” implement the automation process. The table below highlights the differences. Allen discloses claim 2, and thus, claim 10.

Claim 2	Claim 10
“wherein <i>the chat-bot program product is configured to implement the automation process</i> by a client device used by the user.”	“wherein <i>said implementing is performed</i> by a client device used by the user.”

11. Claim 11: “The method of claim 10, wherein said implementing is performed by a browser that is executed by the client device.”

215. Claim 11 recites limitations similar to claim 3, but does not require a “chat-bot program product [be] configured to” implement the automation process.

The table below highlights the differences. Allen discloses claim 3, and thus, claim 11.

Claim 3	Claim 11
“wherein <i>the chat-bot program product is configured to implement the automation process</i> by a browser that is executed by the client device.”	“wherein <i>said implementing is performed</i> by a browser that is executed by the client device.”

12. Claim 12: “The method of claim 9, wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

216. Claim 12 recites limitations similar to claim 4, but requires a “chat-bot” instead of a “chat-bot program product” be configured to converse with the user in natural language using a vocal interface. A chat-bot program product is a chat-bot. Allen discloses claim 4, and thus, claim 12.

Claim 4	Claim 12
“wherein the chat-bot <i>program product</i> is configured to converse with the user in natural language using a vocal interface.”	“wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

13. Claim 13

- a. 13[a]: “The method of claim 9, wherein the automation process is configured to automatically fill information in an online form in the GUI, wherein the online form comprises the text input field,”**

217. Element 13[a] recites limitations similar to element 5[a], but requires filling information in an online form “in the GUI.” The table below highlights the differences. As shown for element 5[a], Allen’s online form is in a GUI. Thus, Allen discloses element 13[a].

Element 5[a]	Element 13[a]
“wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field,”	“wherein the automation process is configured to automatically fill information in an online form <i>in the GUI</i> , wherein the online form comprises the text input field,”

b. 13[b]: “wherein the widget is configured to invoke submission of the online form,”

218. Element 13[b] mirrors element 5[b]. The table below shows the similarities. Allen discloses element 5[b], and thus, element 13[b].

Element 5[b]	Element 13[b]
“wherein the widget is configured to invoke submission of the online form,”	“wherein the widget is configured to invoke submission of the online form,”

c. 13[c]: “wherein the method further comprises requesting user validation from the user before submitting the online form.”

219. Element 13[c] mirrors element 5[c], but does not require the “chat-bot program product [be] configured” to request user validation. The table below

highlights the differences. Allen renders obvious element 5[c], and thus, element 13[c].

Element 5[c]	Element 13[c]
“wherein <i>the chat-bot program product is configured to</i> request user validation from the user before submitting the online form.”	“wherein the method further comprises requesting user validation from the user before submitting the online form.”

14. Claim 14: “The method of claim 9, wherein the information comprises information of the user, whereby personalizing the automation process.”

220. Claim 14 mirrors claim 6. The table below shows the similarities.

Allen discloses claim 6, and thus, claim 14.

Claim 6	Claim 14
“wherein the information comprises information of the user, whereby personalizing the automation process.”	“wherein the information comprises information of the user, whereby personalizing the automation process.”

15. Claim 15: “The method of claim 9, wherein said implementing the automation process comprises: automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”

221. Claim 15 recites limitations similar to claim 8. The table below shows the similarities. Allen discloses claim 8, and thus, claim 15.

Claim 8	Claim 15
“wherein implementation of the automation process <i>is performed by</i> automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI.”	“wherein said implementing the automation process <i>comprises</i> : automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”

16. Claim 16

- a. **Preamble: “A computerized apparatus comprising a processor and coupled memory, the processor being adapted to perform:”**

222. Allen discloses the preamble. Allen’s “system includes software running on a user’s computational device.” (EX1014, Abstract.) A computational device has a processor and coupled memory. A computational device is a computerized apparatus.

- b. **16[a]: “receiving a user instruction from a user using a natural language interface of a chat-bot;”**

223. Element 16[a] is similar to element 1[a], but requires the natural language interface be “of a chat-bot.” The table below highlights the differences.

Element 1[a]	Element 16[a]
“receive a user instruction from a user using a natural language interface.”	“receiving a user instruction from a user using a natural language interface <i>of a chat-bot</i> ”

224. As shown for the preamble of claim 1, Allen discloses a chat-bot.

Allen discloses element 1[a], and thus, element 16[a].

c. 16[b]: “determining to implement an automation process on a web-based target system in response to the user instruction;”

225. Element 16[b] mirrors element 1[b]. The table below shows the similarities. Allen discloses element 1[b], and thus, element 16[b].

Element 1[b]	Element 16[b]
“determine to implement an automation process on a web-based target system in response to the user instruction;”	“determining to implement an automation process on a web-based target system in response to the user instruction;”

d. 16[c]: “collecting information to be used as part of the automation process; and”

226. Element 16[c] mirrors element 1[c]. The table below shows the similarities. Allen discloses element 1[c], and thus, element 16[c].

Element 1[c]	Element 16[c]
“collect information to be used as part of the automation process; and”	“collecting information to be used as part of the automation process; and;”

e. 16[d]: “implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

227. Element 16[d] mirrors element 1[d]. The table below shows the similarities. Allen renders obvious element 1[d], and thus, element 16[d].

Element 1[d]	Element 16[d]
“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”	“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

f. 16[e]: “wherein the automation process is performed with respect to third-party web-based target system and”

228. Element 16[e] mirrors element 1[e]. The table below shows the similarities. Allen discloses element 1[e], and thus, element 16[e].

Element 1[e]	Element 16[e]
“wherein the automation process is performed with respect to third-party web-based target system and”	“wherein the automation process is performed with respect to third-party web-based target system and”

g. 16[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

229. Element 16[f] mirrors element 1[f]. The table below shows the similarities. Allen discloses element 1[f], and thus, element 16[f].

Element 1[f]	Element 16[f]
“without relying on an Application Programming Interface (API) exposed by the web-based target system,”	“without relying on an Application Programming Interface (API) exposed by the web-based target system,”

- h. 16[g]: “wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”**

230. Element 16[g] mirrors element 1[g]. The table below shows the similarities. Allen discloses element 1[g], and thus, element 16[g].

Element 1[g]	Element 16[g]
“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”	“wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

- i. 16[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

231. Element 16[h] mirrors element 1[h]. The table below shows the similarities. Allen discloses element 1[h], and thus, element 16[h].

Element 1[h]	Element 16[h]
“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”	“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

- 17. Claim 17: “The computerized apparatus of claim 16, wherein the computerized apparatus is a client device used**

by the user, wherein said implementing is performed by a browser that is executed by the client device.”

232. A portion of claim 17 recites limitations similar to claim 2 and claim 3, but requires a “computerized apparatus” be a client device and be configured to implement the automation process. The table below shows the differences.

Claim 2	Claim 3	Claim 17
“wherein <i>the chat-bot program</i> product is configured to implement the automation process by a client device used by the user.”	“wherein <i>the chat-bot program product</i> is configured to implement the automation process <i>by a browser that is executed</i> by the client device.”	“wherein the <i>computerized apparatus</i> is a client device used by the user, wherein said implementing is <i>performed by a browser that is executed</i> by the client device.”

233. A chat-bot program product is a computerized apparatus. Allen discloses claim 2 and claim 3, and thus, claim 17.

18. Claim 18: “The computerized apparatus of claim 16, wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

234. Claim 18 recites limitations similar to claim 4, but requires a “chat-bot” instead of a “chat-bot program product” be configured to converse with the user in natural language using a vocal interface. A chat-bot program product is a chat-bot. Allen discloses claim 4, and thus, claim 18.

Claim 4	Claim 18
“wherein the chat-bot <i>program product</i> is configured to converse with the user in natural language using a vocal interface.”	“wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

19. Claim 19

- a. 19[a]: “The computerized apparatus of claim 16, wherein the automation process is configured to automatically fill information in an online form in the GUI, wherein the online form comprises the text input field,”

235. Element 19[a] recites limitations similar to element 5[a], but requires filling information in an online form “in the GUI.” The table below highlights the differences. As shown for element 5[a], Allen’s online form is in a GUI. Thus, Allen discloses element 19[a].

Element 5[a]	Element 19[a]
“wherein the automation process is configured to automatically fill information in an online form, wherein the online form comprises the text input field,”	“wherein the automation process is configured to automatically fill information in an online form <i>in the GUI</i> , wherein the online form comprises the text input field,”

- b. 19[b]: “wherein the widget is configured to invoke submission of the online form,”

236. Element 19[b] mirrors element 5[b]. The table below shows the similarities. Allen discloses element 5[b], and thus, element 19[b].

Element 5[b]	Element 19[b]
“wherein the widget is configured to invoke submission of the online form,”	“wherein the widget is configured to invoke submission of the online form,”

- c. **19[c]: “wherein the method further comprises requesting user validation from the user before submitting the online form.”**

237. Element 19[c] is similar to element 5[c], but does not require the “chat-bot program product [be] configured” to request user validation. The table below highlights the differences. Allen renders obvious element 5[c], and thus, element 19[c].

Element 5[c]	Element 19[c]
“wherein <i>the chat-bot program product is configured to</i> request user validation from the user before submitting the online form.”	“wherein the method further comprises requesting user validation from the user before submitting the online form.”

20. **Claim 20: “The computerized apparatus of claim 16, wherein said implementing the automation process comprises: automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”**

238. Claim 20 recites limitations similar to claim 8. The table below shows the similarities. Allen discloses claim 8, and thus, claim 20.

Claim 8	Claim 20
“wherein implementation of the automation process <i>is performed by</i> automatically identifying the widget in the GUI and by automatically identifying the text input field in the GUI.”	“wherein said implementing the automation process <i>comprises</i> : automatically identifying the widget in the GUI; and automatically identifying the text input field in the GUI.”

C. Claims 1-4, 6, 7, 9-12, 14, and 16-18 are Obvious Over Krishnan and Radhakrishnan

21. Summary of Krishnan

239. Krishnan was filed on January 30, 2018, and claims priority to a provisional application filed on November 3, 2017. Table 1 below provides citations to Krishnan’s provisional application for claim 1 of Krishnan, and Table 2 below provides citations to Krishnan’s provisional application (EX1017) for specific subject matter relied upon as prior art. Table 1 and Table 2 show that Krishnan’s patent (EX1016) is prior art by November 3, 2017, the filing date of the Krishnan provisional application. The Examiner cited, but never characterized or discussed Krishnan during prosecution.

Table 1	
Krishnan’s Claim 1	Krishnan’s Provisional Application (EX1017)
A computer-implemented method, comprising:	[0001] (“A chat bot may be a computer program that conducts a conversation via auditor and/or textual methods.”);

Table 1	
Krishnan's Claim 1	Krishnan's Provisional Application (EX1017)
	<i>see also</i> [0045]-[0058].
receiving, via a user interface of a bot-builder development environment, a selected dialog including one or more interactions between a chat bot and a recipient;	[0003] (“The user may request information or functionality to be performed. The chat bot may receive, interpret, and respond to the request by performing an action, or requesting additional information from the user. The back and forth interaction between a user and chat bot may be referred to as a conversation. A particular conversational thread directed at achieving a particular outcome (acquiring information from a user (such as validating a user), or performing a user-requested or action (such as providing an order status or processing refund), may be referred to as a long. An example dialog of a conversation is shown in Figure 1.”)], [0009] (“The welcome message the bot provides to the user is shown in the message section. The user may respond to the message by saying they need help, in which case an administrator, using the provided chatbot builder user interface (UI) may determine how the bot would respond to the message.”); [0011] (“Using the botbuilder user interface, an administrator may select a call dialog action, in which case the system may populate the dialog name with a drop down or list of previously created dialogs, or enable the

Table 1	
Krishnan's Claim 1	Krishnan's Provisional Application (EX1017)
	administrator to name and create a new dialog. The system may then enable the user to design the interaction for the new dialog.”), Fig. 1; <i>see also</i> [0002]-[0008], [0010], [0012]-[0015], Figs. 2, 3.
receiving, via a trigger area of the user interface, a plurality of phrases corresponding to triggering an execution of the selected dialog;	[0009] (“The welcome message the bot provides to the user is shown in the message section. The user may respond to the message by saying they need help, in which case an administrator, using the provided chatbot builder user interface (UI) may determine how the bot would respond to the message. Example bot responses include waiting for user input, showing a menu of options, or calling another dialog.”); [0029] (“The bot builder UI may train the dialog with different keywords/phrases to make the flow better and the particular dialog smarter and responsive to a broader or narrower range of inputs/users. For example, in the figure below, the dialog can be configured with multiple different triggers by which the dialog may invoke, including: checking my order status, where is my order, it’s been too long, where is my food, etc.”), Fig. 7; <i>see also</i> [0010], [0011]-[0015], [0030].
configuring the selected dialog wherein upon a receipt of one of the plurality of phrases, the chat bot	[0024] (“As can be seen on the main menu dialog of Fig. 5, the admin can provide the user a selection of different

Table 1	
Krishnan's Claim 1	Krishnan's Provisional Application (EX1017)
executes the selected dialog;	menu options. . . the user may type in text, key words, questions, or commands, which may be evaluated and the corresponding dialog may be loaded and executed by the bot. The bot builder UI may enable the administrator to visually configure how the bot responds to different user input, by performing different actions, requesting information, or calling various dialogs"); <i>see also</i> [0004], [0019].
receiving a threshold indicating a percentage of words for one of the plurality of phrases that trigger the execution within the bot-builder development environment;	[0030] ("the bot builder UI may enable a user to set a matching threshold by which a system would not need to find all the words to trigger the dialog, but a particular percentage may be good enough. Or for example, phrases such as 'Where order,' or 'were ordeeer' may be close enough to trigger the dialog based on the customer intent.").
receiving the one of the plurality of phrases via a preview area of the user interface of the bot-builder development environment that is configured to simultaneously display the preview area with the trigger area within the bot-builder development environment, wherein the preview area displays a simulated interaction between the chat bot and a recipient; and	[0021] ("The bot builder UI enables an administrator to visually create the conversational flow from the bot's point of view and have a real-time immediate preview of the flow from the user's perspective."), [0032] ("The bot builder UI enables an admin to design and preview in real-time a customer interaction with a bot, using different dialogs with the ability to visually call or reference different dialogs based on user input or responses to prompts or menus."), Fig. 6; <i>see also</i> [0020],

Table 1	
Krishnan's Claim 1	Krishnan's Provisional Application (EX1017)
	[0032].
executing functionality associated with the selected dialog responsive to the received one of the plurality of phrases based on a determination that the received one of the plurality of phrases exceeds the threshold.	[0030] (“the bot builder UI may enable a user to set a matching threshold by which a system would not need to find all the words to trigger the dialog, but a particular percentage may be good enough. Or for example, phrases such as ‘Where order,’ or ‘were ordeeer’ may be close enough to trigger the dialog based on the customer intent.”), [0032] (“a dialog may include a function or module that runs when a particular keyword or menu item is detected or triggered. A dialog may include responses, including actions that are performed by the bot. . . . Dialogs may include a backend actions and do not have to be user facing.”); <i>see also</i> [0003], [0004], [0014], [0024], [0028].

Table 2	
Specific Subject Matter in Krishnan (EX1016)	Krishnan's Provisional Application (EX1017)
A visual chatbot builder that executes the selected dialog, conduct a conversation with a user, and is designed to simulate a human conversational pattern.	[0001], [0003], [0015], [0018], [0019], [0021], [0024]
A chat bot executing on a computing device and configured to simulate a conversation a user via text or auditory methods.	[0001], [0045]

Table 2	
Specific Subject Matter in Krishnan (EX1016)	Krishnan's Provisional Application (EX1017)
A chat bot that provides customer service to users via the Internet or a website to fulfill customer service requests, configured to interact and interface with customers or consumers who are looking for answers to frequently asked questions. Some of these questions may include “how do I get a refund?” “what is my order status?” “how do I cancel my order?” “What is your return policy?” “What is the cheapest flight to Europe?”	[0002], [0003], [0032], [0053], [0054]
A chat bot that serves as an interface between the customer and one or more backend databases or other information processing systems.	[0002]
A chat bot that responds to customer input by retrieving and providing the requested information, requesting additional input or information from the user, or performing an action and providing the user with status updates about the progress of the action. This back-and-forth interaction between a user and chat bot may be referred to as a conversation.	[0003]
A chat bot that executes a welcome dialog/basic green such as “Hello, how may I help you today?” From there, depending on how the user answers, different dialogs, programs, or other sub-routines may be called by the bot to interact with user, or the	[0006], [0025], [0009], [0011], Fig. 9

Table 2	
Specific Subject Matter in Krishnan (EX1016)	Krishnan's Provisional Application (EX1017)
chatbot may execute a function.	
The user requests that the chatbot book a flight ("Book Flight" 110). The chatbot responds by acknowledging the customer's request to book a flight ("Ok, let's book a flight"), along with requesting additional information ("Where would you like to fly to?"). The user selects an option for the destination. The chatbot requests information necessary to book the flight and processes the request.	[0003], [0004], Fig. 1
Through executing dialogs, a chatbot can provide the user with information retrieved from another system, request information from the user (including user verification), and perform actions to fulfill user requests (such as account transfers, refunds, processing payments, booking tickets, etc.).	[0010], [0012], [0014], [0026]
A bot-builder interface that enables the developer to visually configure how the bot responds to different user input, by performing different actions, requesting information, or calling various dialogs.	[0024]
A natural language processing engine	[0030]
A processor of the computer system, which implements the natural language interface. The processors are also called central processing units or CPUs.	[0046], [0048]

Table 2	
Specific Subject Matter in Krishnan (EX1016)	Krishnan's Provisional Application (EX1017)
A computer system including main or primary memory, such as random access memory (RAM).	[0049]
A tangible, non-transitory computer usable or readable medium having control logic (software) stored thereon may also be referred to herein as a computer program product or program storage device. This includes, but is not limited to, a computer system, main memory, secondary memory, and removable storage units, as well as tangible articles of manufacture embodying any combination of the foregoing. Such control logic, when executed by one or more data processing devices (such as a computer system), may cause such data processing devices to operate as described herein.	[0057]

240. Krishnan discloses a “visual [chat] bot builder,” where “the chat bot executes the selected dialog.” (EX1016, Abstract.) The chatbot interfaces with customers via a conversation: “[t]he chat bot may respond to customer input by retrieving and providing the requested information, requesting additional input or information from the user, or performing an action and providing the user with status updates about the progress of the action.” (*Id.*, 2:40-44.) Fig. 1 (annotated

below) shows an example interface and conversation between the chatbot and a customer.

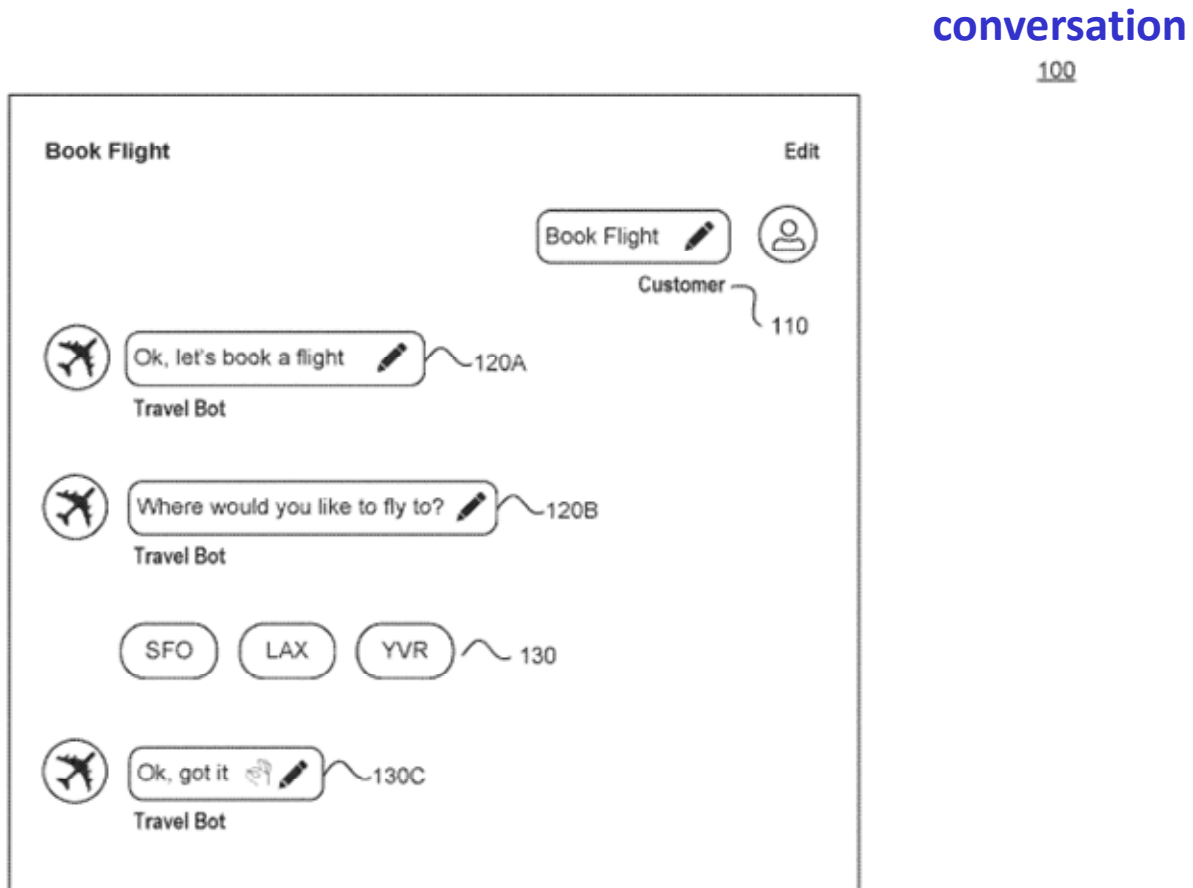


FIG. 1

22. Summary of Radhakrishnan

241. Radhakrishnan was filed on October 30, 2017. The Examiner never found, and Applicant never cited, Radhakrishnan during prosecution.

