Machine Test: Send a request to the site with a device that supports relative measures correctly and check the values of `font-size` are not absolute or pixels.

5.4.9 Style Sheets

[STYLE_SHEETS_USE] Use style sheets to control layout and presentation, unless the device is known not to support them.

[STYLE_SHEETS_SUPPORT] Organize documents so that if necessary they may be read without style sheets.

[STYLE_SHEETS_SIZE] Keep style sheets small.

5.4.9.1 What it means

Style information may be contained in an externally linked style sheet or, in HTML, may be contained either in a style element or in a style attribute on specific elements.

Mobile devices offer varying support for style sheets. Some provide full implementations, including caching of external style sheets; some do not cache external style sheets; some do not support the `style` element; some implementations do not support more than one style sheet and some do not support style sheets at all.

If style sheets are turned off or not supported, content will be rendered in document order, so it is important that the content makes sense when read in document order.

5.4.9.2 How to do it

It is preferable to share style information between pages, but if the device does not support caching of style sheets then this approach would result in the same style sheet being retrieved for each page. Consequently, in order of preference: if the device caches style sheets, put style information in a single external style sheet (see also 5.2.9 Externally Linked Resources); if the device supports the `style` element, use it; otherwise use an external style sheet.

Optimize style information so that only styles that are used are included.

When creating style sheets, take advantage of the CSS media types (these may be used both in the CSS `@media` rule and in the `media` attribute of the `link` element) to specify styles that apply to handheld rendering. The CSS Media types that apply are "handheld" and "all". If handheld rendering is not specified, browsers may download other style sheets even if they are identified as applicable to non-handheld rendering

5.4.9.3 What to test

STYLE_SHEETS_USE Machine Test: Send a request to the site with a device that supports CSS and check that style sheets are used and that the page does not use formatting tags (e.g. `font`).

STYLE_SHEETS_SUPPORT Human Test: Disable style sheets and check that the page is still readable.

STYLE_SHEETS_SIZE Machine Test: Check that the elements in a style sheet are used in at least one of the pages that reference it.

5.4.9.4 References

This relates to WCAG 3.3 and 6.1

5.4.10 Minimize

[MINIMIZE] Use terse, efficient markup.

5.4.10.1 What it means

Content which is marked up in languages such as XML can often be made smaller while preserving exactly the same semantics merely by removal of redundant white space (i.e. spaces and new lines).

Marking fonts, colors and other stylistic effects in-line can cause considerably larger page sizes when compared with using logical markup, and use of the HTML `class` attribute for application of style (see also 5.4.9 Style Sheets).

5.4.10.2 How to do it

While it is not intended that authors should create their content in a single line to remove white space altogether, it is suggested that authors should not contribute to page weight by introducing unnecessary white space. Note that "pretty printing" (the formatting of markup with indentation) can generate large amounts of white space and hence add to page weight.

If "pretty printing" is an important part of the authoring process, then try to arrange that redundant white space is stripped when serving a page.

Even though some network proxies strip white space that they think is redundant, not all do so, so it is not best practice to rely upon this behavior.

Use of structural markup (see 5.4.3 Structural Elements) contributes to minimizing the size of the markup on a page, as does centralizing the style descriptions using CSS [CSS].

5.4.10.3 What to test

Machine Test: Count the number of non-significant white space characters in the document.

5.4.11 Content Types

[CONTENT_FORMAT_SUPPORT] Send content in a format that is known to be supported by the device.

[CONTENT_FORMAT_PREFERRED] Where possible, send content in a preferred format.

5.4.11.1 What it means

Transferring content that a device cannot display wastes users' time and money. A device may express a preference for formats. In this case it is good practice to respect the device's preference, as it may have a fuller implementation of those formats.

5.4.11.2 How to do it

To determine what formats a device supports, Web sites may use any combination of device profile information such as the HTTP User-Agent header, HTTP Accept headers and UAProf.

There are problems with using any one approach to the exclusion of the others. Some issues that have been noted by the BPWG in this context are:

- Some devices do not supply accept headers;
- Some devices mis-state their capabilities;
- Some operator gateways supplement the accept headers to include formats that they adapt;
- User agent headers do not always uniquely identify the device;
- UAProf information may not be available or may be incomplete.