Radhakrishnan discloses a “robotic process automation enabled file diagnosis and

correction.” (EX1018, Abstract.) The RPA module (also referred to as an RPA bot) “simulate[] user operations with the user device to generate an operation record for automatically generating and transmitting a diagnosis and correction request to a workstation.” (*Id.*) The operation record is generated and transmitted when a computer software-related error occurs (*e.g.*, “program crash, file corruption, or the like”) (Fig. 2 (below)). (*Id.*) The operation record is used for reproducing, diagnosing, and rectifying the error. (*Id.*, 1:7-19.)

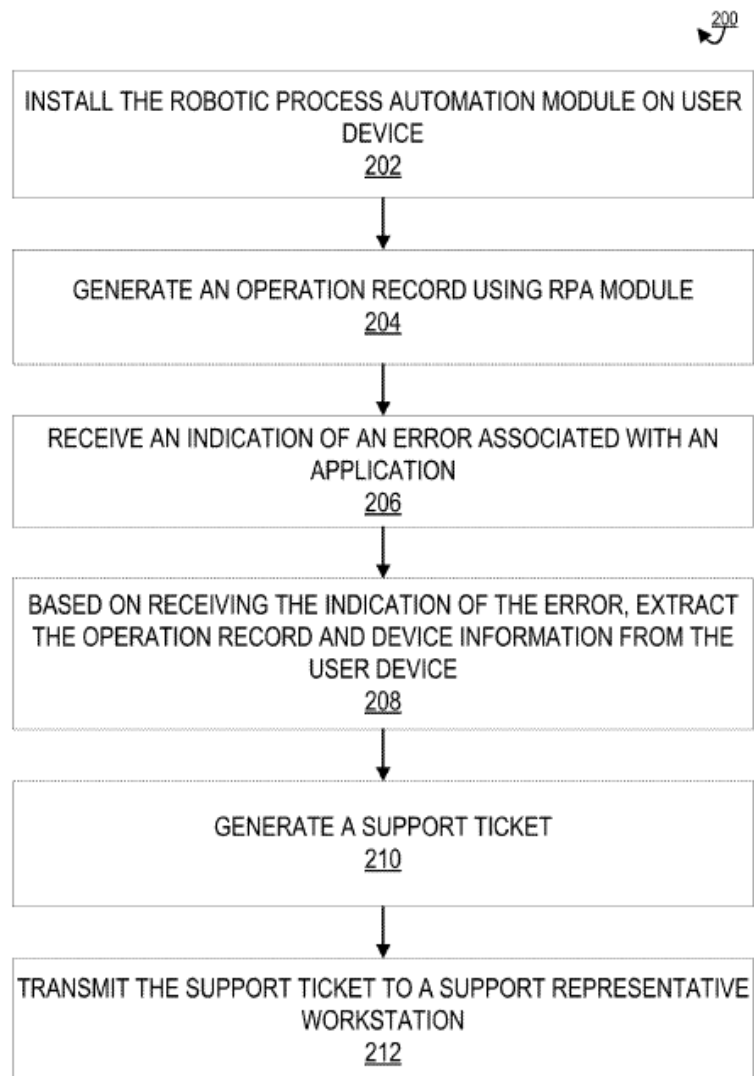


FIG. 2

23. Claim 1

- a. Preamble: “A chat-bot program product, wherein the chat-bot program product comprises non-transitory computer readable medium retaining program**

instructions, which program instructions when read by a processor, cause the processor to:”

242. Krishnan-Radhakrishnan renders obvious the preamble. Krishnan discloses “a chat bot [] be configured to interact and interface with customers or consumers who are looking for answers to frequently asked questions.” (EX1016, 2:27-29.) Krishnan also discloses a “tangible, non-transitory computer usable or readable medium having control logic (software) stored thereon may also be referred to herein as a computer program product or program storage device. This includes, but is not limited to, computer system 1100, main memory 1108, secondary memory 1110, and removable storage units 1118 and 1122, as well as tangible articles of manufacture embodying any combination of the foregoing. Such control logic, when executed by one or more data processing devices (such as a computer system 1100), may cause such data processing devices to operate as described herein.” (*Id.*, 12:43-55.)

243. Radhakrishnan discloses “[a]ny suitable transitory or non-transitory computer readable medium may be utilized. . . . a computer readable medium may be any medium that can contain, store, communicate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device.” (EX1018, 13:44-64.)

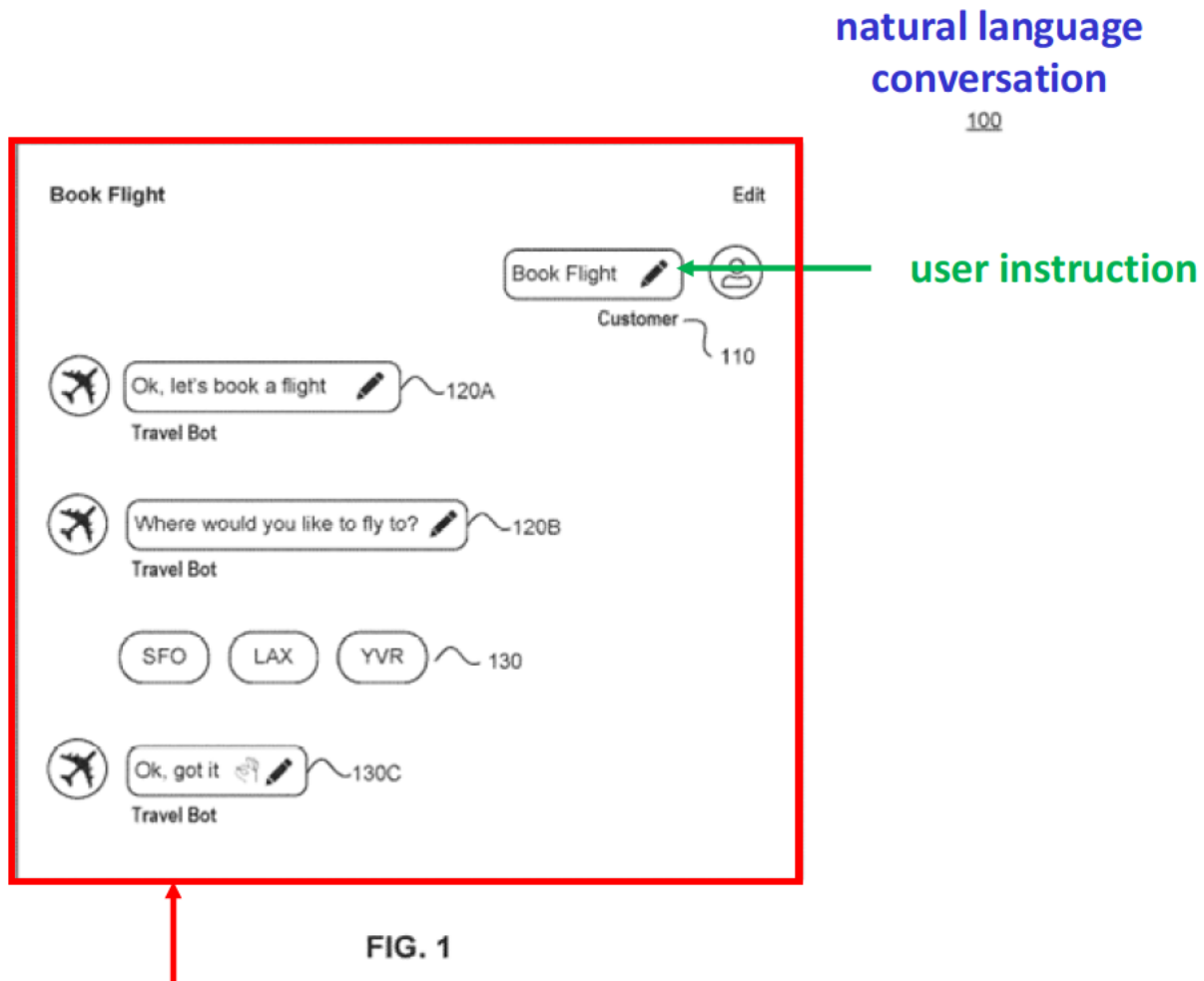
b. 1[a]: “receive a user instruction from a user using a natural language interface;”

244. According to the ’317 patent, “[a] Chat bot is a computer program implementing natural language interface to conduct a conversation via auditory or textual methods with the user.” (EX1001, 1:28-30.)

245. Krishnan discloses “a chat bot [] configured to interact and interface with customers or consumers who are looking for answers to frequently asked questions. Some of these questions may include ‘how do I get a refund?’ ‘what is my order status?’ ‘how do I cancel my order?’ ‘What is your return policy?’ ‘What is the cheapest flight to Europe?’” (EX1016, 2:27-33.) Hence, Krishnan’s chat bot “uses a natural language interface.”

246. “An interaction with a bot may begin with the bot executing a welcome dialog that includes a basic greeting from the bot such as ‘Hello, how may I help you today?’” (*Id.*, 2:56-58.) The user provides a user instruction to book a flight (“Book Flight” 110), using the natural language interface shown in Fig. 1 (annotated below)). (*Id.*, 2:62.) “The chat bot may respond to customer input by retrieving and providing the requested information, requesting additional input or information from the user, or performing an action and providing the user with status updates about the progress of the action. This back-and-forth

interaction between a user and chat bot may be referred to as a conversation.” (*Id.*, 2:40-46.)



247. The chatbot may “continue executing the book flight dialog until the user books a flight or cancels the request.” (*Id.*, 3:7-9.) Krishnan’s chatbot is

configured with certain functions that should be executed as part of the dialog. (*Id.*, 5:59-61.) “Example functions include sending a response 420A, such as presenting text or information to the user. The function may include asking a question to request information from a user 420B. Asking a question may include displaying a text box, radio-button, or drop-down selection area which the user may use to respond to the question.” (*Id.*, 5:62-67.) The questions from the chatbot and responses from the user resemble language spoken by a human, and hence, are part of a natural language conversation between the natural language interface and the user. Thus, Krishnan discloses element 1[a] (“receive a user instruction from a user using a natural language interface”).

c. 1[b]: “determine to implement an automation process on a web-based target system in response to the user instruction;”

248. As discussed for element 1[a], in Krishnan’s flight booking task, the user’s instruction is to book a flight (“Book Flight” 110).

249. As Krishnan explains, “[t]hrough executing dialogs, a chatbot can provide the user with information retrieved from another system, request information from the user, and perform actions to fulfill user requests (such as account transfers, refunds, processing payments, booking tickets, etc.).” (EX1016,

3:46-50.) Hence, Krishnan discloses determining to implement an automation process (booking a flight) in response to the user instruction.

250. Krishnan discloses implementing an automation process on web-based target system. “[A] user may visit or log into a website and have customer service requests fulfilled by a chat bot.” (*Id.*, 2:10-12.) Thus, Krishnan discloses “determine to implement an automation process on a web-based target system in response to the user instruction.”

d. 1[c]: “collect information to be used as part of the automation process; and”

251. In Krishnan’s flight booking task, the chat bot asks, “Where would you like to fly to?” prompting the user by “requesting additional information 120B.” (EX1016, 2:65-66, Fig. 1.) The chat bot [] requests information “necessary to book the flight, such as locations, dates of travel, seat preferences, and credit card information[.]” (*Id.*, 2:66-3:3.)

252. After the chatbot prompts the user and the user provides the information necessary to book the flight, the values are then used to “process[] the request to book the flight.” (*Id.*, 3:2-3.) As discussed for element 1[b], the flight booking task is an automation process. Thus, Krishnan discloses element 1[c] (“collect information to be used as part of the automation process”).

- e. **1[d]: “implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”**

253. Krishnan discloses “[t]hrough executing dialogs, a chatbot can provide the user with information retrieved from another system, request information from the user, and perform actions to fulfill user requests (such as account transfers, refunds, processing payments, booking tickets, etc.).” (EX1016, 3:46-50.) In the flight booking task, the chatbot requests information from the user (the destination) and performs actions to fulfill the user request (book a flight). Hence, Krishnan discloses implementing the automation process.

254. Krishnan also discloses a GUI of a web-based target system. “[A] user may visit or log into a website and have customer service requests fulfilled by a chat bot.” (*Id.*, 2:10-12.)

255. Krishnan is silent regarding whether or not its automation process interacts with a GUI and simulates user input to the GUI.

256. Radhakrishnan discloses “leverag[ing] the unique interactions of a robotic process automation module with a user interface of an application to simulate user operations with the user device to generate an operation record for automatically generating and transmitting a diagnosis and correction request to a

workstation.” (EX1018, Abstract.) The RPA “[b]ots are trained to mimic user activity directly with a UI” of the device. (*Id.*, 5:49-50.) Hence, Radhakrishnan discloses an automation process that interacts with a GUI. “[T]he simulated operations generated by the RPA module may act as a mirror of the actual user operations and be used as an accurate, complete operation record which details the user operations in their entirety.” (*Id.*, 11:32-36.) Radhakrishnan’s RPA bots simulate user input to the GUI.

257. It would have been obvious to combine Krishnan and Radhakrishnan to a POSITA at the time of the ’317 priority date. Krishnan discloses using a chatbot to “provide customer service to users.” (EX1016, 2:5-11.) Radhakrishnan discloses an RPA bot that helps diagnose computer software-related errors, such as “program crashes, file corruption, and operating errors” that are typical when using “computer systems . . . to complete various tasks and increase user and/or process efficiency.” (EX1018, 1:7-12.) Both Krishnan and Radhakrishnan use bots for the advantages that they offer: “greater convenience for users and lower costs for an organization (relative to having a live person interact with the user and handle the requests” and “reduce[d] [] amount of computing resources required to execute

tasks when compared with executing the same tasks using a traditional application programming interface (API).” (EX1016, 2:5-11; EX1018, 6:21-34.)

258. Krishnan teaches the “chat bot may serve as an interface between the customer and one or more backend databases or other information processing systems.” (EX1016, 2:33-35.) Radhakrishnan teaches “RPA modules or ‘bots’ are used throughout various industries to automatically execute one or more processes typically on the back-end of a computing system or device.” (EX1018, 5:21-23.) The RPA bots automatically execute processes, such as a back-end process, at the presentation layer of the computing system (without an API). (*Id.*, 5:49-52.)

259. Krishnan provides a customer service example for its chatbot: “a user may visit or log into a website and have customer service requests fulfilled by a chat bot.” (EX1016, 2:10-12.) A POSITA would have been motivated to fulfill these customer service requests by an RPA bot, particularly when the customer service request involves more complicated tasks, since “bots also have the capability, based on their configuration, to perform more complex tasks” (EX1018, 5:32-35.) Diagnosing and rectifying computer software-related errors is a complicated task that deals with “errors [that] can be extremely technical in nature requiring expert knowledge or specialized tools outside the realm of a

typical user in order to repair.” (*Id.*, 1:9-15.) Computer systems and device errors are complex, and proper correction requires a high level of technical detail. (*Id.*, 1:15-18.) Radhakrishnan teaches RPA bots that automatically and efficiently diagnoses and corrects process failures, program crashes, bugs, glitches, file corruption, missing files, and operating errors, thereby lowering operational costs, and increasing user and/or process efficiency. (*Id.*, 1:9-36, 11:65-67.) For customer service requests involving user input into a GUI, the Krishnan-Radhakrishnan combination renders obvious element 1[d] (“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system”).

f. 1[e]: “wherein the automation process is performed with respect to third-party web-based target system and”

260. Krishnan discloses “[a]n example use of a chat bot may be to provide customer service to users via the Internet or a mobile application. . . . For example, a user may visit or log into a website and have customer service requests fulfilled by a chat bot.” (EX1016, 2:5-11.)

261. It would have been obvious that the Internet or a website (for customer service requests) is a third-party web-based target system. Krishnan discloses different types of customer service requests being fulfilled such as booking a flight or “validating a user, making purchases, requesting status updates, processing refunds, or modifying information” (among others). (*Id.*, 2:22-24.) These different types of customer services may be for web-based systems that belong to different parties. For example, booking a flight may be a customer service request that involves an airline company as one party, and making purchases may be a customer service request that involves an online merchant as another party. The airline company and/or the online merchant are likely (and typically) not the same party, so at least one would have a third-party web-based target system. Thus, Krishnan renders obvious element 1[e] (“wherein the automation process is performed with respect to third-party web-based target system”).

g. 1[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”

262. Radhakrishnan discloses “[t]he RPA module may typically interact[] with a presentation layer of the computer system (i.e., the UI) and avoids more

demanding interaction with any underlying data layers or logic of the underlying system in order to instruct operation of the user device, computer system, application, or the like.” (EX1018, 5:10-15.) Radhakrishnan provides an explanation for why a POSITA would want to execute the automation process without utilizing an API, because it “avoid[s] the costly and time consuming process of building a new and complicated API-based system[] from the ground up.” (*Id.*, 6:25-29.) “By leveraging robotic process automation technology, the present invention reduces the amount of computing resources required to execute tasks when compared with executing the same tasks using a traditional application programming interface (API).” (*Id.*, 6:21-25.) Hence, Radhakrishnan discloses “without relying on an Application Programming Interface (API).”

263. As discussed for element 1[b], Krishnan discloses implementing the automation process on a web-based target system. The RPA module interacting with a presentation layer of the computer system means there is no exposure by a web-based system. As discussed for element 1[d], it would have been obvious to combine Krishnan and Radhakrishnan. Thus, Krishnan-Radhakrishnan renders obvious element 1[f].

- h. 1[g]: “wherein the implementation of the automatic process comprises: automatically and without user**

input, filling in a text input field in the GUI of the web-based target system; and”

264. In Radhakrishnan, the user’s operations include “selecting options (e.g., a check box, drop-down menu), completing a field” (EX1018, 11:17-19.) “[T]he simulated operations generated by the RPA module may act as a mirror of the actual user” (*Id.*, 11:29-36.) Completing a field means filling in a text input field in the GUI. Since the user’s operations comprise completing a field (filling in a text input field in the GUI), and the RPA module mirrors the user’s operations, then the RPA module’s interactions comprise automatically and without user input, filling in a text input field in the GUI. (*Id.*, 7:12-17.)

265. As discussed for element 1[b], Krishnan discloses implementing the automation process on a web-based target system. As discussed for element 1[d], it would have been obvious to combine Krishnan and Radhakrishnan. Thus, Krishnan-Radhakrishnan renders obvious element 1[g].

i. 1[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

266. In Radhakrishnan, the user’s operations include “selecting options (e.g., a check box, drop-down menu).” (EX1018, 11:17-19.) “[T]he simulated operations generated by the RPA module may act as a mirror of the actual user

...” (*Id.*, 11:29-36.) Selecting options such as a check box, drop-down menu, etc. means interacting with a widget in the GUI to cause the system to perform a predefined functionality. Since the user’s operations comprise selecting options such as a check box, drop-down menu, etc., and the RPA module mirrors the user’s operations, then the RPA module’s interactions comprise automatically and without user input, interacting with a widget in the GUI. (*Id.*, 7:12-17.)

267. As discussed for element 1[b], Krishnan discloses implementing the automation process on a web-based target system. As discussed for element 1[d], it would have been obvious to combine Krishnan and Radhakrishnan. Thus, Krishnan-Radhakrishnan renders obvious element 1[h].

24. Claim 2: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to implement the automation process by a client device used by the user.”

268. Krishnan discloses the “bot-builder interface, described herein, may be a computing device” (EX1016, 3:23-24.) A computing device is a client device used by the user. Thus, Krishnan discloses claim 2 (“wherein the chat-bot program product is configured to implement the automation process by a client device used by the user”).

25. Claim 3: “The chat-bot program product of claim 2, wherein the chat-bot program product is configured to implement the automation process by a browser that is executed by the client device.”

269. As discussed for claim 1, Krishnan-Radhakrishnan discloses an automation process. As discussed for claim 2, Krishnan discloses implementing the automation process by a client device.

270. Krishnan discloses “[a]n example use of a chat bot may be to provide customer service to users via the Internet or a mobile application. . . . For example, a user may visit or log into a website and have customer service requests fulfilled by a chat bot.” (EX1016, 2:5-11.) The Internet is accessed by a browser. Thus, Krishnan discloses claim 3 (“wherein the chat-bot program product is configured to implement the automation process by a browser that is executed by the client device”).

26. Claim 4: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to converse with the user in natural language using a vocal interface.”

271. Krishnan discloses “a chat bot executing on a computing device (hereinafter simply ‘chat bot’ or ‘bot’) may be configured to simulate a conversation a user may otherwise have with a human user via text or auditory methods.” (EX1016, 2:1-4.) Conversing via auditory methods involves

conversing a vocal interface. The conversation includes the user providing a user instruction to book a flight (“Book Flight” 110) in natural language, for example. (*Id.*, 2:62, Fig. 1.) Thus, Krishnan discloses claim 4 (“wherein the chat-bot program product is configured to converse with the user in natural language using a vocal interface”).

27. Claim 6: “The chat-bot program product of claim 1, wherein the information comprises information of the user, whereby personalizing the automation process.”

272. Krishnan discloses a user verification dialog, where “the user verification dialog may ask the user for particular information: email address, password, username, address, answers to security questions, pin number, order number, ticket number, or other information by which the system or bot can verify the user’s identity and/or retrieve information about the user’s account.” (EX1016, 4:41-46.) As Krishnan explains, “[v]erification may be required to perform or process various requests including providing the user account balance information, performing transfer requests, or purchasing items.” (*Id.*, 4:46-49.) Hence, when the automation process involves, *e.g.*, user-specific information, such as providing the user account balance information, performing transfer requests, or purchasing items, then Krishnan discloses collecting information of the user and performing

the automation with the user-specific information, whereby personalizing the automation process. Thus, Krishnan discloses claim 6.

28. Claim 7: “The chat-bot program product of claim 1, wherein the chat-bot program product is configured to utilize credentials of the user, whereby the chat-bot is privy to user-specific information.”

273. As discussed for claim 6, Krishnan discloses certain automation processes that involve user-specific information. (EX1016, 4:46-49.)

274. According to the ’317 patent, “the user may associate parameters available to the chat bot, such as based on the user credentials” (EX1001, 20:14-16.) The ’317 patent provides example user credentials: Employee ID, First Name, and Last Name. (*Id.*, 20:17-27.)

275. “[T]he user verification dialog may ask the user for particular information: email address, password, username, address, answers to security questions, pin number, order number, ticket number, or other information by which the system or bot can verify the user’s identity and/or retrieve information about the user’s account.” (EX1016, 4:41-46.) Email address, password, username, address, and other information for verifying the user’s identity and about the user’s account are credentials of the user that are provided to the chat-bot, whereby the chat-bot is privy to user-specific information. Thus, Krishnan discloses claim 7.

29. Claim 9

a. Preamble: “A method comprising:”

276. Krishnan discloses the preamble. Krishnan discloses a method of “conduct[ing] a conversation with a user and is designed to simulate a human conversational pattern.” (EX1016, 1:17-19.)

b. 9[a]: “receiving a user instruction from a user using a natural language interface of a chat-bot;”

277. Element 9[a] is similar to element 1[a], but requires the natural language interface be “of a chat-bot.” The table below highlights the differences.

Element 1[a]	Element 9[a]
“receive a user instruction from a user using a natural language interface.”	“receiving a user instruction from a user using a natural language interface <i>of a chat-bot</i> ”

278. As shown for the preamble of claim 1, Krishnan discloses a chat-bot. Krishnan discloses element 1[a], and thus, element 9[a].

c. 9[b]: “determining to implement an automation process on a web-based target system in response to the user instruction;”

279. Element 9[b] mirrors element 1[b]. The table below shows the similarities. Krishnan discloses element 1[b], and thus, element 9[b].

Element 1[b]	Element 9[b]
“determine to implement an automation process on a web-based target system in response to the user instruction;”	“determining to implement an automation process on a web-based target system in response to the user instruction;”

d. 9[c]: “collecting information to be used as part of the automation process; and”

280. Element 9[c] mirrors element 1[c]. The table below shows the similarities. Krishnan discloses element 1[c], and thus, element 9[c].

Element 1[c]	Element 9[c]
“collect information to be used as part of the automation process; and”	“collecting information to be used as part of the automation process; and;”

e. 9[d]: “implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

281. Element 9[d] mirrors element 1[d]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[d], and thus, element 9[d].

Element 1[d]	Element 9[d]
“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”	“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

- f. **9[e]: “wherein the automation process is performed with respect to third-party web-based target system and”**

282. Element 9[e] mirrors element 1[e]. The table below shows the similarities. Krishnan renders obvious element 1[e], and thus, element 9[e].

Element 1[e]	Element 9[e]
“wherein the automation process is performed with respect to third-party web-based target system and”	“wherein the automation process is performed with respect to third-party web-based target system and”

- g. **9[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”**

283. Element 9[f] mirrors element 1[f]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[f], and thus, element 9[f].

Element 1[f]	Element 9[f]
“without relying on an Application Programming Interface (API) exposed by the web-based target system,”	“without relying on an Application Programming Interface (API) exposed by the web-based target system,”

- h. **9[g]: “wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”**

284. Element 9[g] mirrors element 1[g]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[g], and thus, element 9[g].

Element 1[g]	Element 9[g]
“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”	“wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

- i. **9[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

285. Element 9[h] mirrors element 1[h]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[h], and thus, element 9[h].

Element 1[h]	Element 9[h]
“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”	“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

30. **Claim 10: “The method of claim 9, wherein said implementing is performed by a client device used by the user.”**

286. Claim 10 recites limitations similar to claim 2, but does not require a “chat-bot program product [be] configured to” implement the automation process. The table below highlights the differences. Krishnan discloses claim 2, and thus, claim 10.

Claim 2	Claim 10
“wherein <i>the chat-bot program product is configured to implement the automation process</i> by a client device used by the user.”	“wherein <i>said implementing is performed</i> by a client device used by the user.”

31. Claim 11: “The method of claim 10, wherein said implementing is performed by a browser that is executed by the client device.”

287. Claim 11 recites limitations similar to claim 3, but does not require a “chat-bot program product [be] configured to” implement the automation process. The table below highlights the differences. Krishnan discloses claim 3, and thus, claim 11.

Claim 3	Claim 11
“wherein <i>the chat-bot program product is configured to implement the automation process</i> by a browser that is executed by the client device.”	“wherein <i>said implementing is performed</i> by a browser that is executed by the client device.”

32. Claim 12: “The method of claim 9, wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

288. Claim 12 recites limitations similar to claim 4, but requires a “chat-bot” instead of a “chat-bot program product” be configured to converse with the user in natural language using a vocal interface. A chat-bot program product is a chat-bot. Krishnan discloses claim 4, and thus, claim 12.

Claim 4	Claim 12
“wherein the chat-bot <i>program product</i> is configured to converse with the user in natural language using a vocal interface.”	“wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

33. Claim 14: “The method of claim 9, wherein the information comprises information of the user, whereby personalizing the automation process.”

289. Claim 14 mirrors claim 6. The table below shows the similarities. Krishnan discloses claim 6, and thus, claim 14.

Claim 6	Claim 14
“wherein the information comprises information of the user, whereby personalizing the automation process.”	“wherein the information comprises information of the user, whereby personalizing the automation process.”

34. Claim 16

a. Preamble: “A computerized apparatus comprising a processor and coupled memory, the processor being adapted to perform:”

290. Krishnan-Radhakrishnan render obvious the preamble. Krishnan discloses “[c]omputer system 1100 may include one or more processors (also called central processing units, or CPUs), such as a processor 1104.” (EX1016, 11:13-15.) “Computer system 1100 may also include a main or primary memory 1108, such as random access memory (RAM).” (*Id.*, 11:29-33.) The computer system including both the processor and the memory means the memory is coupled memory.

291. Radhakrishnan discloses its user device being “any device that employs a processor and memory and can perform computing functions.” (EX1018, 3:36-43.)

292. As shown for element 1[d], it would have been obvious to combine Krishnan and Radhakrishnan.

b. 16[a]: “receiving a user instruction from a user using a natural language interface of a chat-bot;”

293. Element 16[a] is similar to element 1[a], but requires the natural language interface be “of a chat-bot.” The table below highlights the differences.

Element 1[a]	Element 16[a]
“receive a user instruction from a user using a natural language interface.”	“receiving a user instruction from a user using a natural language interface <i>of a chat-bot</i> ”

294. As shown for the preamble of claim 1, Krishnan discloses a chat-bot. Krishnan discloses element 1[a], and thus, element 16[a].

c. 16[b]: “determining to implement an automation process on a web-based target system in response to the user instruction;”

295. Element 16[b] mirrors element 1[b]. The table below shows the similarities. Krishnan discloses element 1[b], and thus, element 16[b].

Element 1[b]	Element 16[b]
“determine to implement an automation process on a web-based target system in response to the user instruction;”	“determining to implement an automation process on a web-based target system in response to the user instruction;”

d. 16[c]: “collecting information to be used as part of the automation process; and”

296. Element 16[c] mirrors element 1[c]. The table below shows the similarities. Krishnan discloses element 1[c], and thus, element 16[c].

Element 1[c]	Element 16[c]
“collect information to be used as part of the automation process; and”	“collecting information to be used as part of the automation process; and;”

- e. **16[d]: “implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”**

297. Element 16[d] mirrors element 1[d]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[d], and thus, element 16[d].

Element 1[d]	Element 16[d]
“implement the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”	“implementing the automation process by interacting with a Graphical User Interface (GUI) of the web-based target system and simulating user input to the GUI of the web-based target system,”

- f. **16[e]: “wherein the automation process is performed with respect to third-party web-based target system and”**

298. Element 16[e] mirrors element 1[e]. The table below shows the similarities. Krishnan renders obvious element 1[e], and thus, element 16[e].

Element 1[e]	Element 16[e]
“wherein the automation process is performed with respect to third-party web-based target system and”	“wherein the automation process is performed with respect to third-party web-based target system and”

- g. **16[f]: “without relying on an Application Programming Interface (API) exposed by the web-based target system,”**

299. Element 16[f] mirrors element 1[f]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[f], and thus, element 16[f].

Element 1[f]	Element 16[f]
“without relying on an Application Programming Interface (API) exposed by the web-based target system,”	“without relying on an Application Programming Interface (API) exposed by the web-based target system,”

- h. 16[g]: “wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”**

300. Element 16[g] mirrors element 1[g]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[g], and thus, element 16[g].

Element 1[g]	Element 16[g]
“wherein the implementation of the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”	“wherein said implementing the automatic process comprises: automatically and without user input, filling in a text input field in the GUI of the web-based target system; and”

- i. 16[h]: “automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”**

301. Element 16[h] mirrors element 1[h]. The table below shows the similarities. Krishnan-Radhakrishnan renders obvious element 1[h], and thus, element 16[h].

Element 1[h]	Element 16[h]
“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”	“automatically and without user input, interacting with a widget in the GUI to cause the web-based target system to perform a predefined functionality.”

35. Claim 17: “The computerized apparatus of claim 16, wherein the computerized apparatus is a client device used by the user, wherein said implementing is performed by a browser that is executed by the client device.”

302. A portion of claim 17 recites limitations similar to claim 2 and claim 3, but requires a “computerized apparatus” be a client device and be configured to implement the automation process. The table below shows the differences.

Claim 2	Claim 3	Claim 17
“wherein <i>the chat-bot program</i> product is configured to implement the automation process by a client device used by the user.”	“wherein <i>the chat-bot program product</i> is configured to implement the automation process <i>by a browser that is executed by</i> the client device.”	“wherein the <i>computerized apparatus</i> is a client device used by the user, wherein said implementing is <i>performed by a browser that is executed by</i> the client device.”

303. A chat-bot program product is a computerized apparatus. Krishnan discloses claim 2 and claim 3, and thus, claim 17.

36. Claim 18: “The computerized apparatus of claim 16, wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

304. Claim 18 recites limitations similar to claim 4, but requires a “chat-bot” instead of a “chat-bot program product” be configured to converse with the user in natural language using a vocal interface. A chat-bot program product is a chat-bot. Krishnan discloses claim 4, and thus, claim 18.

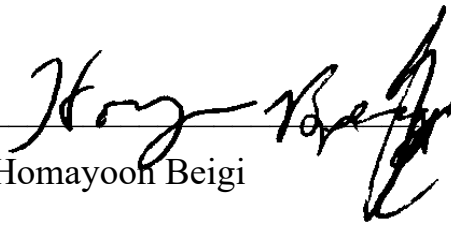
Claim 4	Claim 18
“wherein the chat-bot <i>program product</i> is configured to converse with the user in natural language using a vocal interface.”	“wherein the chat-bot is configured to converse with the user in natural language using a vocal interface.”

* * *

I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code.

Expert Declaration of Dr. Homayoon Beigi
Inter Partes Review of USP 11,558,317

Dated: October 18, 2023


Homayoon Beigi

Homayoon Beigi

*49 Bouton Road
South Salem, NY 10590
Beigi@RecoTechnologies.com
<http://www.RecoTechnologies.com/beigi>
+1-914-997-5676 Ext. 201*

Table of Contents

*Red and Blue text elements are hyperlinks
(Red: local link Blue: external link)*

- | | | |
|----------------------------|-------------------------------|---------------------------|
| 1. Education | 9. Publications | 10. Products |
| 2. Employment | Books | Product Libraries |
| 3. Computer Skills | Book Chapters & Encyclopedias | 11. Research Projects |
| 4. Awards and Honors | Patents | 12. Standards Development |
| 5. Professional Activities | Journal Publications | |
| 6. Review Committees | Invited Tutorials | |
| 7. Society Membership | Conference Publications | |
| 8. Miscellaneous | Keynote Speeches | |
| | Invited Talks and Radio | |
| | Technical Reports | |

Education

Columbia University, New York, NY
Feb 1991 Doctor of Engineering Science
Defended 9/1990 Major: [Mechanical Engineering](#)
Thesis: [Neural Network Learning and Learning Control Through Optimization Techniques](#)

May 1985 Master of Science
Major: [Mechanical Engineering](#)
[Kinematics, Dynamics, and Control Systems](#)

May 1984 Bachelor of Science
Major: [Mechanical Engineering](#)
Machine Design

[Return to Table of Contents](#)

Employment

Recognition Technologies, Inc.
Jul. 2003-Present *Chief Executive Officer and President*
Duties:
[Research and Development](#) of Human Biometric systems and Human Language Technologies.
Architecture design and implementation of the research results into practical systems including engines and applications.
Consulting:
Expert evaluations on legal and forensic cases involving speaker identities and speech related topics for many legal entities and patent agencies.
[Products:](#)
Exclusive design and implementation of several recognition engines, fully developed in-house (containing over 1.5 million lines of original C++ code, written personally over the past 18 years – 12 years of research and development on code which was from 3 years of research at [Internet Server Connections, Inc.](#) and 3 years at [Applied Mathematics Research](#))
[RecoMadeEasy® Speaker Recognition Engine](#)
Winner of the [2011 Frost & Sullivan North American Speaker Verification Biometrics, New Product Innovation Award](#)
[RecoMadeEasy® Large Vocabulary Speech Recognition Engine](#)
[RecoMadeEasy® Face Recognition Engine](#)
[RecoMadeEasy® Audiovisual Recognition Engine](#)
[RecoMadeEasy® Automatic Language Proficiency Rating Engine](#)
[RecoMadeEasy® Interactive Voice Response \(IVR\)](#)

RecoMadeEasy® Signature Compression and Verification Engine
RecoMadeEasy® Online Handwriting Recognition Engine
RecoMadeEasy® Keystroke Recognition Engine

Department of Computer Science, Columbia University

Jan. 2012-Present *Adjunct Professor*

Multidisciplinary Graduate Courses:

COMS-E6998 (Fundamentals of Speech Recognition)

Overall Instructor Evaluations by Students:

2022 Spring : 5.0/5.0
2021 Spring - Fall : 4.5/5.0 - 5.0/5.0
2020 Spring - Fall : 4.1/5.0 - 5.0/5.0
2019: 5.0/5.0
2018: 4.0/5.0
2017: 4.8/5.0
2016: 4.9/5.0
2015: 4.6/5.0
2014: 4.13/5.0

COMS-W4995 (Mathematics of Machine Learning and Signal Recognition)

COMS-E6998 (Fundamentals of Speaker Recognition)

Overall Instructor Evaluations by Students:

2014: 4.9/5.0
2013: 4.6/5.0
2012: 5.0/5.0

Department of Mechanical Engineering, Columbia University

Jan. 2003-Present *Adjunct Professor*

Multidisciplinary course:

MECE-E6620 (Applied Signal Recognition)

2020 Spring : 5.0/5.0
2019 Spring : 4.0/5.0

Cross-listed course across mechanical and electrical engineering:

EEME-E4601 (Digital Control Systems)

Overall Instructor Evaluations by Students:

2019 Spring : 4.0/5.0

Department of Electrical Engineering, Columbia University

1995 and 1996 *Adjunct Assistant Professor*

EE-E6820 (Speech and Handwriting Recognition)

Internet Server Connections, Inc.

Feb. 2000-Present *Vice President and Chief Technology Officer*

Complete technical operation of [Internet Server Connections](#).

Duties include:

Research and development related to customized products and services for customized hosting and networking clients.

Design and implementation of complex networks using [Cisco®](#) (as a certified [Cisco®](#) vendor) and customized Linux routers using [BGP4](#) and other IP routing, load and redundancy optimization.

Network and IP resource management, telephony network design and implementation based on Dialogic T1/E1 telephony cards and the [RecoMadeEasy®](#) IVR engine.

Projects include:

[Portfolio Optimization algorithm](#) development for [Merrill-Lynch Research](#) (Real-time, constrained non-linear optimization problem of over 38,000 variables)

The [complete creation, cross-referencing, indexing, specialized search \(specialized Unicode alphabet\) and management](#) of the digital libraries related to the [Encyclopædia Iranica](#), a thirty five year old scholarship endeavor at the [Center for Iranian Studies](#) at [Columbia University](#)

Projects with over 180 other Corporate clients in telecommunications, including customized in-house design, implementation, and hosting services in our dedicated data center with customers, some of which receive an average of over a million hits per day.

Feb. 1991-

[IBM T.J. Watson Research Center](#)

Feb. 2001

Research Staff Member

Feb. 1997-

[Human Language Technologies](#)

Research in the fields of Speaker Recognition (Verification and Identification) and Speech Recognition and Automatic Segmentation. Developed algorithms for the IBM Research Speaker Verification and Identification Engine which is now being used in an array of different products while conducting research for improving its performance. This system is completely Text Independent and is in addition Language Independent.

Feb. 1991-

[Pen Technologies](#)

Feb. 1997

Principal Investigator and Project Leader for [Pen-Based Music Editor](#). Research in on-line Handwriting Recognition: Developed and implemented Handwriting Recognition Algorithms in every area related to the topic. These areas include: Research and development of the Front-End, Search, and Language Models for Cursive and Unconstrained Recognition systems as well as Discrete and Run-on versions. System architect for IBM Unconstrained Recognition system.

Developed a model of handwriting and a model of human hand-dynamics for application in recognition of Cursive and Natural writing. Combined the above models in a Hidden Markov Model (HMM) framework. Developed a set of normalization, desloping and deslanting schemes for application to cursive handwriting recognition. Conducted research and

developed on-line training capabilities for the IBM cursive recognizer.

Research in optimal clustering techniques and Neural Network Models of Handwriting. Responsible for a major part of search and language models in the IBM product ThinkWriteTM.

Department of Mechanical Engineering, Columbia University

Jan. 1997-
Dec. 2003

Adjunct Associate Professor

Cross-listed course between mechanical and electrical engineering:
EEME-E4601 (Digital Control Systems)

Department of Electrical Engineering, Columbia University

Sep. 1995-
Dec. 1996

Adjunct Assistant Professor

Graduate course EE-E6820 (Speech and Handwriting Recognition)

The first session was also made available through the [Columbia Video Network](#), with an additional 10 students remotely connected through satellite connection.

Applied Mathematics Research, Sole Proprietorship

Jul. 1993-
Jul. 1996

Software Development

Wrote the first Matlab[®] to C++ translator. Wrote all internal matrix functions of Matlab[®] in C++. Wrote a Matlab[®] parser and a C++ code generator to produce C++ code, translating any Matlab[®] code to 100% C++ code. This project entailed optimal implementation of complex numerical matrix manipulation functions. The code libraries were later inherited by [Internet Server Connections, Inc.](#)

Center for Telecommunication Research, Columbia University

Oct 1990-
Feb 1991

Research Specialist

Research in digital image coding, image data compression and transmission. Developed and implemented schemes for lossless image compression and transmission, and ultra-fast methods of image display through low-level hardware control.

Department of Mechanical Engineering, Columbia University

Dec 1987-
Sep 1990

Research Assistant

Doctoral Research in the field of Learning-Adaptive Control and Neural Network Learning – [Research abstract](#) enclosed.
Advisers: Prof. C. James Li and Prof. Richard W. Longman.