5.4.11.3 What to test

CONTENT_FORMAT_SUPPORT Machine Test: Check MIME types of content with various devices.

CONTENT_FORMAT_PREFERRED Machine Test: Check MIME types of content with various devices and check that the preferred format is sent or that the format is compatible with the Default Delivery Context.

5.4.12 Character Encoding

[CHARACTER_ENCODING_SUPPORT] Ensure that content is encoded using a character encoding that is known to be supported by the device.

[CHARACTER_ENCODING_USE] Indicate in the response the character encoding being used.

5.4.12.1 What it means

As in the previous section, content should not be sent to a device if it can not use it.

5.4.12.2 How to do it

The supported character encodings for a device may be obtained either from a device profile or by examining the value of the HTTP Accept-Charset header.

The character encoding being used in a response may be indicated using the HTTP Content-Type header.

```
Example:
Content-Type: text/html; charset=utf-8
```

Additionally for XML [XML] documents the character encoding may be indicated in the encoding declaration, although this will generally be ignored if an HTTP Content-Type header is present.

```
Example:
<?xml version="1.0" encoding="UTF-8"?>
```

Encoding of the content to a desired character encoding is dependent on the authoring tools being used, Web server configuration and the server side scripting technology being used (if any). For a discussion of this see [CHARSET1] and [CHARSET2].

Unicode is a good choice for representing content when served in multiple languages. The amount of bandwidth required to transmit content can vary significantly depending on the character encoding used. Text consisting principally of characters from the Latin alphabet will encode more efficiently in UTF-8, whereas text consisting principally of characters from ideographic scripts will encode more efficiently in UTF-16. When choosing a character encoding, consider the efficiency of the available encodings.

Since the Default Delivery Context specifies use only of UTF-8, all applications should support UTF-8.

5.4.12.3 What to test

Machine Test: Check that the encoding is declared in some way and is supported. The content type may be declared in one or more of the following ways: The Content-Type HTTP header, the XML declaration for XML-based content, the CSS @charset rules for CSS, the Content-Type Meta element for HTML content.

5.4.12.4 References

See [XML] Character Encoding in Entities for a discussion of character encoding in XML documents.

5.4.13 Error Messages

[ERROR_MESSAGES] Provide informative error messages and a means of navigating away from an error message back to useful information.

5.4.13.1 What it means

It is inevitable that, on occasions, a mobile user will not be successful in accessing the content or information they sought. Providing easy navigation away from the error is particularly important in the mobile arena, where browsers may not have an easy-to-find "back" button, where contextualization is frequently difficult and where re-entry of URIs as a means of error recovery is particularly difficult.

It is noted that errors due to networking, connection and some kinds of mistyping of URIs are not within the control of the content provider, which has no way to influence how such errors are presented to the user. However, where errors are within the control of the content provider the user should be provided with clear information regarding the fault they have experienced. This should help them to understand whether the fault was temporary or permanent, whether they should retry the attempt to access the content and how they may be able to escalate the problem.

It should also be possible for the user to escape from the error condition. They should either be able to return to the page they were on prior to the error, or to be able to move onwards to a convenient part of the service from where they can retry or alter the transaction they were attempting.

5.4.13.2 How to do it

It is noted that many Web servers provide a default error page, especially in the event of a request for a non-existent page (404) or an internal error (500). Where possible (see [TOMCAT], [APACHE] and [IIS]), applications should trap all error conditions by overriding the default pages if necessary, and handle them in a user-friendly, and graceful, way.

Error messages should be provided in the same language as the application that was being used. They should be clear and concise, adhering to the same Best Practices as the rest of the application. They should be provided in a format that the device can handle.

The error message should detail whether the issue is likely to be temporary or permanent, whether the user may be able to solve the issue themselves (for example, by changing input data or a handset setting), or whether it is an issue that can be escalated to the content provider or network operator. In the latter case, contact details, such as an SMS address or a support line number, might be appropriate.