Jun 1989-
Dec 1990

Research Assistant

Fault detection of mechanical systems and machine health prognosis, funded by the U.S. Navy and supervised by Prof. C. James Li. Developed practical signal processing techniques for the health prognosis of mechanical components such as bearings, gears, cutting tools, etc. This project included the design and implementation of the sensors and the data acquisition apparatus, as well extensive pattern recognition algorithm design and implementation for the automatic detection of faults in different components.

Sep 1988- May 1989	<i>Teaching Assistant</i> Instruction of Mechanical Engineering Laboratory (E3018, E3028, E3038).
Jan 1986- Sep 1986	<i>Research Assistant</i> Worked with the late Prof. Ferdinand Freudenstein on a generalization theory on Kinematic Analysis of commonly used linkages Publication: (Freudenstein and Beigi 1986).
Sep 1985- Jan 1986	<i>Departmental Research Assistant and Teaching Assistant</i> Laboratory Consultant for Computer Aided Design (MECE E3408). Teaching assistance in the lectures and laboratory instruction for a graduate level course entitled Introduction to Robotics (MECE-E4602).
Sep 1984- May 1985	<i>Research Assistant</i> Digital image processing applied to fluid mechanics for the analysis of lubricants' behavior in zero gravity – An experiment conducted in conjunction with the first NASA Spacelab project – STS-9 . Designed and created the digitization platform using a sonic digitizer and wrote drivers for the digitizer in C, on an IBM PC platform. Digitized frames of 24-fps film taken of the spreading of fluids with different viscosities by the crew of the Columbia Shuttle in the Spacelab module. This data was used by Prof. Coda Pan to formulate the equations that describe the spreading characteristics of fluids on different smooth surfaces in zero gravity.

Computer Skills

Extensive experience in [Linux](#) and other [Unix-like](#) operating systems, [Android](#), [Windows](#), [VM](#), [VMS](#), [OS2](#), and many other operating systems.
In excess of 33 years of experience with many languages such as [C](#), [C++](#), [FORTRAN77](#), [Java](#), [JavaScript](#), [HTML](#), [PHP](#), [XML](#), [ASSEMBLY](#), [LaTeX](#), and several shell scripting languages such as [AWK](#), [tcsh](#) and [csh](#), [ksh](#), [Bourne shell](#), etc.
Network Design, TCP/IP, [BGP4](#), Load Balancing, Telephony Network design, [Cisco®](#) Operating System, [Zebra](#) and [Quagga Routing](#), [GateD routing](#).

[Return to Table of Contents](#)

Awards and Honors

Feb. 2023	<i>Finalist for Columbia University Presidential Awards for Outstanding Teaching</i> One of only 11 finalists for 5 Presidential Awards for Outstanding Teaching across all Columbia University faculty.
Jan. 2021	<i>IMAC Best Paper Award: Dynamics of Civil Structures</i> “ Transfer Learning from Audio Domains a Valuable Tool for Structural Health Monitoring ”

Jan. 2012	<i>2011 Frost & Sullivan North American Speaker Verification Biometrics, New Product Innovation Award</i> – For RecoMadeEasy® Speaker Recognition
2003	<i>IBM Research Top 10% Valuable Patent</i> Patent Number 6,421,645 was recognized by IBM Research as one of top 10% Valuable Patents.
2002	<i>Best of Show Award</i> The Internet World Show, held at the Jacob Javits Center in New York City In the E-Commerce category for the CommerceMadeEasy® software engine
Oct. 25, 2002	<i>The Linux Journal Product of the Day Award</i> For the CommerceMadeEasy® software engine
2001	<i>Third Plateau Invention Award, IBM Research</i> presented to an individual with 12 accepted patent disclosures
April 1999	<i>Adventurous System & Software Research (ASSR) Award, IBM Research</i> 2 nd Extension of the ASSR award for further features of the speaker recognition engine
1999	<i>Second Plateau Invention Award, IBM Research</i> presented to an individual with 8 accepted patent disclosures
April 1998	<i>Adventurous System & Software Research (ASSR) Award, IBM Research</i> Extension of the ASSR award for improving the speaker recognition engine
1998	<i>Research Division Award, IBM Research</i> For the creation of the Virage transcription system
1998	<i>First Plateau Invention Award, IBM Research</i> presented to an individual with 4 accepted patent disclosures
April 1997	<i>Adventurous System & Software Research (ASSR) Award, IBM Research</i> ASSR award for creating a speaker recognition engine for IBM Research
1997	<i>Research Division Award, IBM Research</i> For the success of the Network Vehicle (speech enabled vehicle)
1996	<i>Research Division External Honors Award, IBM Research</i> In Recognition of the honor of being elected as an Associate Editor of the Intelligent Automation and Soft Computing Journal
April 1996	<i>Adventurous System & Software Research (ASSR) Award, IBM Research</i> Extension of the ASSR award for another year to conduct adventurous research in the field of Handwriting Recognition for CMN (Common Music Notation) .

- April 1995 *Adventurous System & Software Research (ASSR) Award, IBM Research*
The largest ASSR award (\$110,000) for research in the field of
[Handwriting Recognition for CMN \(Common Music Notation\)](#).
- 1995 *Extraordinary Ability Status by U.S. Immigration and Naturalization*
Obtained U.S. Permanent Residence through the extremely rare
[Extraordinary Ability category](#) which resulted in obtaining permanent
residence in three weeks after submitting the application.
- 1995 *Research Division External Honors Award, IBM Research*
In Recognition of the honor of being elected as The Conference Chair for
the Conference on Technological Advancement in Developing Countries
- 1995 *Sigma Xi Scientific Research Honor Society*
Elected to the Columbia University (Kappa) Chapter
- March 1994 *IBM Research Division Award*
For “Online Discrete Handwriting Recognition”
- 1993 *IBM Research Patent Award*
For the Invention of “A Post-Processing Error Correction Scheme Using a
Dictionary for On-Line Boxed and Run-On Handwriting Recognition”
- 1990 *IEEE Best Paper Award*
Homayoon Beigi and C. James Li, “[New Neural Network Learning Based on
Gradient-Free Optimization Methods](#),” 1990 IEEE Conf. on Neural Networks
- 1990 *IEEE Best Paper Award*
Homayoon Beigi and C. James Li, “[Neural Network
Learning Based on Quasi-Newton Methods with Initial Scaling of
Inverse Hessian Approximate](#),” 1990 IEEE Conference on Neural Networks
- 1984 Scholarship Award, [Mechanical Engineering Department, Columbia University](#)
Scholarship covering tuition and fees for the last semester of the senior year.
- 1981 *Bausch & Lomb Science Award*
Presented for academic excellence in science, rigor of courses taken in
the sciences, and SAT Math and ACT Science and Math scores.
- 1980 *The Maria Elena Arosemena Cup*
Trophy is given to the international student who has gained the greatest
command and understanding of the English Language
- 1980 *The Bancroft Phinney Award*
Presented to the Student with Highest Academic Standing

[Return to Table of Contents](#)

Professional Activities

- 2012 *VIPG Panel of Experts*
Voted into an 8-person Voice Identification Policy Group (VIPG) of the United States Embassy in Mexico for the registration of the Federal police of Mexico.
- 2009 - 2010 *Interoperability Committee Member*
U.S. Government Interagency Symposium for Investigatory Voice Biometrics
[Federal Bureau of Investigation Interoperability Committee](#)
- Sep. 2009 *Interoperability Panelist*
[U.S. Government Interagency Symposium for Investigatory Biometrics](#)
- 2005 *Invited Keynote Speaker*
“Challenges of Large-Scale Speaker Recognition,” *Keynote Speech* at the European standards, [COST275 Workshop on Biometrics on the Internet](#), October 27-28, 2005, Hertfordshire University, Hatfield, United Kingdom.
- 2003 – 2010 *Active Liaison*
Active liaison for the U.S. Delegation of [ISO/SC 37 JTC 1 WG3 \(Biometrics\)](#) standards development for the Common Biometric Exchange Format.
- 2003 – 2010 *Active Liaison and Driving Force for Speaker Recognition*
[ANSI/INCITS Standards development for Biometric Data Interchange Formats](#).
- 2003 – 2010 *Active Liaison and Driving Force*
[VoiceXML Forum](#) standards development for Speaker Recognition (Speaker Biometrics) Data Format.
- 2002 *Guest Editor*
Special issue on Learning and Repetitive Control, Volume 8, No. 2, 2002
[Intelligent Automation and Soft Computing Journal](#)
- 1998 – 2010 *Chief Digital Editor*
[Encyclopaedia Iranica](#), Published by [Columbia University](#), New York City, NY
Managed, designed, implemented, and maintained the entire digitization and publication in digital form of the encyclopedia.
- 1998 *Organizer*
The Pattern Recognition Division of the [World Automation Congress](#), Alaska
- 1997 and 1999 *Organizer*
The World Manufacturing Congress
- 1997 *Technical Chair*
[The IEEE International Conference on Robotics and Automation](#)

1994-2013	<i>Associate Editor</i> Intelligent Automation and Soft Computing Journal
1994 - 2010	<i>Advisory Board Member</i> The Advisory Board Committee of the IEEE Spectrum Magazine on Technological Advancement in Developing Nations
1997	<i>Invited Keynote Speaker</i> Foundations of Distributed Information Systems (FDIS 97) Conference , Aspen, Colorado, Jun. 1997
1994 - 2003	<i>Editor</i> For Applications of Soft Computing to Handwriting Recognition for the Berkeley Initiative on Soft Computing (BISC)
1995	<i>Conference Chair</i> Third Annual Conference on Technological Advancement in Developing Countries, Columbia University, New York, 1995
1991-1995	<i>Executive Committee Member</i> The Society for Technological Advancement in Developing Countries
1991-1993	<i>Technical Chair</i> Mechanical, Chemical, and Industrial Engineering Division of the Society for Technological Advancement in Developing Countries
Jul 23-24, 1993	<i>Organizing Committee Member</i> The Annual Conference on Technological Advancements in Developing Countries

[Return to Table of Contents](#)

Scientific and Technical Review Committees

[Journal Reviewer Public Profile](#)

2016 - Present	European Intelligence and Security Informatics Conference (EISIC)
2012 - Present	IET Signal Processing Journal
2014 - Present	Journal of Phonetics
2012 - Present	IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2012, 2013, 2014, 2015, 2016, 2018, 2019, 2020, 2021, 2022
2010 - Present	Parsa Wireless Communications Intellectual Property and Patent Evaluation Organization

Patent Evaluation Expert for Speech and Handwriting related patents

- 2009 - Present [Qatar Foundation](#) Grant Review Committee
Review of grant proposals to the Qatar Foundation, including technical correctness, budget review, and feasibility analysis
- 2009 - Present The Interspeech Conference – [2009](#), [2010](#), [2011](#), [2012](#), 2013, [2014](#), [2015](#), [2016](#), [2017](#), [2018](#), [2019](#), [2020](#), [2021](#)
- 2006 - Present [Pattern Recognition Journal](#)
- 1991 - Present Regular reviewer for the following journals
[IEEE Transactions on Pattern Analysis and Machine Intelligence \(PAMI\)](#)
[IEEE Transactions on Neural Networks](#)
[International Journal of Control](#)
[The American Institute of Aerospace and Avionics \(AIAA\) Journal](#)

[Return to Table of Contents](#)

Society Membership

- 2010 - Present *Member*
[The New York Academy of Sciences](#)
- 2000 - Present *Senior Member*
The Institute of Electrical and Electronic Engineers ([IEEE](#))
 Member, [The IEEE Signal Processing Society](#)
 Member, [The IEEE Control Systems Society](#)
- 1988 - 2000 *Member*
The Institute of Electrical and Electronic Engineers ([IEEE](#))
- 1995 - Present *Full member*
[Sigma Xi Scientific Research Honor Society](#)
 Elected to the Columbia University (Kapp) Chapter
- 1991 - 2001 *Member*
[The Canadian Society for Electrical and Computer Engineering \(CSECE\)](#)
- 1991 - 1995 *Executive Committee Member*
 The Society for Technological Advancements in Developing Countries (STADC)
- 1991 - 2001 *Member*
[The American Society of Mechanical Engineers \(ASME\)](#)
- 1983 - 1991 *Student Member*
[The American Society of Mechanical Engineers \(ASME\)](#)

Miscellaneous

Citizenship United States of America

Languages Native level fluency in English and Persian.
Studied Japanese, up to Intermediate Level.

Hobbies [Persian Classical Musical performance in Fusion with Modern Western Music](#)
Instrumets: Tar, Kamancheh, Dotar, Tanbur, and Barbat

Hobbies include Etymology and the study of Archaic Languages such as
Middle English and Middle Persian.

Academics Participated in several courses in Statistical Speech Recognition and Advanced
Information Theory at IBM T.J. Watson Research Center, over many years.
In theses each participant taught a topic and attended the lectures of
other participants.

Publications

Books

Homayoon Beigi, *Mathematics of Machine Learning and Signal Recognition*, Springer, New York, 2023 (Expected).

More Information: This will be a textbook with over 900 pages of material. It has been solicited by colleagues who felt that background mathematics for machine learning and pattern recognition were not easy to find in a comprehensive manner with a single source. This book is nearly complete and is set to be released in the Fall of 2022.

Homayoon Beigi, *Fundamentals of Speaker Recognition*, 2nd Edition, Springer, New York, 2022.

More Information: Due to the success of the first edition of this book and the dynamic nature of the topic, this edition will provide the original text plus coverage of many new topics and techniques. The 2nd edition will be release toward the end of 2022.

Homayoon Beigi, *Fundamentals of Speaker Recognition*, 1st Edition, Springer, New York, 2011, ISBN: 978-0-387-77591-3.

More Information: This is the first and only textbook on speaker recognition. It contains a comprehensive coverage of the subject, with an inclusive treatment of all the prerequisites in 26 chapters and about 1000 pages of material. This book has consistently ranked as the top 25% downloaded E-Book on the Springer (publisher) site. This book took 4 full years of active research and writing, at an average of 10 hours per day, to be completed. Over 80,000 downloads have been recorded so far.

Downloads: 80,000 downloads as of December 2022 – According to Springer.com. This does not include the number of e-books and hardcopies sold. Interestingly, this book is becoming more popular every year. here are the statistics of the downloads for each year since it released:

- Downloads in 2018: **6,793**
- Downloads in 2017: **7,157**
- Downloads in 2016: **8,745**
- Downloads in 2015: **11,409**
- Downloads in 2014: **12,566**
- Downloads in 2013: **6,914**
- Downloads in 2012: **3,598**
- Downloads in 2011: **667** (only in the last 6 days of the year since it was release at the end of December, 2011)

Homayoon Beigi, “*Neural Network Learning and Learning Control Through Optimization Techniques*,” *Doctoral Thesis*, School of Engineering and Applied Science,

Columbia University, New York City, New York, 1991.

Book Chapters and Encyclopedia Articles

Kyle L. Hom, Homayoon Beigi, and Raimondo Betti “[Application of Speaker Recognition x-Vectors to Structural Health Monitoring](#)”, In: Mao, Zhu (ed), Model Validation and Uncertainty Quantification, Vol. 3, Ch. 13, pp. 139-148, isbn: 978-3-030-77348-9. Springer International Publishing, Berlin, Heidelberg, 2022, [10.1007/978-3-030-77348-9_18](#).

Eleonora M. Tronci, Homayoon Beigi, Maria Q. Feng, and Raimondo Betti “[Transfer Learning from Audio Domains: A Valuable Tool for Structural Health Monitoring](#)”, In: Grimmelsman, Kirk (ed), Dynamics of Civil Structures, Vol. 2, Ch. 11, pp. 99-107, isbn: 978-3-030-77142-3. Springer International Publishing, Berlin, Heidelberg, 2022, [10.1007/978-3-030-77143-0_11](#).

Homayoon Beigi, “[Speaker Recognition](#)”, In: Jajodia S., Samarati P., Yung M. (eds), Encyclopedia of Cryptography, Security and Privacy. Springer, Berlin, Heidelberg, 2021 [10.1007/978-3-642-27739-9_747-2](#).

Homayoon Beigi, “Andranik Aroustamian,” *invited article entry* in the [Encyclopædia Iranica](#), Center for Iranian Studies, Columbia University, 2021.

Homayoon Beigi, “[A Hybrid Approach to Automated Rating of Foreign Language Proficiency](#),” Where Humans Meet Machines – Innovative Solutions for Knotty Natural-Language problems, Amy Neustein and Judith Markowitz (Eds.), Springer, New York, 2013, ISBN: 980-953-307-576-6.

Homayoon Beigi, “[Speaker Recognition: Advancements and Challenges](#),” New Trends and Developments in Biometrics, Jucheng Yang and Shan Juan Xie (Eds.), 2012, ISBN: 980-953-307-576-6 DOI: 10.5772/52023.

Downloads: 3,533 downloads as of Jan. 6, 2022 – According to intechopen.com.

Homayoon Beigi, “[Speaker Recognition](#),” *invited article*, Encyclopedia of Cryptography and Security (2nd ed.), Henk C.A. van Tilborg and Sushil Jajodia (Eds.), Springer, New York, 2011, pp. 1232–1242, ISBN: [978-1-4419-5906-5](#), DOI: [10.1007/978-1-4419-5906-5_747](#).

Homayoon Beigi, “[Speaker Recognition](#),” [Biometrics / Book 1](#), Jucheng Yang (ed.), Intech Open Access Publisher, 2011, Ch. 1, pp. 3–28, ISBN: [978-953-307-618-8](#).

Downloads: 3,902 downloads as of Dec. 15, 2014 – According to intechopen.com.

Homayoon Beigi, “[Pre-Processing the Dynamics of On-Line Handwriting Data, Feature Extraction and Recognition](#),” Progress in Handwriting Recognition, A.C. Downton and S. Impedovo (eds.), World Scientific Publishers, New Jersey, 1997, pp. 191–198, ISBN: 978-981-023-084-5.

T. Fujisaki, H.S.M. Beigi, C.C. Tappert, M. Ukelson, and C.G. Wolf, “[Online Recognition of Unconstrained Handprinting: A Stroke-based System and Its Evaluation](#),” From Pixels to Features III: Frontiers in Handwriting Recognition, S. Impedovo and J.C. Simon (eds.), Elsevier Science Publishers, B.V., 1992, pp.297–312, ISBN: 0-44-489665-1.

Patents (granted)

Patent Number: US 10,042,993 B2

Homayoon Beigi, [Access control through multifactor authentication with multimodal biometrics](#),
Filed: June 23, 2015, Granted: August 7, 2018.

Patent Number: US 9,495,646 B2

Homayoon Beigi, Raimondo Betti, and Luciana Balsamo, [Monitoring Health of Dynamic System Using Speaker Recognition Techniques](#), Filed: June 5, 2014, Provisionally filed: June 5, 2013,
Granted: November 15, 2016; Joint filing between [Recognition Technologies, Inc.](#) and
[Columbia University](#).

Patent Number: US 9,064,257 B2

Homayoon Beigi, [“Mobile Device Transaction using Multi-Factor Authentication,”](#)
Filed: November 2, 2011, Provisionally filed: November 2, 2010, Granted: June 23, 2015.

Patent Number: US 7,474,770

Homayoon Beigi, [“Method and Apparatus for Aggressive Compression, Storage and Verification of the Dynamics of Handwritten Signature Signals,”](#) Filed: June 28, 2005,
Granted: January 6, 2009.

Patent Number: US 6,748,356

Homayoon Beigi and Mahesh Viswanathan, [“Methods and apparatus for identifying unknown speakers using a hierarchical tree structure,”](#) Filed: June 7, 2000, Granted: June 8, 2004.

Patent Number: US 6,684,186

Homayoon Beigi, Stephane Maes, and Jeffrey Sorensen, [“Speaker recognition using a hierarchical speaker model tree,”](#) Filed: January 26, 1999, Granted: January 27, 2004.

Patent Number: US 6,538,187

Homayoon Beigi, [“Method and System for Writing Common Music Notation \(CMN\) using a Digital Pen,”](#) Filed: January 5, 2001, Granted: March 25, 2003.

Patent Number: US 6,421,645

Homayoon Beigi, Alain Trischler, and Mahesh Viswanathan, [“Methods and Apparatus for Concurrent Speech Recognition, Speaker Segmentation and Speaker Classification,”](#)
Filed: June 30, 1999, Granted: July 16, 2002.

Patent Number: US 6,345,252

Homayoon Beigi, Alain Trischler, and Mahesh Viswanathan, [“Methods and Apparatus for Retrieving AudioInformation using Content and Speaker Information,”](#)
Filed: April 9, 1999, Granted: February 5, 2002.

Patent Number: US 6,253,179

Homayoon Beigi, Upendra Chaudhari, Stephane Maes, and Jeffrey Sorensen, [“Method and apparatus for multi-environment speaker verification,”](#) Filed: January 29, 1999, Granted: June 26, 2001.

Patent Number: [US 6,246,982](#)

Homayoon Beigi, Stephane Maes, and Jeffrey Sorensen, [“Method for measuring distance between collections of distributions,”](#) Filed, January 26, 1999, Granted, June 12, 2001.

Patent Number: [US 6,219,640](#)

Sankar Basu, Homayoon Beigi, Stephane Maes, Benoit Maison, Chalapathy Neti, and Andrew Senior, [“Methods and Apparatus for Audio-Visual Speaker Recognition and Utterance Verification,”](#) Filed, August 6, 1999, Granted, April 17, 2001.

Patent Number: [US 5,787,197](#)

Homayoon Beigi, Tetsunosuke Fujisaki, William Modlin, David William and Kenneth Wenstrup, [“Post-processing error correction scheme using a dictionary for on-line handwriting recognition,”](#) Filed: March 28, 1994, Granted: July 28, 1998.

[Return to Table of Contents](#)

Journal Publications

Eleonora M. Tronci, Homayoon Beigi, Raimondo Betti, and Maria Q. Feng “[A Damage Assessment Methodology for Structural Systems using Transfer Learning from the Audio Domain,](#)” Mechanical Systems and Signal Processing (MSSP), Vol. 195, July 15, 2023 ([online: Mar. 23, 2023](#)).

Eleonora M. Tronci, Homayoon Beigi, Maria Q. Feng, and Raimondo Betti, “[A Transfer Learning SHM Strategy for Bridges Enriched by the Use of Speaker Recognition x-Vectors,](#)” , Journal of Civil Structural Health Monitoring, May 23, 2022, DOI: 10.1007/s13349-022-00591-3.

Luciana Balsamo, Raimondo Betti , and Homayoon Beigi “[A structural health monitoring strategy using cepstral features,](#)” Journal of Sound and Vibration, Vol. 333, No. 19, January 2014, pp. 4526–4525.

Homayoon Beigi and Judith Markowitz, “[Standard Audio Format Encapsulation \(SAFE\),](#)” Journal of Telecommunication Systems (Special Issue of Biometrics Systems & Applications), Springer-Verlag, First Online Publication: May 26, 2010, Vol. 47, No. 3-4, 2011, pp. 147–162.

Homayoon Beigi, “[Guest Editorial,](#)” Special Issue on Learning and Repetitive Control, Special Issue of the Intelligent Automation and Soft Computing Journal, Volume 8, No. 2, 2002.

Konstantin Avrachenkov, Homayoon Beigi, and Richard Longman, “[Updating Procedures for Iterative Learning Control in Hilbert Space,](#)” Invited Paper, Intelligent Automation and Soft Computing Journal (Special Issue on Learning and Repetitive Control), Vol. 8, No. 2, 2002.

Mahesh Viswanathan, Homayoon Beigi, Satya Dharanipragada, Fereydoun Maali, and Alain Triteschler, “[Multimedia Document Retrieval Using Speech and Speaker Recognition,](#)” Invited Paper, International Journal on Document Analysis and Recognition, IJDAR, Vol. 2, No. 4, June 20, 2000, pp. 147–162.

Homayoon Beigi, “[Adaptive and Learning-Adaptive Control Techniques based on an Extension of the Generalized Secant Method,](#)” Intelligent Automation and Soft Computing Journal, Vol. 3, No. 2, 1997, pp. 171–184.

Homayoon Beigi and C. James Li “[Learning Algorithms for Feedforward Neural Networks Based on Classical and Initial-Scaling Quasi-Newton Methods,](#)” ISMM Journal of Microcomputer Applications, Vol. 14, No. 2, 1995, pp. 41–52.

C. James Li, Homayoon Beigi, Shengyi Li, Jiancheng Liang, “[Nonlinear Piezo-Actuator Control by Learning Self-Tuning Regulator,](#)” ASME Transactions, Journal of Dynamic Systems, Measurement, and Control, Vol. 115, No.4, December 1993, pp. 720–723.

Homayoon Beigi and C. James Li “[Learning Algorithms for Neural Networks Based on Quasi-Newton Methods with Self-Scaling,](#)” ASME Transactions, Journal of Dynamic Systems, Measurement, and Control, Vol. 115, No.1, March 1993, pp. 38–43.

Ferdinand Freudenstein and Homayoon Beigi, “[On a Computationally Efficient Microcomputer Kinematic Analysis of the Basic Linkage Mechanisms](#),” Invited Paper, Journal of Mechanism and Machine Theory, England, Vol. 21, No. 6, 1986, pp. 467–472.

[Return to Table of Contents](#)

Invited Tutorials

Homayoon Beigi, “[Voice: Technologies and Algorithms for Biometrics Applications](#),” IEEE eLearning Library (formerly IEEE Expert Now eLearning) Tutorial, Sep. 2010
Invited by the IEEE Expert Now organization to provide their tutorial for Speaker Recognition.

Homayoon Beigi, “[Speaker Recognition – Practical Issues](#)”, [SpeechTek University STKU-3, The SpeechTek Conference](#), New York City, August 5, 2010.

[Return to Table of Contents](#)

Conference Publications

Eleonora M. Tronci, Homayoon Beigi, Maria Q. Feng, and Raimondo Betti, “Transfer Learning from Audio Domains a Valuable Tool for Structural Monitoring,” IMAC, A Conference and Exposition on Structural Dynamics, Feb 8-11, 2021.

Kyle L. Hom, Homayoon Beigi, and Raimondo Betti, “Application of Speaker Recognition x -Vectors to Structural Health Monitoring,” IMAC, A Conference and Exposition on Structural Dynamics, Feb 8-11, 2021.

Luciana Balsamo, Raimondo Betti, and Homayoon Beigi, “[Damage Detection Using Large-Scale Covariance Matrix](#),” Proceedings of the 32nd IMAC, A Conference and Exposition on Structural Dynamics, 2014, Ch. 10, vol. 5, pp. 89-98.

Luciana Balsamo, Raimondo Betti, and Homayoon Beigi, “[Structural Damage Detection Using Speaker Recognition Techniques](#),” 11th International Conference on Structural Safety & Reliability (ICOSSAR 2013), June 16-20, 2013, Columbia University, New York City.

Homayoon Beigi “[Effects of Time Lapse on Speaker Recognition Results](#),” 16th International Conference on Digital Signal Processing, Santorini, Greece, July 5-7, 2009, pp. 1–6.

Homayoon Beigi and Judith Markowitz, “[A Standard Audio Encapsulation Method](#),” W3C Workshop on Speaker Biometrics and [VoiceXML 3.0](#), March 5-6, 2009, Menlo Park, CA, USA.

Homayoon Beigi “[Challenges of Large-Scale Speaker Recognition](#),” **Keynote Speech** at the European standards, [COST275 Workshop on Biometrics on the Internet](#), October 27-28, 2005,

Hertfordshire University, Hatfield, United Kingdom.

Homayoon Beigi, "[Aggressive Compression of the Dynamics of Handwriting and Signature Signals](#)," IEEE International Conference on Multimedia and Expo (ICME2004), Taipei, Taiwan, June 27-30, 2004, Vol. 2, pp. 1447-1450.

M. Viswanathan, F. Maali, A. Tritschler, and H. Beigi "[Multimedia Search for the Net.](#)," Proceedings of the SSRR 2001, 2nd International Conference on Advances in Infrastructure for E-Business, E-Science, and E-Education on the Internet, L'Aquila, Italy, August 2001.

Mahesh Viswanathan, Homayoon Beigi, and Fereydoun Maali, "[Information Access Using Speech, Speaker and Face Recognition](#)," IEEE International Conference on Multimedia and Expo (ICME2000), New York City, New York, 2000.

Mahesh Viswanathan, Homayoon Beigi, Alain Tritschler, and Fereydoun Maali, "[TranSegId: A System for Concurrent Speech Transcription, Speaker Segmentation and Speaker Identification](#)," WAC2000, Wailea, USA, June 11-16, 2000.

Mahesh Viswanathan, Homayoon Beigi, Alain Tritschler, and Fereydoun Maali, "[Multimedia Content Indexing and Retrieval Using Speech and Speaker Recognition](#)," Recherche d'Informations Assistee par Ordinateur, RIAO2000, Paris, France, April 12-14, 2000.

Konstantin E. Avrachenkov, Homayoon S.M. Beigi, and Richard W. Longman, "[Operator-Updating Procedures for Quasi-Newton Iterative Learning Control in Hilbert Space](#)," Invited Paper, IEEE Conference on Decision and Control (CDC'99), Phoenix, AZ, December 3-7, 1999.

Mahesh Viswanathan, Homayoon Beigi, Satya Dharanipragada, and Alain Tritschler, "[Retrieval from Spoken Document using Content and Speaker Information](#)," International Conference on Document Analysis and Retrieval, ICDAR99, Bangalore, India, September 20-22, 1999, pp. 576-581.

Homayoon Beigi, Stephane Maes, Upendra Chaudhari, and Jeffrey Sorensen, "[A Hierarchical Approach to Large-Scale Speaker Recognition](#)," Eurospeech'99, Budapest, Hungary, September 5-9, 1999, Vol. 5, pp. 2203-2206.

Upendra Chaudhari, Homayoon Beigi, Stephane Maes, and Jeffrey Sorensen, "[Multi-Environment Speaker Verification](#)," AUTOID'99, New Jersey, 1999.

Homayoon Beigi, Stephane Maes, and Jeffrey Sorensen, "[A Distance Measure Between Collections of Distributions and its Application to Speaker Recognition](#)," International Conference on Acoustics, Speech, and Signal Processing (ICASSP'98), Seattle, Washington, May. 23-27, 1998.

Homayoon Beigi and Stephane Maes, "[Speaker, Channel and Environment Change Detection](#)," Proceedings of the World Congress on Automation, Anchorage, Alaska, May. 18-22, 1998.

Homayoon Beigi, Stephane Maes, Upendra Chaudhari, and Jeffrey Sorensen,

[“IBM Model-Based and Frame-By-Frame Speaker Recognition,”](#) Speaker Recognition and its Commercial and Forensic Applications, Avignon, France, Apr. 20–23, 1998.

Stephane Maes and Homayoon Beigi, [“Open SESAME! Speech, Password or Key to Secure Your Door?”](#), Asian Conference on Computer Vision, Hong Kong, Jan. 8–11, 1998.

Homayoon Beigi “Starting Up as an ISP,” **Keynote Speech** at the Third Annual Workshop on Frontiers in Distributed Information Systems ([FDIS’97](#)) June 1–4, 1997, Aspen, Colorado.

Homayoon Beigi, [“Processing, Modeling and Parameter Estimation of the Dynamic On-Line Handwriting Signal,”](#) Proceedings of the World Congress on Automation, Montpellier, France, May. 27–30, 1996.

Homayoon Beigi, [“Pre-Processing the Dynamics of On-Line Handwriting, Feature Extraction and Recognition,”](#) Proceedings of the International Workshop on Frontiers of Handwriting Recognition, Colchester, England, Sep. 2–5, 1996, pp. 255–258.

Homayoon Beigi, Krishna Nathan, and Jayashree Subrahmonia, [“On-Line Unconstrained Handwriting Recognition Based on Probabilistic Techniques,”](#) ICEE-95, May 15–18, 1995.

Krishna Nathan, Homayoon Beigi, Gregory Clary, Jayashree Subrahmonia, and Hiroshi Maruyama, [“Real Time On-Line Unconstrained Handwriting Recognition using Statistical Methods,”](#) ICASSP-95, Detroit, Michigan, May 8–12, 1995.

Homayoon Beigi, Krishna Nathan, Gregory Clary, and Jayashree Subrahmonia, [“Challenges of Handwriting Recognition in Farsi, Arabic and Other Languages with Similar Writing Styles – An On-line Digit Recognizer,”](#) Proceedings of the 2nd Annual Conference on Technological Advancements in Developing Countries, Columbia University, New York, July 23–24, 1994.

Homayoon Beigi, Krishna Nathan, Gregory Clary, and Jayashree Subrahmonia, [“Size Normalization in Online Unconstrained Handwriting Recognition,”](#) The IEEE International Conference on Image Processing, Austin, Texas, November 13–16, 1994, Vol. I, pp. 169–172.

Homayoon Beigi, [“Neural Network Learning Through Optimally Conditioned Quadratically Convergent Methods Requiring NO LINE SEARCH,”](#) IEEE-36th Symposium on Circuits and Systems, Detroit, Michigan, August 16–18, 1993.

Homayoon Beigi, [“An Overview of Handwriting Recognition,”](#) Proceedings of the 1st Annual Conference on Technological Advancements in Developing Countries, Columbia University, New York, July 24–25, 1993, pp. 30–46.

Homayoon Beigi, [“Automation in Manufacturing,”](#) Proceedings of the 1st Annual Conference on Technological Advancements in Developing Countries, Columbia University, New York, July 24–25, 1993, pp. 85–92.

Tetsu Fujisaki, Krishna Nathan, Wongyu Chom and Homayoon Beigi, “On-line Unconstrained

Handwriting Recognition by Probabilistic Methods,” The 3rd International Workshop on Frontiers of Handwriting Recognition, Buffalo, New York, May 25–27, 1993.

Homayoon Beigi, T. Fujisaki, W. Modlin, and K. Wenstrup, “[A Post-Processing Error-Correction Scheme Using a Dictionary for On-Line Boxed and Runon Handwriting Recognition](#),” Proceedings of the Canadian Conference on Electrical and Computer Engineering, Toronto, Canada, Sep. 13–16, 1992, Vol. II, pp. TM10.5.1–4.

Homayoon Beigi, “[Character Prediction for On-line Handwriting Recognition](#),” Proceedings of the Canadian Conference on Electrical and Computer Engineering, Toronto, Canada, Sep. 13–16, 1992, Vol. II, pp. TM10.3.1–4.

Homayoon Beigi and T. Fujisaki, “[A Character Level Predictive Language Model and Its Application to Handwriting Recognition](#),” Proceedings of the Canadian Conference on Electrical and Computer Engineering, Toronto, Canada, Sep. 13–16, 1992, Vol. I, pp. WA1.27.1–4.

Homayoon Beigi and T. Fujisaki, “[A Flexible Template Language Model and its Application to Handwriting Recognition](#),” Proceedings of the Canadian Conference on Electrical and Computer Engineering, Toronto, Canada, Sep. 13–16, 1992, Vol. I, pp. WA1.28.1–4.

Homayoon Beigi, “[An Adaptive Control Scheme Using the Generalized Secant Method](#),” Proceedings of the Canadian Conference on Electrical and Computer Engineering, Toronto, Canada, Sep. 13–16, 1992, Vol. II, pp. TA7.21.1–4.

Homayoon Beigi, “[A Parallel Network Implementation of The Generalized Secant Learning-Adaptive Controller](#),” Proceedings of the Canadian Conference on Electrical and Computer Engineering, Toronto, Canada, Sep. 13–16, 1992, Vol. II, pp. MM10.1.1–4.

Homayoon Beigi, C. James Li, and R.W. Longman, “Learning Control Based on Generalized Secant Methods and Other Numerical Optimization Methods,” Sensors, Controls, and Quality Issues in Manufacturing, ASME:Atlanta, PED-Vol.55, pp. 163–175, Dec. 1991.

Homayoon Beigi and C. James Li, “Learning Algorithms for Neural Networks Based on Quasi-Newton Methods with Self-Scaling,” Intelligent Control Systems, Dynamic Systems and Control Vol. 23, the ASME Winter Annual Meeting, Dallas, TX, Nov. 25–30, 1990, pp. 23–28.

C. James Li, Homayoon Beigi, Shengyi Li, and Jiancheng Liang, “A Self-tuning Regulator with Learning Parameter Estimation,” Robotics Research, Dynamic Systems and Control Vol. 26, The ASME Winter Annual Meeting, Dallas, TX, Nov. 25–30, 1990, pp. 1–6.

Homayoon Beigi and C. James Li, “New Neural Network Learning Based on Gradient-Free Optimization Methods,” ***Recipient of IEEE Best Paper Award***, The IEEE 1990 Long Island Student Conference on Neural Networks, Old Westbury, NY, April 21, 1990, pp. 9–12.

Homayoon Beigi and C. James Li, “Neural Network Learning Based on Quasi-Newton Methods with Initial Scaling of Inverse Hessian Approximate,” ***Recipient of IEEE Best Paper Award***,

The IEEE 1990 Long Island Student Conference on Neural Networks, Old Westbury, NY, April 21, 1990, pp. 49–52.

Homayoon Beigi and C. James Li, “A New Set of Learning Algorithms for Neural Networks,” Proceedings of the ISMM International Symposium, Computer Applications in Design, Simulation and Analysis, New Orleans, LA, March 5–7, 1990, pp. 277–280.

R. W. Longman, Homayoon Beigi, C. James Li, “Learning Control by Numerical Optimization Methods,” Proceedings of the Modeling and Simulation Conference, Control, Robotics, Systems and Neural Networks, Pittsburgh, PA, Vol. 20, May 1989, pp. 1877–1882.

[Return to Table of Contents](#)

Keynote Speeches

Homayoon Beigi, “[Challenges of Large-Scale Speaker Identification](#),” *Keynote Speech*, First Speaker Identification Reunion of the Mexican Commission on National Security, Mexicali, Mexico, October 26, 2011.

Homayoon Beigi, “Standard Audio Format Encapsulation (SAFE),” *Panel Keynote Speech*, US Government Interagency Symposium for Investigatory Voice Biometrics, National Institute of Standards (NIST) Campus, Bethesda, MD, March 5, 2009.

Homayoon Beigi, “[Challenges of Large-Scale Speaker Recognition](#),” *Keynote Speech* at the European standards, [COST275 Workshop on Biometrics on the Internet](#), October 27–28, 2005, Hertfordshire University, Hatfield, United Kingdom.

Homayoon Beigi “Starting Up as an ISP,” *Keynote Speech* at the Third Annual Workshop on Frontiers in Distributed Information Systems ([FDIS’97](#)) June 1–4, 1997, Aspen, Colorado.

[Return to Table of Contents](#)

Invited Talks and Radio Interviews

Homayoon Beigi, “[Mobile Device Transaction Using Multi-Factor Authentication](#)” *Invited Talk*, [IEEE Tappan Zee Subsection Joint Meeting with Westchester Chapters of ASME & SME](#), New York Medical College, Valhalla, NY, May. 26, 2015.

Homayoon Beigi, “[Multimedia Diarization through Audio-Visual Recognition \(Fusion of Speaker and Face Biometrics with Speech Recognition and Natural Language Processing for achieving a Robust Diarization\)](#),” *Invited Talk*, [Philips Research, North America](#), Briarcliff, NY, Feb. 11, 2015.

Homayoon Beigi, “[Iterative Learning, Repetitive, and Learning-Adaptive Control](#),” *Invited Talk*, Cardio-Pulmonary Group, [Philips Research, North America](#), Briarcliff, NY, Feb. 11, 2015.

Homayoon Beigi, “Large-Scale Speaker Diarization,” [Goldman Sachs](#),

Invited Talk, Jersey City, New Jersey, U.S.A., July 21, 2014.

Homayoon Beigi, “[Mobile Device Transactions using Multi-Factor Authentication](#),” *Invited Talk*, [AVIOS Local Chapter Meeting](#), Columbia University, New York City, October 8, 2013. ([Photos and Abstract](#))

Homayoon Beigi “Mobile Device Transaction Using Multi-Factor Authentication,” *Invited Talk*, [The SpeechTEK Conference 2013](#), New York City, August 19-21, 2013.

Homayoon Beigi, “RecoMadeEasy[®] Speaker, Face, & Speech Engines,” *Invited Talk*, [Language Testing international](#), White Plains, NY, August 9, 2013.

Homayoon Beigi, “Speaker Recognition,” *Invited Talk*, [DIRECTV](#), El Segundo, CA, February 8, 2013.

Homayoon Beigi, “The RecoMadeEasy[®] Speaker Recognition Engine,” *Invited Talk*, [Booz Allen Hamilton](#), Falls Church, VA, December 12, 2012.

Homayoon Beigi, “Research Projects at Recognition Technologies, Inc.,” *Invited Talk*, [Center for Language Studies, Brigham Young University](#), Provo, UT, U.S.A., December 3, 2012.

Homayoon Beigi, “The Status of Speaker Recognition Research,” *Invited Talk*, In Commemoration of the 40th Anniversary of Prof. Richard Longman, [Department of Mechanical Engineering, Columbia University](#), New York City, NY December 9, 2011.

Homayoon Beigi, “Pattern Recognition for Fraud Detection,” *Invited Talk*, [Analytics Group, Citibank Headquarters](#), Long Island City, NY, November 30, 2011.

Homayoon Beigi, “Automatic Large-Scale Speaker Recognition,” *Invited Talk*, [University of Florida](#), Gainesville, FL, U.S.A., September 28, 2011.

Homayoon Beigi, “Speaker Recognition, Anywhere, in Any Language,” *Invited Talk*, [Analytics Group, Citibank Headquarters](#), Long Island City, NY, October 24, 2010.

Homayoon Beigi, “[Speaker Recognition – Practical Issues](#),” *Invited Mini Course*, [The SpeechTek Conference](#), New York City, August 5, 2010.

Homayoon Beigi, [Invited Radio Interview](#), the Greenwich Entrepreneur Program (Community Radio), WGCH AM 1490 Talk Radio, Greenwich, CT, May 14, 2010.

Homayoon Beigi, “Speaker Recognition (A Tutorial),” *Invited Talk*, [Analytics Group, Citibank Headquarters](#), Long Island City, NY, January 10, 2010.

Homayoon Beigi, [Invited Radio Interview](#), the Greenwich Entrepreneur Program (Community Radio), WGCH AM 1490 Talk Radio, Greenwich, CT, October 27, 2009.

Homayoon Beigi, “[Speaker Recognition in Distance Learning](#),” *Invited Talk*,

[AVIOS Local Chapter Meeting](#), AT&T, New York City, September 17, 2009.

Homayoon Beigi “Speaker Recognition in Distance Learning,” *Invited Talk*
[The SpeechTEK Conference](#), New York City, August 25, 2009.

Homayoon Beigi, “Speaker Recognition (An Overview),” *Invited Talk*,
[David Sarnoff Research Center](#), Princeton, NJ, August 8, 2009.

Homayoon Beigi, “Large-Scale Real-Time Constrained Nonlinear Optimization,”
Invited Talk, [Merrill-Lynch Research](#), World Financial Center, New York City, June 23, 2000.

Homayoon Beigi, “On-Line Unconstrained Handwriting Recognition,” *Invited Talk*,
Electrical Engineering Department, [Sharif University of Technology](#), May 22, 1995.

Homayoon Beigi and Fazlollah Reza, “Advancement in Developing Countries,”
Invited Talk, [Voice of America in Persian](#), August 5, 1994.

Homayoon Beigi, “Unconstrained Online Handwriting Recognition,” *Invited Talk*,
[MIT Media Lab](#), Cambridge, MA, June 1993.

Homayoon Beigi, “Online Handwriting Recognition,” *Invited Talk*,
[the American Chemical Society](#), Fishkill, NY, November 1992.

Homayoon Beigi, “Neural Network Learning Through Optimization Techniques,”
Invited Talk, The IBM Neural Networks Internal Technological Liaison (ITL),
IBM Fishkill, NY, May 11-13, 1992.

Homayoon Beigi, “Neural Network Learning and Learning Control Through,” *Invited Talk*,
[Department of Mechanical Engineering, California Institute of Technology \(Caltech\)](#),
Pasadena, CA, December 1990.

[Return to Table of Contents](#)

Technical Reports

Xing Yi Liu and Homayoon Beigi, “Efficient Ensemble with Time-Delay Neural
[Network for Punctuation Restoration](#),” [Recognition Technologies](#) Technical Report
No. [RTI-20230224-01](#), Feb. 24, 2023 , DOI: [10.13140/RG.2.2.29800.75528](#).

J. Nathaniel Holmes and Homayoon Beigi, “A Transaction Represented with Weighted
[Finite-State Transducers](#),” [Recognition Technologies](#) Technical Report No.
[RTI-20230131-01](#), Jan. 31, 2023 , DOI: [10.13140/RG.2.2.35825.15205](#).

M. Eyceoz, J. Lee, and H. Beigi, “Modernizing Open-Set Speech Language
[Identification](#),” [Recognition Technologies](#) Technical Report No. [RTI-20220520-01](#),
May 20, 2022, DOI: [10.13140/RG.2.2.24797.28647](#).

S. S. Nijhawan and H. Beigi, “Bi-LSTM Scoring Based Similarity Measurement with

Agglomerative Hierarchical Clustering (AHC) for Speaker Diarization,”
Recognition Technologies Technical Report No. RTI-20220519-02, May 19, 2022,
DOI: [10.13140/RG.2.2.13977.29288](https://doi.org/10.13140/RG.2.2.13977.29288).

B. Kepecs and H. Beigi, “Automatic Spoken Language Identification using a Time-Delay
Neural Network,” Recognition Technologies Technical Report No. RTI-20220519-01,
May 19, 2022, DOI: [10.13140/RG.2.2.21631.89763](https://doi.org/10.13140/RG.2.2.21631.89763).

A. Ananthram, K. K. Saravanakumar, J. Huynh, and H. Beigi, “Multi-Modal Emotion
Detection with Transfer Learning,” Recognition Technologies Technical Report
No. RTI-20201113-01, Nov. 13, 2020, DOI: [10.13140/RG.2.2.31373.97760](https://doi.org/10.13140/RG.2.2.31373.97760).

S. Zhou and H. Beigi, “A Transfer Learning Method for Speech Emotion Recognition from
Automatic Speech Recognition,” Recognition Technologies Technical Report
No. RTI-20200330-01, Mar. 30, 2020.

L. Ai, S. Jeng and H. Beigi, “A New Approach to Accent Recognition and Conversion for
Mandarin Chinese,” Recognition Technologies Technical Report No. RTI-20200218-01,
Feb. 18, 2020.

J. Leon, G. Chacko, H. Eng, D. Weise, E. Ruiz, and H. Beigi, “A Study of Smartphone
and Laptop Security Acceptability and Ease-of-Use,” Recognition Technologies Technical
Report No. RTI-20150128-01, May 1, 2015, DOI: [10.13140/RG.2.1.2103.1848](https://doi.org/10.13140/RG.2.1.2103.1848).

J. Justiniano, C. Javier, A. Blecher, and H. Beigi, “Acceptability Research for
Audio Visual Recognition Technology,” Recognition Technologies Technical
Report No. RTI-20150128-01, Jan. 28, 2015, DOI: [10.13140/2.1.3521.0565](https://doi.org/10.13140/2.1.3521.0565).

Homayoon Beigi, “Audio Source Classification using Speaker Recognition Techniques,”
Recognition Technologies Technical Report No. RTI-20110201-01, Feb. 2011.

Homayoon Beigi, “Computer Rating of Oral Test Responses using Verbosity,” Recognition
Technologies Technical Report No. RTI-20091211-01, Dec. 2009.

Homayoon Beigi, “Whether Computer Analyses Can Predict Human Ratings of Speaking
Proficiency,”
Recognition Technologies Technical Report No. RTI-20081205-01, 2008.

Konstantin Avrachenkov, Homayoon Beigi, and Richard Longman “Updating Procedures for
Iterative Learning Control in Hilbert Space,” IBM Research Tech. Report No. RC21558, 1999.

Homayoon Beigi, Stephane H. Maes “Speaker, Channel and Environment Change Detection,”
IBM Research Technical Report No. RC21022, 1997.

Homayoon Beigi, “Neural Network Learning Through Optimization Techniques,” *Invited Talk*,
Neural Networks Internal Technological Liaison (ITL), IBM Fishkill, NY, May 11-13, 1992.

[Return to Table of Contents](#)

Products

I have designed and implemented many different engines and algorithms and in fact I have personally done all the coding related to all the products of Recognition Technologies, Inc. and Internet Server Connections, Inc. Jsut the products at [Recognition Technologies, Inc.](#) include in excess of 1.5 million lines of highly optimized C^{++} code with almost no duplication. All engines have a common API standard and provide results in XML, HTML, and Text formats. All engines share the vast libraries which I been developed over the last 15 years. These libraries are listed at the [bottom of this section](#).

RecoMadeEasy[®] Speaker Recognition Engine (2003 – Present)

Winner of the [2011 Forst & Sullivan North American Speaker Verification Biometrics, New Product Innovation Award](#)

I personally designed and wrote all the code related to this engine in C^{++} . This is a state-of-the-art language- and text-independent speaker recognition system (voice biometrics system) which has been developed to work in different environments. Large-Scale and Small-Scale versions of this speaker identification and speaker verification (SIV) engine have been developed over many years of research to work in telephone, as well as stand-alone environments.

The speaker recognition engine is the only engine in the market, capable of realtime large-scale speaker identification, handling hundreds of thousands of models in realtime. It runs on all Linux systems as well as iOS and Windows. It features its own efficient database and a very small footprint. Another very important discriminator of this product is related to the *amount of data* that is available to us in the training of the models. Having data from about 1.5 million speakers, the engine has been optimized to handle very large training samples and to allow for parallel processing in both training and testing scenarios.

This engine has been trained on the speech of about 1.5 million of distinct speakers, most of whom have more than one session of recording. Being a statistical engine, large amounts of data and ample variability among types of data are essential to its success. For over a decade, customers in the financial, education, security, and government sectors have been using this product and coming back for more licenses.

RecoMadeEasy[®] Large Vocabulary Speech Recognition Engine (2003 – Present)

This large vocabulary speech recognition engine was written in C^{++} and shares many libraries from the Speaker Recognition engine as well as many other mathematical libraries developed over the last 15 years. At the moment, this is the most active area of research at [Recognition Technologies, Inc.](#)

Many applications are aslo being developed in collaboration with customers who are using this engine for phrase spotting in financial applications and proficiency rating in distance learning settings, etc. The speech recognition engine shares an API with the rest of the engines at Recognition Technologies to make the job of fusion of different engines much easier.

RecoMadeEasy® Face Recognition Engine (2010 – Present)

The face recognition effort was started in 2010 to complement the speaker recognition engine, in providing an audiovisual fusion engine, as well as a standalone full-frontal face recognition engine. As with all other engines, it shares an API with all RecoMadeEasy® products and provides very robust recognition results which perform well in different lighting conditions as well as in discrepancies related to glasses and other anomalies.

RecoMadeEasy® Audiovisual Recognition Engine (2010 – Present)

Since the inception of the Face Recognition engine, it has been fused with the speaker recognition engine, as well as speech recognition engine. This engine is the most complete engine which allows for the use of speaker, face, and speech recognition. The speech recognition is used to allow for random prompts to test for liveness, as well as simple standalone speech recognition functionality. As far as I know, we are the only company which has all these engines integrated into a single engine with a shared API.

RecoMadeEasy® Access Control Engine and Android Application (2012 – Present)

The [RecoMadeEasy® Audiovisual Recognition Engine](#) which includes [speaker recognition](#), [face recognition](#), and [speech recognition](#) with an extra capability that allows it to communicate with different access control devices. This allows for the engine to be used to gain access to different physical or virtual locations. See [video demonstrations](#). The demonstration also shows the [Android](#) application which communicates with this engine and was also developed completely in-house at [Recognition Technologies, Inc.](#) See [video demonstrations](#).

RecoMadeEasy® Automatic Language Proficiency Rating Engine (2007 – Present)

Certain clients of [Recognition Technologies, Inc.](#), who are involved in performing enormous amounts of ratings for their own clients in over 100 languages started asking about the possibility of rating the proficiency of an oral test, preferably independent of the language. Aside from the fact that this would save time and money for the client, there was a much more important reason for needing this service. Human raters can only provide proficiency scores up to a certain granularity. Practically, about 80% of all the rated tests would fall into a single human-rated category. I was able to provide three subratings for Intermediate Mid range which allowed for practical rating of the proficiency of over 1.5 million tests since 2007. See the related [book chapter](#) and [related reports](#).

RecoMadeEasy® Interactive Voice Response (IVR) (2003 – Present)

The Interactive voice response application was the first product of [Recognition Technologies, Inc.](#). It has been designed so that even a top executive who knows nothing about coding would be able to create a new IVR process using a Graph language which I wrote, specifically for defining IVR processes. It resembles C syntax, but it allows for defining nodes of a graph and the relation between nodes. It allows for very complex graphs. This product has been in use, nonstop, since 2003 and has been used to record at least 30,000,000 minutes of conversational audio. It is capable of recording conversations in split channel settings and is fully compatible with the most popular [Dialogic](#) telephony cards.

RecoMadeEasy® Signature Compression and Verification Engine (2005 – Present)

The signature compression product implements the content of my [U.S. patent 7,474,770](#). This application allows for an aggressive compression such that all the dynamics of a signature are compressed into less than 54 bytes. This is the number of bytes of storage which is available on the back of standard magnetic credit or debit card. Since the dynamics are preserved, the signature verification engine uses this information to match templates for users.

RecoMadeEasy® Online Handwriting Recognition Engine (2003 – Present)

The handwriting recognition engine is capable of performing unconstrained online handwriting recognition on a series of sampled points, coming back from a tablet, for example a [Wacom tablet](#).

RecoMadeEasy® Keystroke Recognition Engine (2007 – Present)

The keystroke engine has been developed to allow for the identification of individuals based on their typing habits. It uses the timeline associated with keystrokes. It is text and language independent, although the enrollment in different languages would need to be associated with the corresponding test language in order to have higher accuracies.

CommerceMadeEasy® (2000 – Present)

Winner of the Best of Show Award at the 2002 Internet World Show, held at the Jacob Javits center, in New York City, it was also the recipient of the Product of the Day award for October 25, 2002 from the Linux Journal. [CommerceMadeEasy®](#) is a Linux-based server software, all components of which were developed in C^{++} for optimal performance and security. It is completely cookie-less and provides a secure Internet Wizard for creating new accounts. It provides services such as "Sales, Auction, Access and Contribution and now E-Learning." The Wizard may be used to quickly setup a commerce site from an existing website with advanced search, a credit card gateway, security, and many other capabilities.

[CommerceMadeEasy®](#) may be used to set up and create complex commerce sites in any industry, in a matter of days. As the package was developed entirely in-house and since it includes all components such as an optimal database interface, it permits easy customization.

Encyclopedia Digital Library (1998 – 2012)

[Encyclopædia Iranica](#) is a thirty five year old scholarship endeavor at the [Center for Iranian Studies](#) at [Columbia University](#). Over the course of 14 years, I developed the procedure for the conversion of classically written articles into articles in Unicode-16. This included working with the editors of the encyclopedia, in detail, to design a character set based on Unicode to be able to handle hundreds of Indo-Iranian languages (modern and archaic). In addition, automation software was developed for the conversion of articles, while indexing and cross-referencing then automatically. Created search mechanisms for searching the articles in any of the many nonstandard transcription techniques used by the readers to transcribe the relevant languages, such as Persian, into the latin alphabet. The unicode mapping, unicode search, and multiple transcription style mapping are quite

complex. The indexing included automatic indexing plus keyword references created manually by editors and incorporated into the search and indexing. The product was the creation of a fully functional digital library by which the print version was and is made available online.

Portfolio Optimization (1998 – 2001)

While working at IBM Research, I was approached by a team at [Merrill-Lynch Research](#), who had heard about my doctoral work on nonlinear optimization, to help them with a problem they had with optimizing portfolios. At the time, [Merrill-Lynch](#) was using a very expensive product, [the Barra Optimizer](#), for portfolio optimization. To conserve funds, they requested the creation of a portfolio optimization program that would do the job of Barra, as well as providing more features. After clearing it with IBM, I took on the optimization research while at IBM and then in February of 2000, I decided to leave IBM and work on this problem fulltime. My manager asked me to reconsider and to take a one-year sabbatical instead and see whether I would like to return to IBM. In February 2001, I decided to continue working on this challenging optimization problem and left IBM Research.

As the VP/CTO of [Internet Server Connections, Inc.](#), I wrote an optimization program, capable of optimizing portfolios based on constrained nonlinear optimization of 35,000+ international securities in realtime. I solved this problem using sparse nonlinear optimization techniques and was able to match the results obtained by Barra. Unfortunately, due to the unfortunate event of September 11, 2001, the Merrill-Lynch research group fell apart and the project was terminated. However, very useful optimization libraries were developed in the process and added to the [Internet Server Connections](#) mathematical libraries which were later transferred to [Recognition Technologies, Inc.](#) and made a lot of the basic functions possible for the development of the many recognition engines described above.

The very large-scale portfolios optimization project for [Merrill-Lynch Research](#) took place from 1998 to 2001.

Matlab to C++ Translator (1993 – 1996)

Wrote the first Matlab® to C++ translator. Wrote all internal matrix functions of Matlab® in C++. Wrote a Matlab® parser and a C++ code generator to produce C++ code, translating any Matlab® code to 100% C++ code. This project entailed optimal implementation of complex numerical matrix manipulation functions. The code libraries were later inherited by [Internet Server Connections, Inc.](#) and later by [Recognition Technologies, Inc.](#)

IBM ViaVoice®

Personally wrote the first version of the IBM Speaker Recognition Engine, from scratch, which became a part of the IBM ViaVoice® line of products. I worked on this project from 1996 to 2000 until I took a one-year sabbatical to work on the [portfolio optimization problem](#) at [Internet Server Connections, full-time](#). In the course of these four years, I created an engine which allowed for large-scale speaker identification, as well as the basic speaker verification modality. I created hierarchical algorithms for organizing speaker models, in order to be able to perform massive identifications runs by only matching a logarithmic number of models instead of matching all models, which was the norm at the time.

During the process of creating a large-scale speaker identification product, I formulated a *directed divergence method* between collections of probability densities, which may use any classic divergence or distance between two densities.

IBM ThinkScribe® (CrossPad®) (1993 – 1996)

Contributions to the underlying algorithms of the IBM ThinkScribe® products, Compression (sole investigator and developer), Handwriting Recognition Technology (principal investigator and developer). This product offered a very robust unconstrained handwriting recognition engine with over 85% word-level accuracy, in conjunction with applications which were shipped with the CrossPad®.

IBM ThinkWrite® (1991 – 1993)

Contributions to the underlying algorithms of the IBM ThinkWrite® products, Run-On and Discrete Handwriting Recognition Technology one of the principal investigators and developers.

Wholesale Inventory and Sales Control® (1984 – 1990)

Wrote a complete inventory and sales control application, originally in BASIC and translated it to C in 1985. The more superior C version ran on the Xenix® operating system (a flavor of Unix®) and was sold to 5 wholesale carpet companies which used it for well over a decade. The product featured a colorful menu-driven interface, written using *curses*. This interface was state-of-the-art at that time. In addition, I developed drivers for a wireless pen-based barcode reader with memory which could hold up to 100 barcodes, using a serial port interface. I also used serial communication to connect dumb terminals and serial printers in the different locations of the showroom and the loading zones. I wrote the whole application entirely in C, with the following flow structure:

1. Salesperson accompanies client to the showroom with a barcode reading pen in his/her pocket.
2. Client chooses different carpets and salesperson swipes the barcode using the pen.
3. They go back to the office and salesperson puts pen in a pen-holder which is connected to his terminal through a serial interface.
4. The salesperson brings up the client's account in an empty invoice and the barcode numbers are loaded into the sales slip.
5. The salesperson completes the sale and prints the invoice in the office.
6. In the meanwhile a loading receipt is automatically printed in the docking area and the pieces are prepared for pickup.

The above scenario is very simple to handle with today's technology, but with the technology available in 1985, there were many hurdles to handle. The system was so stable that even as late as 1995 one of the clients was still using the system.

At the time, DOS solutions were only capable of using Novell networks with very limited capabilities, at great costs. For example, a 5-station system would have used an IBM PC for each station and would only allow peer-to-peer networking and would cost around \$30,000. However, this pure multiuser solution only cost 6,000.

Products *(Libraries written by Homafoon Beigi in the last 15 years)*

These libraries contain over 1.5 million lines of code and include, but are not limited to,

1. *Encryption Library* – Capable of doing encryption/decryption, hashing, coding, etc.
2. *Database Library* – Includes completely inclusive database design, all developed in-house. This library is capable of 64-bit addressing, advanced regular expression searches. It is fully configurable through text configuration files and allows over 25 different types including Credit Card information, passwords, hash codes, basic types, Email addresses, URLs, etc.
3. *Common Gateway Interface (CGI) Library* – This library allows for the conversion of any C++ application to one operating as a CGI application which would run through a web interface. This library has been under development since 1996 and has great capabilities. It natively links with the Database and Encryption libraries listed above, to create a seamless CGI interface. This library is one of the core components of the **CommerceMadeEasy® product**, as well as other products.
4. *Error Handling Library* – This library allows for error handling and tracking, throughout any product that links with it. It is an essential part of all products and makes interactions quite practical.
5. *Mathematics Library* – This library contains a very rich set of mathematical functions which are based on the handling of matrices in a C++ setting. The interface to this library is very simple and all functions have been optimized over the past 15 years to provide very fast operations. It is an integral part of all product engines.
6. *Licensing Library* – This complete license management library provides licensing capabilities to all products. It makes the distribution of the engine possible. It is designed so that new modules may be easily configured into the license with full backward compatibility. It works in conjunction with the encryption library and it is an essential part of all products. This library supports different operating systems including Linux, Mac, and Windows, each of which require different techniques for handling the license management.
7. *RecoMadeEasy® Library* – This is a base library which is capable of handling configuration files, understands how to interact with the licensing library, includes most of the basic operations that any recognition would use. It is inherited by all product engines with this registered trademark name.
8. *Access Library* – This library provides functionality for access-control related applications. See [video demonstrations](#).
9. *API Library* – Since all engines have a unified API, this base library, once inherited, allows for all API functionality.
10. *Audio Library* – This most essential library handles all lower level processes related to handling different audio formats and codecs. It is used by any engine which requires an audio interface.

11. *AudioVisual Library* – This library works in conjunction with the audio and image libraries to allow the handling of a combination of these two media, including video codecs.
12. *Clustering Library* – This library handles all supervised and unsupervised clustering functionality. It is essential to all pattern recognition engines.
13. *Face Library* – This library provides all the lower level (algorithmic) functionality related to face recognition.
14. *Handwriting Library* – This library handles all handwriting and signature related functionality including algorithmic, capture, and feature extraction mechanisms.
15. *Image Library* – This library handles arrays of images, their representations, manipulations, algorithmic and input/output aspects. It is the analogue of the audio library in the image domain.
16. *Input Library* – This library handles all input related aspects including keyboard, tablet, etc.
17. *IVR Library* – The Interactive Voice Response (IVR) library is a complete library which includes agnostic IVR functionality as well as wrappers for popular cards such as most [Dialogic](#) cards. It uses a fully connected graph mechanism which is a part of the tree library. This is the most essential part of the IVR engine and has been in use at Recognition Technologies, more than any other library, in an intense use scenario. This library has been used to record more than 30,000,000 minutes of conversational audio within the systems of [Recognition Technologies, Inc.](#)
18. *Language Modeling Library* – This library is capable of loading and manipulating huge language models including NGram models and grammars in highly efficient proprietary formats, in the memory. It is an essential part of the speech recognition engine.
19. *Media Library* – The media library is an intermediate library which combines and understands audio and image formats and knows how to relate them to video codecs. It is an integral part of all audiovisual engines.
20. *Memory Map Library* – This library allows for the seamless use of memory-mapped files in place of regular memory, in any type of process within the RecoMadeEasy® family of engines and products.
21. *Search Library* – This library includes the implementation of several optimal and suboptimal search algorithms which, for example, are used for speech and handwriting recognition engines.
22. *Speech Library* – This library provides most algorithmic functionality needed by speech recognition, which is not in common with speaker recognition.
23. *Speech Front-End Library* – This library handles all signal processing, feature extraction, and feature manipulation for speech and speaker recognition.
24. *Speech Utility Library* – This library provides a large number of algorithmic and manipulation functionalities for use with speech and speaker recognition.
25. *Speaker Library* – This library provides higher level speaker recognition functionality such as different speaker recognition modalities.

26. *Transform Library* – This library provides many mathematical transformation libraries. This library is used by most recognition engines.
27. *Windows Library* – This library provides transformations in order to unify the code so that there is no need to write specific code to run on the Windows operating system. This library makes the job of maintaining a Windows engine much easier.
28. *Text IO Library* – This library provides many text manipulation and input/output functionalities which is at the heart of all other libraries. These could be parser functions for the language related to tree or graph definitions, configuration file definitions, etc.
29. *Tree Library* – This last, but by no means least library provides a very powerful set of functionalities for handling different types of trees and graphs. It has been optimized immensely through the past 15 years and it is used by most libraries, including the handling of recursive and cascaded configuration files, graph definitions for the IVR, etc.

[Return to Table of Contents](#)

Research Projects

The following is a non-exhaustive list of the research projects, in which I have been indulging. They include work that started as far back as when I was doing my graduate studies at Columbia University. Many of the projects are still alive and there is ongoing research in all aspects. Because of the many years of work that has been done on these research projects and due to special attention to organization and generalization of the code, a highly efficient research effort has been brewing for many years. I owe most of this to the multi-disciplinary attitude that I have had, since the inception of my research effort. I believe that nothing should be done more than once and that it is important to generalize both formulation of a problem and its implementation so that a lot of the work may be reused for other problems that may come up.

Speaker Recognition (1996 – Present)

I started working on speaker recognition by developing algorithms and creating the first version of the [speaker recognition engine of IBM Research](#) [1]. I have developed a number of successful techniques for large population speaker recognition in addition to general research and development in the field of pattern recognition with applications to Speaker Recognition (Large-Scale Identification and Verification), Segmentation, hybrid systems with face recognition and text processing, large scale search techniques, etc. This work is still ongoing and has been turned into a [product](#) [2] at [Recognition Technologies, Inc.](#) which became the only winner of the [2011 Frost & Sullivan North American Speaker Verification Biometrics, New Product Innovation Award](#).

As a most important achievement, I wrote the first and only [textbook on speaker recognition](#) [3] and many other [book chapters](#) [4, 5, 6], [journal papers](#), etc. I have also extended these techniques to the problem of [structural health monitoring](#).

An important aspect, making the research results practical enough to be used in [products](#) [2] is the immense *data collection initiative* at [Recognition Technologies, Inc.](#) for the past 11 years. Due to the statistical nature of the algorithms, it is essential that large representative amounts of data are used. Due to this effort, [Recognition Technologies, Inc.](#) has the largest number of speakers recorded in natural settings with different sessions. This data has been used for the training of the speaker recognition models. This data entails over 1.5 million distinct speakers in many different settings, many of whom have been recorded in multiple sessions.

In the past three years, I have also hired an average of three interns per summer and a full time research assistant from the computer science and electrical engineering departments of Columbia University and have mentored them in conducting research and developing products in speaker [2] and speech recognition [7] engines.

Face Recognition (2010 – Present)

Developed very robust face recognition algorithms with practical implementations, used in the identification and verification of full-frontal faces in very large populations with great success. This work is still ongoing and has been turned into a [product](#) [8] at [Recognition Technologies, Inc.](#) As

with the rest of the research projects, I am reusing most of the [library functions](#) that I have used for speaker recognition, speech recognition, and other projects in the face recognition research. This is the main reason for the quick results obtained in this work, leading to product-level code.

Speech Recognition (1996 – Present)

Extensive participation in the research connected with the [IBM ViaVoice product](#) [1]. Research areas include all aspects of Large-Vocabulary speech recognition, speech segmentation, and Small-Vocabulary systems.

In 2003, I continued performing speech recognition research at [Recognition Technologies, Inc.](#) This led to the development of several [speech recognition engines](#) [7], as well as a [language proficiency rating engine](#) [9]. The speech recognition and speaker recognition engines share as many of the [libraries](#) as possible. This allows for faster implementation of new algorithms across the two paths. Currently, most of my effort is on the improvement of the search engines and models. Currently a combination of Hidden Markov Models and Deep Belief Networks are being used in my research. The [nonlinear neural network learning algorithms](#) that I developed in the late 1980s and early 1990s are being used for the training of the networks. Also, techniques such as islands of high probabilities are being utilized for improving the search results.

Since 2012, interns from [Columbia University](#) have been helping with the large number of training and test experiments that need to be done for testing different algorithms.

Biometric Fusion and Multimedia Diarization (1997 – Present)

Speaker diarization is one of the most practical research projects which benefits any industry which archives and searches through multimedia files such as video and audio files. This has been a major focus of my work and has led to my development of some of the [first speaker segmentation](#) [10] and [large-scale speaker identification](#) [11] systems. I have been creating the constituents of a good diarization system by working, for years, on speaker recognition, face recognition, speech recognition, and methods for combining the results. In the process, I have developed many valuable products based on fusing these systems at [Recognition Technologies, Inc.](#) such as [RecoMadeEasy® Audiovisual Recognition Engine](#) [12] and [RecoMadeEasy® Access Control Engine](#) [13] and many products at [IBM Research](#).

One of my major accomplishments at IBM Research was the [fusion of speaker recognition, face recognition, and speech recognition](#) [14] engines to produce quality diarization of video files. This produced meta data based on segmenting the media according to speaker change and a change in the output of the face recognition system. This produced labels for the video content. Simultaneous transcription through the speech recognition engine provided transcribed text. The combination of these knowledge sources produced searchable meta-data which allowed for further searches on specific content by requesting segments in which a specific individual spoke about a certain topic. This work produced many journal papers and patents. One of these patents was deemed as one of the Top 10% Valuable Patents (Patent Number [6,421,645](#)) at [IBM Research](#).

Mechanical and Structural Health Monitoring (1989 – 1990 & 2012 – Present)

Structural Health Monitoring (2012 – Present)

Joint (multidisciplinary) research with the civil engineering department of Columbia University on the use of speaker recognition techniques for the monitoring and prognosis of the health of structures such as bridges and buildings. This work was done with [Prof. Raimondo Betti](#) of the [civil engineering department](#) and it provided the material for a full PhD thesis [15], just defended by Luciana Balsamo. This work has resulted in a PhD thesis, as well as [several joint publications](#) [16, 17, 18] with [Prof. Betti](#) and Dr. Balsamo. [Prof. Betti](#) and I are still collaborating on this project and will be conducting a lot more joint research. In addition, at [Recognition Technologies, Inc.](#) I have rekindled the research on mechanical health monitoring and will be having new products for the market, in the near future.

Mechanical Health Monitoring (Jan. 1989 – Dec. 1990)

Fault detection of mechanical systems and machine health prognosis, funded by the [U.S. Navy](#) and supervised by Prof. C. James Li. Developed practical signal processing techniques for the health prognosis of mechanical components such as bearings, gears, cutting tools, etc. This project included the design and implementation of the sensors and the data acquisition apparatus, as well as extensive pattern recognition algorithm design and implementation for the automatic detection of faults in different components.

Information Theory and Language Modeling (1991 – Present)

I performed extensive research in information theory and language modeling, pertaining to usage with Text Processing, Handwriting Recognition, Speech Recognition, and Hybrid Search systems using Textual Language as well as Speaker Voice and Face information. This also evident from the extensive treatment of highly compressed dictionaries [19], statistical N-Grams [20, 21], Template Language models [22] and Decision-Tree-based Language Models.

Aside from my original work in the above topics, in my [textbook on speaker recognition](#) [3], I have given a very complete and detailed coverage to information theoretic concepts since they lie at the basis of most of the work in machine learning and pattern recognition. I have also made a lot of observations regarding different concepts that have baffled researchers in the past such as the relation between the definition of Information as presented by Wiener and Shannon in the same year (1948), along with in depth analysis of the seminal works in the field.

Education and Language Proficiency Testing (2006 – Present)

At Recognition Technologies, Inc., starting with [Language Testing International](#) (LTI, now a division of [Samsung](#)) and the [American Council on the Teaching of Foreign Languages](#) (ACTFL), I was asked to automate the rating of tests conducted on over 100 languages for assessing proficiency of professionals seeking to be hired by large multi-national corporations and government agencies. [ACTFL](#) is a nonprofit organization defining proficiency test procedures and [LTI](#) is a profit-oriented company administering such tests. [LTI](#) administers oral, conversational, and written tests in different capacities. In their oral tests, their best raters rate 80% of the population as Intermediate Mid level which is one of 10 possible levels (Novice Low, Novice Mid, Intermediate Low, Intermediate

Mid, Intermediate High, Advanced Low, Advanced Mid, Advanced High, and Superior. This had presented them with the quandary of handling ratings in a more granular fashion to produce more discriminability. In addition, they rate over 1,000 tests per day for just a single site in Korea. It was very important to automate this rating process.

I worked on the problem starting in 2006 and by 2008 was able to modify our speech and speaker recognition engines to automatically rate oral tests at a rate of 10 times realtime, attaining three times the granularity of the human ratings (breaking Intermediate Mid to IM1, IM2, and IM3 levels). These granularities are not achievable by normal raters. In fact, in order to evaluate our engine's rating quality, [LTI](#) asked three of their top 1% most experienced raters to independently rate over 1000 tests at the higher granularity. Then our engine rated the same tests. For the ratings where 2 of the 3 raters agreed, the rating was over 85% correlated with our results. This was more than double the consistencies achieved by their best human raters. Also, the best their raters would do, in terms of speed, would be understandably at best realtime since they would have to listen to the whole test to make a decision. Whereas we rated each test at ten times realtime! (See the related [book chapter](#) [23] and [technical reports](#) [24, 25].

The success of this research warranted large grants over the past 7 years from [LTI](#), [ACTFL](#), and [the Center for Language Studies of Brigham Young University](#) for this and other research on written and elicited oral testing. Currently, I am utilizing our [RecoMadeEasy® Speech Recognition engine](#) [7] in conjunction with our [RecoMadeEasy® Automatic Language Proficiency Rating engine](#) [9] to rate elicited responses for English and Italian.

My research in this field has attracted two more companies (an Italian company and a [start-up company](#) created by two of my ex-IBM colleagues. We are in the process of creating completely automated rating systems for students of English and Italian, to be following with other languages such as Polish.

Optimization (1986 – Present)

At the heart of most of my research, lies a strong base of optimization techniques. My [doctoral thesis](#) [26] was mostly based on the application of different nonlinear optimization techniques to the [learning-control](#) [27, 28, 29, 30, 31, 32, 33] and [neural network learning](#) [34, 35, 36, 37, 38, 39, 40, 41] problems.

In addition to the thesis, many of my other peer-reviewed publications show the use of these techniques. This was the motivation behind the [Merrill Lynch research group](#) seeking my expertise in solving a very tough constrained optimization problem with over 35,000 variables, for realtime solutions. See the related [portfolio optimization product](#) I produced, as a result of this research.

Handwriting Recognition (1991 – 1996 & 2003 – Present)

Over 6 years of research in large-vocabulary, unconstrained (any free-form combination of cursive, run-on, or discrete) handwriting recognition. Extensive work on Feature Extraction [42, 43, 44], Language Modeling, Search [45, 46, 47], Compression [48], Segmentation and Normalization [49] leading to a fully operational system [50]. I am presently, active in this area area of research through

In the feature extraction area, I devised many different kinds of new features for run-on, cursive, and eventually unconstrained handwriting recognition, including the creation of **new dynamic features** [42, 43] based on an approximation of the hand motion with a time-variant second order differential equation and the use of the time varying parameters as truly compressed features of handwriting for each stationary part of the signal. This was complemented by the **fusion of these dynamic features with static features** [46, 45], as different codebooks, to increase the accuracy of recognition.

The orientation and size normalization of unconstrained handwriting is a very hard problem. I presented **break-through techniques** for finding and correcting the orientation of handwriting, slant correction, rotation of the handwriting to abide by principal lines and finally size normalization so that increased robustness to orientation, size, slant, and other transformations was achieved. [49]

Search was another problem that I addressed and achieved considerably great results in improvements made to our system at **IBM Research**. Using the techniques that were used in speech recognition, namely **beam and envelope search**, and modifying them to work with unconstrained online handwriting recognition, I made great advancement in the field. This made having unconstrained handwriting recognition possible and allowed us to leap from discrete and run-on techniques to unconstrained recognition. A very important problem that I addressed in this algorithm design and implementation was the handling of delayed strokes such as dots of “i” and “j” and the cross of “t,” etc.

Another one of my very important contributions was the creation of predictive language models based on a highly compressed representation of **character ngrams** [20, 21] with extremely fast access, **templates** [22], **predictive and postprocessing dictionaries** [19], and many other aspects which mean the integration of these techniques with my advanced search techniques.

This research has led to many publications which have been cited considerably. In addition, it produced a few products, both at **IBM** and at **Recognition Technologies, Inc.** Since the **handwriting recognition group at IBM Research** was a sister department with the **speech recognition group**, the first 6 years of research I did were in conjunction with the **speech recognition group** and most of the algorithms were shared. In fact aside from the front-end and the analogy of characters in handwriting recognition to words in speech recognition, the rest was pretty much the same. I worked on all aspects of the problem and this was a great opportunity for me to get to understand the details of speech recognition as well, helping me to transit into the **speech recognition group** for the last few years I was at IBM.

A segment of online handwriting is an analogue of a frame in speech, but far less number of segments come up in a sentence in unconstrained handwriting recognition. This gave me the opportunity to get a better understanding and visualization of the search process than starting on the speech problem. Later, extending that experience to speech recognition helped me excel beyond other researchers who may have started directly in speech recognition.

This is yet another milestone in helping create my multi-disciplinary background.

Neural Network Learning (1986 – Present)

One of the most important works that I presented was the tensor formulation of the problem of learning in general feedforward neural networks which enabled me to apply many different advanced nonlinear learning algorithms to learning in neural networks. This work was presented in many publications as well as my [doctoral thesis](#) [26]. This notation which I picked up from my background in mechanical engineering, namely the treatment of solid and fluid mechanics stress, strain, and energy equations, makes the mathematical representation of the output of a neural network to its input, weights, and activation functions much more manageable.

With this, I made break-throughs in highly efficient learning schemes using second order techniques in optimization for Feed-Forward Neural Network Systems, right at the onset of the subject, in fact just a year after Rumelhart's seminal paper on back-propagation. I produced several journal and conference papers in this field including **two IEEE Best Paper Awards** for two conference papers in the field [39, 40].

This work has been revived around 2005 and on, in the name of deep belief networks. Some of this work is using stochastic versions of well-known optimization techniques. My work, which were published in the 1990s, are now being reinvented for learning in deep belief networks. Examples were presented at the New York meeting in 2013 which were foreseeing the possibility of using nonlinear techniques for solving the multilayer neural network learning problem. However, I had published these and more complex techniques, in the 1990s, facilitated by my tensor-notation based formulation of the learning problem in multi-layered neural networks. [34, 35, 36, 37, 38, 39, 40, 41] Still many of these works have not been explored by researchers in the field, where they have already been published as part of my doctoral thesis in 1991 [26] and a few conference and journal papers, including the two that received the IEEE best paper awards listed above.

I have been working on extending the stochastic versions my learning theory for multi-layered neural networks, in order to produce much more efficient learning algorithms in deep belief networks (DBN). This work is quite useful in the training of our latest DBN-based (fused with HMM-based) speech recognition engine.

Another very important contribution to the field of neural network learning, which works well for deep belief networks is the better handling of local minima toward attaining a global direction. I handled this by creating a [dynamic architecture](#) [36] which helps reduce the chance of falling into local minima, while performing an [optimally conditioned quadratic optimization](#) [36]. This done by adding new neurons while going through the optimization process. This adaptive architecture creates experts for part of the optimization problem as it builds the hidden layers.

Learning Adaptive Control (1986 – 2002)

I formulated, *for the first time*, learning-adaptive control [32, 33] strategies and provided solutions based on Optimization Methods [29, 26, 31], adaptive ideas [28, 30], and new parameter estimation schemes [31]. This was the first learning-adaptive control strategy ever devised. It became a major part of my [doctoral thesis](#) [26] and became a breakthrough in the field. This was especially important for manufacturing processes. In fact that is why I was invited as the [guest editor of a special issue on learning and repetitive control](#) [51].

I also formulated a [continuous learning-adaptive control system](#) [28] with considerable improvement over existing techniques and theoretically sound control strategies based on my previous work on adaptive parameter estimation and optimization.

Another breakthrough came when for the first time, with the help of two colleagues, we extended my learning-adaptive control formulation the [continuous space in the space of repetitions, using a Hilbert Space mapping](#) [27, 30].

Pen-Based Music Editor (1995 – 1997)

I proposed the treatment of Online Common Music Notation using a Pen, while I was in the handwriting recognition group of IBM Research. This was in 1995 and was the beginning of a program called [Adventurous Systems and Software Research \(ASSR\)](#). The total grant allocation for the first year was \$250,000 which was allotted to be distributed among 5 finalists coming from hundreds of proposals which were given from different parts of the research division of IBM. My project received \$110,000 of the total and the rest (\$140,000) was distributed among the next 4 winning projects.

In the first year, I was able to create a complete editor using pen gestures, capable of inputting music common notation (CMN). This led to another \$250,000 being allotted to my project in the second year. Unfortunately, since IBM decided to dissolve the online handwriting recognition group, the project was hibernated along with the rest of the handwriting recognition group. Sensing this, I had already moved to the speech group and had started working on speaker recognition.

This project is very close to my heart since [I am also a musician](#). I have been spuriously approached by several individuals who had read an article about my project in the IBM Research Journal, written in 1996. They have shown interest in what I was doing. I believe I will try to revive that work if I get a chance.

Image Compression (1990 – 1991)

In September 1990, after defending my doctoral thesis, I was hired by the [Center for Telecommunications Research](#) for two different, but related subprojects. These were part of a project with [the Library of Congress](#), designed to digitize and preserve works of art, at the library. Since the pieces of art were generally at museums, lossless images of them needed to be digitized and transmitted over the slow lines of 1990 to the library of congress to be preserved. This needed to be done for thousands if not millions of works of art. Speed was essential. In addition, this needed to be done on DOS with all its limitations including 640 MB RAM and lack of a windowing system.

The first thing I did was to create a lossless compression scheme based on hybrid Huffman coding which I designed so that it would be able to minimize errors due to transmission. The algorithm used multiple codebooks with the usual variable length Huffman code. The codebooks related to the different color bins available in the art piece. This was done for true-color TIFF files with 32-bit color depth. I finished the algorithm and its implementation before leaving for IBM Research in February of 1991.

In addition, I wrote drivers in C for ultra-fast rendering of the images on a super-VGA screen which had just come out. Native drivers did not exist for it or were very slow. I implemented the driver by painting the 4 levels in fragments of memory and then dumping the contents onto the screen, using proper interrupts, etc.

Signature Compression and Recognition (1991 – Present)

Some of the experience from the image compression research, with the later work on handwriting recognition provided a natural mindset of compressing signatures to fit on the back of credit and debit cards. The magnetic strip on the back of a credit card can hold 54 bytes of data. It is a natural process to expect saving the signature information in whole on the back of a credit card. Also, developing dynamic features [42, 52, 43] and segmentation for handwriting recognition provided the means for coding the dynamics of signatures into a stream of features. These features are naturally possess the shortest code-length for describing online handwriting. This is because the **dynamic features**, that I derived, were based on parameters of differential equations approximating the segment of writing. Combining the two concepts, I was able to create a **very aggressive compression algorithm** [48] and **apparatus** [53] for describing all the dynamics and shape of a signature in less than 54 bytes, making it possible to store up to three dynamic signature templates on the magnetic strips on the back of credit or debit cards.

In addition, having these templates, I used statistical techniques similar to those used in speaker recognition to create useful signature verification and identification algorithms. This led to the creation of another **product at Recognition Technologies, Inc** [54].

Kinematics (1984 – 1986)

My concentration in my masters program at the mechanical engineering department of Columbia University was **Kinematics, Dynamics, and Control**. One of the greatest researchers in the field of Kinematics, known as the father of Kinematics, the late Prof. Ferdinand Freudenstein taught me three courses in the course of this degree. After completing my masters degree I approached Prof. Freudenstein and asked whether he had any research projects for me. He came up with the task of generalizing the equation of motion for the kinematic analysis of different 4-bar linkages into one single general equation that would cover the different types of 4-bar linkages, namely, “Planar, Spherical, Skew, and a special case of planar known as the plane slider-crank.” He said he was approached by the **Journal of Mechanism and Machine Theory** to provide original work as an invited paper for their anniversary edition. He asked whether I would be able to derive the equations and to program the equations of motions of each type of linkage with the new general equation as a comparison, before the deadline. I worked on the derivation and coding and produced my first paper [55], being a journal paper, with Prof. Freudenstein, a pleasure and a steep learning experience.

Image Processing for Fluid Mechanics Analysis (1984 – 1985)

Right at the beginning of my master program, I worked on digital image processing applied to fluid

mechanics for the analysis of lubricants' behavior in zero gravity – An experiment conducted in conjunction with the [first NASA Spacelab project – STS-9](#).

I designed and created a digitization platform using a sonic digitizer and wrote drivers for the digitizer in C, on an IBM PC platform. Digitized every frame of 24 frame-per-second film, taken of the spreading of fluids on different surfaces, with different viscosities, by the crew of the Columbia Shuttle in the Spacelab module. This data was used by Prof. Coda Pan to formulate the equations that describe the spreading characteristics of fluids on different smooth surfaces in zero gravity.

Doctoral Research Abstract (Jun. 1985 – Sep. 1990)

The problem of learning in general Feed-Forward Neural Networks has been formulated as a minimization problem. Several new algorithms have been developed for **learning in Feed-Forward Neural Networks** which are based on classical and modern **Quasi-Newton** minimization techniques. These methods achieve quadratic convergence by approximating the inverse of the Hessian of the objective function for neural network learning and thus providing Newton-like search directions. Benchmark simulation results have shown **two to three orders of magnitude improvement on the rate of learning** and **many orders of magnitude improvement in the accuracy** of these learning schemes when compared to the state-of-the-Art in Neural Network Learning which have previously been limited by steepest descent methods.

Due to the complexity of gradient evaluations even for a neural network of moderate size, learning algorithms requiring no gradient evaluations are called for. Learning algorithms have been developed based on **gradient-free** minimization techniques. These algorithms **require only the output of the network** to perform learning. Consequently, no analytical gradient expressions would have to be provided to the system, eliminating the need for a conventional computer to carry out the gradient evaluations. This makes the algorithms **independent of the architecture** of the neural network. As a result, negligible amount of computation and software/hardware is needed for learning. In addition, independence of the architecture implies that *if a neuron is damaged or a connection is severed, the learning could still be carried out to the full extent*. The result is a **more implementable network** which could *learn much more quickly and more independently*. Simulation results have shown a great reduction in the computation time and practicality of these algorithms.

The general **learning control of repetitive linear time-variant processes** (such as manufacturing processes) is formulated as a minimization problem. Many different minimization schemes have been evaluated for solving this problem. Finally, a **learning controller, which requires the theoretical minimum number of repetitions of the task for convergence**, has been developed based on a **Modified Generalized Secant method** of solving a set of linear equations. This controller has shown an **outstanding performance and robustness** when applied to the control of **nonlinear systems**. There is **no on-line computational burden** in the use of this controller.

Additionally, a **Recursive Learning Parameter Estimator** has been developed for usage in a **Learning Self-Tuning Regulator** applied to the *control of repetitive processes*. Simulation results show great improvement in the performance of conventional controllers when used with this

parameter estimator in the repetition domain. The on-line computational burden of this controller is only slightly higher than that of a PID controller.

References

- [1] IBM Corporation, “ViaVoice Speech Recognition,” Software, 1996–2003, <http://www-01.ibm.com/software/pervasive/viavoice.html>.
- [2] Homayoon Beigi, “RecoMadeEasy[®] Speaker Recognition,” Software, 2003–2014, <http://www.recotechnologies.com>.
- [3] Homayoon Beigi, *Fundamentals of Speaker Recognition*, Springer, New York, 2011, ISBN: 978-0-387-77591-3.
- [4] Homayoon Beigi, “Speaker Recognition: Advancements and Challenges,” in *New Trends and Developments in Biometrics*, Jucheng Yang and Shan Juan Xie, Eds. Intech Open Access Publisher, 2012, ISBN: 980-953-307-576-6 DOI: 10.5772/52023.
- [5] Homayoon Beigi, “Speaker Recognition,” in *Biometrics*, Jucheng Yang, Ed., pp. 3–28. Intech Open Access Publisher, Croatia, 2011, ISBN: 978-953-307-618-8.
- [6] Homayoon Beigi, *Speaker Recognition*, pp. 1232–1242, Springer US, 2011, ISBN: 978-1-4419-5906-5, DOI: 10.1007/978-1-4419-5906-5.
- [7] Homayoon Beigi, “RecoMadeEasy[®] Large Vocabulary Speech Recognition,” Software, 2010–2014, <http://www.recotechnologies.com>.
- [8] Homayoon Beigi, “RecoMadeEasy[®] Face Recognition,” Software, 2010–2014, <http://www.recotechnologies.com>.
- [9] Homayoon Beigi, “RecoMadeEasy[®] Automatic Language Proficiency Rating,” Software, 2006–2014, <http://www.recotechnologies.com>.
- [10] Homayoon S.M. Beigi and Stephane S. Maes, “Speaker, Channel and Environment Change Detection,” in *Proceedings of the World Congress on Automation (WAC1998)*, May 1998.
- [11] Homayoon S.M. Beigi, Stephane H. Maes, Upendra V. Chaudhari, and Jeffrey S. Sorensen, “A Hierarchical Approach to Large-Scale Speaker Recognition,” in *EuroSpeech 1999*, Sep 1999, vol. 5, pp. 2203–2206.
- [12] Homayoon Beigi, “RecoMadeEasy[®] AudioVisual Recognition,” Software, 2010–2014, <http://www.recotechnologies.com>.
- [13] Homayoon Beigi, “RecoMadeEasy[®] Access Control,” Software, 2011–2014.
- [14] Mahesh Viswanathan, Homayoon S.M. Beigi, and Fereydoun Maali, “Information Access using Speech, Speaker and Face Recognition,” in *IEEE International Conference on Multimedia and Expo (ICME2000)*, Jul 2000.
- [15] Luciana Balsamo, *Stastical Pattern Recognition Based Structural Damage Detection Strategies*, Columbia University, New York, 2014, Doctoral Thesis: School of Engineering and Applied Science.

- [16] Luciana Balsamo, Raimondo Betti, and Homayoon Beigi, “A Structural Health Monitoring Strategy using Cepstral Features,” *Journal of Sound and Vibration*, vol. 333, no. 19, pp. 4526–4542, January 2014.
- [17] Luciana Balsamo and Raimondo Betti and Homayoon Beigi, “Damage Detection using Large-Scale Covariance Matrix,” in *Proceedings of the 32nd IMAC, A Conference and Exposition on Structural Dynamics*, 2014, vol. 5.
- [18] Luciana Balsamo and Raimondo Betti and Homayoon Beigi, “Structural Damage Detection using Speaker Recognition Techniques,” in *11th International Conference on Structural Safety and Reliability (ICOSSAR-2013)*, Jun 2013.
- [19] Homayoon S. M. Beigi, Tetsu Fujisaki, William Modlin, and Ken Wenstrup, “A Post-Processing Error-Correction Scheme Using a Dictionary for On-Line Boxed and Runon Handwriting Recognition,” in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, Sep 1992, vol. II, pp. TM10.5.1–TM10.5.4.
- [20] Homayoon S.M. Beigi, “Character Prediction for On-line Handwriting Recognition,” in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, Sep 1992, vol. II, pp. TM10.3.1–TM10.3.4.
- [21] Homayoon S.M. Beigi and Tetsu Fujisaki, “A Character Level Predictive Language Model and Its Application to Handwriting Recognition,” in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, Sep 1992, vol. I, pp. WA1.27.1–WA1.27.4.
- [22] Homayoon S.M. Beigi and Tetsu Fujisaki, “A Flexible Template Language Model and its Application to Handwriting Recognition,” in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, Sep 1992, vol. I, pp. WA1.28.1–WA1.28.4.
- [23] Homayoon Beigi, “A Hybrid Approach to Automated Rating of Foreign Language Proficiency,” in *Where Humans Meet Machines – Innovative Solutions for Knotty Natural-Language problems*, Amy Neustein and Judith Markowitz, Eds., pp. 285–297. Springer, New York, 2013, ISBN: 980-953-307-576-6.
- [24] Homayoon Beigi, “Whether Computer Analyses Can Predict Human Ratings of Speaking,” Technical Report, Dec. 2008.
- [25] Homayoon Beigi, “Computer Rating of Oral Test Responses using Verbosity,” Technical Report, Dec. 2009.
- [26] Homayoon S. M. Beigi, *Neural Network Learning and Learning Control Through Optimization Techniques*, Columbia University, New York, 1991, Doctoral Thesis: School of Engineering and Applied Science.
- [27] Konstantin E. Avrachenkov, Homayoon S.M. Beigi, and Richard W. Longman, “Updating Procedures for Iterative Learning Control in Hilbert Space,” *Intelligent Automation and Soft Computing Journal*, vol. 8, no. 2, 2002, Special Issue on Learning and Repetitive Control.
- [28] Homayoon S.M. Beigi, “Adaptive and Learning-Adaptive Control Techniques based on an Extension of the Generalized Secant Method,” *Intelligent Automation and Soft Computing Journal*, vol. 3, no. 2, pp. 171–184, 1997.

- [29] C. James Li, Homayoon S.M. Beigi, Shengyi Li, and Jiancheng Liang, "Nonlinear Piezo-actuator Control by Learning Self-Tuning Regulator," *ASME Transactions, Journal of Dynamic Systems, Measurement, and Control*, vol. 115, no. 4, pp. 720–723, 1993.
- [30] Konstantin E. Avrachenkov, Homayoon S.M. Beigi, and Richard W. Longman, "Operator-Updating Procedures for Quasi-Newton Iterative Learning Control in Hilbert Space," in *IEEE Conference on Decision and Control (CDC99)*, Dec 1999, vol. 1, pp. 276–280, Invited Paper.
- [31] Homayoon S.M. Beigi, C. James Li, and R.W. Longman, "Learning Control Based on Generalized Secant Methods and Other Numerical Optimization Methods," in *Sensors, Controls, and Quality Issues in Manufacturing, the ASME Winter Annual Meeting*, Dec 1991, vol. 55, pp. 163–175.
- [32] C. James Li, Homayoon S.M. Beigi, Shengyi Li, and Jiancheng Liang, "Self-tuning Regulator with Learning Parameter Estimation," in *Intelligent Control Systems, Dynamic Systems and Control, the ASME Winter Annual Meeting*, Nov 1990, vol. 26, pp. 1–6.
- [33] Richard W. Longman, Homayoon S.M. Beigi, and C. James Li, "Learning Control by Numerical Optimization Methods," in *Proceedings of the Modeling and Simulation Conference, Control, Robotics, Systems and Neural Networks*, May 1989, vol. 20, pp. 1877–1882.
- [34] Homayoon S.M. Beigi and C. James Li, "Learning Algorithms for Feedforward Neural Networks Based on Classical and Initial-Scaling Quasi-Newton Methods," *ISMM Journal of Microcomputer Applications*, vol. 14, no. 2, pp. 41–52, 1995.
- [35] Homayoon S.M. Beigi and C. James Li, "Learning Algorithms for Neural Networks Based on Quasi-Newton Methods with Self-Scaling," *ASME Transactions, Journal of Dynamic Systems, Measurement, and Control*, vol. 115, no. 1, pp. 38–43, Mar 1993.
- [36] Homayoon Beigi, "Neural Network Learning Through Optimally Conditioned Quadratically Convergent Methods Requiring NO LINE SEARCH," in *IEEE-36th Midwest Symposium on Circuits and Systems*, Aug 1993, vol. 1, pp. 109–112.
- [37] Homayoon S. M. Beigi, "Neural Network Learning Through Optimization Techniques," in *Neural Networks Internal Technological Liaison (ITL1992)*, May 1992.
- [38] Homayoon S.M. Beigi and C. James Li, "Learning algorithms for neural networks based on quasi-newton methods with self-scaling," in *Intelligent Control Systems, Dynamic Systems and Control, the ASME Winter Annual Meeting*, Nov 1990, vol. 23, pp. 23–28.
- [39] Homayoon S.M. Beigi and C. James Li, "New Neural Network Learning Based on Gradient-Free Optimization Methods," in *The IEEE 1990 Long Island Student Conference on Neural Networks*, Apr 1990, pp. 9–12, Recipient of Best Paper Award.
- [40] Homayoon S.M. Beigi and C. James Li, "Neural Network Learning Based on Quasi-Newton Methods with Initial Scaling of Inverse Hessian Approximate," in *The IEEE 1990 Long Island Student Conference on Neural Networks*, Apr 1990, pp. 49–52, Recipient of Best Paper Award.
- [41] Homayoon S.M. Beigi and C. James Li, "A New Set of Learning Algorithms for Neural Networks," in *Proc. of the ISMM International Symposium, Computer Applications in Design, Simulation and Analysis*, Mar 1990, pp. 277–280.

- [42] Homayoon S.M. Beigi, “Pre-Processing the Dynamics of On-Line Handwriting Data, Feature Extraction and Recognition,” in *Progress in Handwriting Recognition*, A.C. Downton and S. Impedovo, Eds., pp. 191–198. World Scientific Publishers, New Jersey, 1997, ISBN: 981-02-3084-2.
- [43] Homayoon S.M. Beigi, “Pre-Processing the Dynamics of On-Line Handwriting, Feature Extraction and Recognition,” in *Proceedings of the World Congress on Automation (WAC1996*, Sep 1996, pp. 255–258.
- [44] T. Fujisaki, H.S.M. Beigi, C.C. Tappert, M. Ukelson, and C.G. Wolf, “Online Recognition of Unconstrained Handprinting: A Stroke-based System and Its Evaluation,” in *From Pixels to Features III: Frontiers in Handwriting*, S. Impedovo and J.C. Simon, Eds., pp. 297–312. North Holland, Amsterdam, 1992, ISBN: 0-444-89665-1.
- [45] Krishna S. Nathan, Homayoon S.M. Beigi, Gregory J. Clary, Jayashree Subrahmonia, and Hiroshi Maruyama, “Real-Time On-Line Unconstrained Handwriting Recognition using Statistical Methods,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP95)*, May 1995, vol. 4, pp. 2619–2622.
- [46] Homayoon S.M. Beigi, Krishna S. Nathan, and Jayashree Subrahmonia, “On-Line Unconstrained Handwriting Recognition Based on Probabilistic Techniques,” in *Iranian Conference on Electrical Engineering (ICEE95)*, May 1995.
- [47] Homayoon S.M. Beigi, Krishna Nathan, Gregory J. Clary, and Jayashree Subrahmonia, “Challenges of Handwriting Recognition in Farsi, Arabic and Other Languages with Similar Writing Styles – An On-line Digit Recognizer,” in *Proceedings of the 2nd Annual Conference on Technological Advancements in Developing Countries*, Jul 1994.
- [48] Homayoon S.M. Beigi, “Aggressive Compression of the Dynamics of Handwriting and Signature Signals,” in *IEEE International Conference on Multimedia and Expo (ICME2004)*, Jun 2004, vol. 2, pp. 1447–1450.
- [49] Homayoon S.M. Beigi, Krishna Nathan, Gregory J. Clary, and Jayashree Subrahmonia, “Size Normalization in Online Unconstrained Handwriting Recognition,” in *The IEEE International Conference on Image Processing (ICIP94)*, Nov 1994, vol. I, pp. 169–172.
- [50] IBM Corporation, “ThinkWrite Online Unconstrained Handwriting Recognition,” Software, 1991–1996, <http://www.research.ibm.com/electricInk/glossary.html>.
- [51] Homayoon S.M. Beigi, “Special Issue on Learning and Repetitive Control,” *The Intelligent Automation and Soft Computing Journal*, vol. 8, no. 2, 2002.
- [52] Homayoon S.M. Beigi, “Processing, Modeling and Parameter Estimation of the Dynamic On-Line Handwriting Signal,” in *Proceedings of the World Congress on Automation (WAC1996*, May 1996.
- [53] Homayoon Beigi, “7,474,770 b2,” Jan 2009, Method and Apparatus for Aggressive Compression, Storage and Verification of the Dynamics of Handwritten Signature Signals, <http://www.google.com/patents/US7474770>.
- [54] Homayoon Beigi, “RecoMadeEasy[®] Signature Compression,” Software, 2005–2014.

- [55] Homayoon S. M. Beigi Ferdinand Freudenstein, “On a Computationally Efficient Microcomputer Kinematic Analysis of the Basic Linkage Mechanisms,” *Journal of Mechanism and Machine Theory (formerly Journal of Mechanisms)*, vol. 21, no. 6, pp. 467–472, 1986.

[Return to Table of Contents](#)

Standards Development

2003–2010	<i>VoiceXML 3.0</i> As an active liaison and driving force, helped define the speaker recognition functionality of VoiceXML 3.0.
2003–2010	<i>Standard Audio Format Encapsulation (SAFE)</i> Personally defined a new standard for audio format encapsulation (SAFE) which has been used by several standards committes for handling interoperability among government organizations and the private industry. This format (SAFE) was adopted by ANSI/INCITS and ISO without any modification. ANSI/INCITS As an active liaison and driving force of the ANSI/INCITS Standards Development for Biometric Formats, proposed SAFE to be considered as an ANSI standard. It was presented and underwent a public review. It has been incorporated into an American National Standard Institute (ANSI) standard for audio format of raw data interchange for use in speaker recognition without any opposition. ISO As an active liaison for the U.S. delegation of the International Organization for Standardisation (ISO), proposed SAFE to the committe. The SAFE proposal has been incorporated in the speaker recognition data interchange draft standards by the ISO/IEC JTC1/SC37 project 19794-13 (voice data).
1996–1999	<i>Speaker Verification Application Programming Interface (SVAPI)</i> Helped define and test SVAPI 1.0 and 2.0 as an active member and driving force representing IBM Research in the SVAPI consortium including IBM, Novell, Dialogic, ITT Industries, Motorola, Texas Instruments, T-NETIX Inc. and some U.S. Government organizations.
1996 – 1997	<i>SVAPI 1.0</i> Definition and Implementation Implemented the SVAPI 1.0 standrd interface for the IBM Speaker Verification engine and identified missing functionality to be included in version 2.0.
1997 – 1999	<i>SVAPI 2.0</i> Definition and Implementation Worked to define and implement the new functionality need to do identifcaion and classification.

[Return to Table of Contents](#)

Homayoon Beigi

Testimonies and Depositions

- *MModal Inc. et al v. Nuance Communications, Inc.*, IPR2019-00210, filed expert declaration, was deposed twice.
- *MModal Inc. et al v. Nuance Communications, Inc.*, IPR2019-00306, filed expert declaration, was deposed.
- *MModal Inc. et al v. Nuance Communications, Inc.*, IPR2018-01355, filed expert declaration, was deposed twice.
- *MModal Inc. et al v. Nuance Communications, Inc.*, IPR2018-01291, filed expert declaration.
- *MModal Inc. et al v. Nuance Communications, Inc.*, IPR2018-01337, filed expert declaration.
- *Global Tel*Link Corporation v. Securus Technologies, Inc.*, IPR2016-01220, filed expert declaration, was deposed.
- *Global Tel*Link Corporation v. Securus Technologies, Inc.*, IPR2015-01225, filed expert declaration, was deposed.
- *Global Tel*Link Corporation v. Securus Technologies, Inc.*, IPR2015-01220, filed expert declaration, was deposed.
- *Securus Technologies, Inc. v. Global Tel*Link Corporation*, IPR2015-00155, served as an expert consultant.
- US 9,113,131 B1, prepared expert declaration, was settled before filing an IPR petition.