The error message should provide one or more of the following navigational constructs:

1. A "back" link to return to the previous page (particularly for devices that do not have an easy to find back button);
2. A "retry" link to attempt the relevant part of the transaction again (note that this may not be equivalent to a page "refresh");
3. A "home" link to allow the user to return to the main part of the application.

The error message can provide an error code to be used for diagnosis of the issue. However, the use of an error code is not a substitute for a human-readable message. While some users may understand "404" to mean "page cannot be found", this must not be assumed to be true for all users.

5.4.13.3 What to test

Enter an extraneous URI, known not to represent an actual resource on the site, and check that a HTTP 404 error response is accompanied by a page whose markup is appropriate for the requesting device, or the default context.

Human Test: Check that the page returned contains an explanation of the error and appropriate corrective actions, without assuming any technical knowledge on the part of the end user.

5.4.14 Cookies

[COOKIES] Do not rely on cookies being available.

5.4.14.1 What it means

Cookies are frequently used to carry out session management, to identify users and to store user preferences. Many mobile devices do not implement cookies or offer only an incomplete implementation. In addition, some gateways strip cookies and others simulate cookies on behalf of mobile devices.

5.4.14.2 How to do it

Test that cookies are supported by the device on its current access path. If they are not supported, use URI decoration for session management, being careful not to exceed the device's maximum length for such strings. Some gateways provide user identification without setting cookies.

5.4.14.3 What to test

Machine Test: Check that an alternative to cookies is used for session management when they are not available.

5.4.15 Cache Headers

[CACHING] Provide caching information in HTTP responses.

5.4.15.1 What it means

Limited bandwidth and high latency can reduce the usability of Web sites on mobile devices. Using caching information effectively can reduce the need to reload data such as style sheets, images and pages, thus improving performance and reducing cost of use. It can also prevent the reuse of content where this is not appropriate, for example content that is adapted for one device should not be re-used by different devices. Devices and network caches are both affected by caching information.

5.4.15.2 How to do it

Set expiry times in a way that is appropriate to your application. Consider using `Cache-Control: public` to allow sharing of pages between devices, `Cache-Control: private` to allow re-use but only by the requesting device and `Cache-Control: nocache` to prevent caching.

The HTTP 1.1 specification [HTTP1.1] and Techniques document [Techniques] contain discussions of caching.

5.4.15.3 What to test

Machine Test: Check for the presence of cache headers on the HTTP response.

5.4.15.4 References

Section 13 Caching in HTTP of [HTTP1.1] discusses caching.

5.4.16 Fonts

[FONTS] Do not rely on support of font related styling.

5.4.16.1 What it means

Mobile devices often have few fonts and limited support for font sizes and effects (bold, italic etc.) As a result of this, the use of font size, face or effect, for example to highlight an answer or a stressed word, may not achieve the desired effect. See also 5.4.3 Structural Elements.

5.4.16.2 How to do it

For the Default Delivery Context do not use font related styling.

5.4.16.3 What to test

Machine Test: Check for the presence of font related styling in an environment that does not support it.

Human Test: If present, ensure that the author's intentions remain clear.

5.5 User Input

This section contains statements relating to user input. This is typically more restrictive on mobile devices than on desktop computers (and often a lot more restrictive). For example, mobile devices may lack pointing devices and often do not have a standard keyboard for text entry.

5.5.1 Input

[MINIMIZE_KEYSTROKES] Keep the number of keystrokes to a minimum.

[AVOID_FREE_TEXT] Avoid free text entry where possible.

[PROVIDE_DEFAULTS] Provide pre-selected default values where possible.

[DEFAULT_INPUT_MODE] Specify a default text entry mode, language and/or input format, if the device is known to support it.

5.5.1.1 What it means

Given the typical input limitations of a mobile device, the interface must as far as possible minimize user input. Where possible, use selection lists, radio buttons and other controls that do not require typing.

Some markup languages allow the specification of an input mode, which is particularly useful in cases where user input is to be restricted, for example to numeric only. It is anticipated that XHTML-Basic [XHTML-Basic] will support this functionality in the future.

5.5.1.2 How to do it

There are a number of techniques available for this, including:

- Where possible use previous entries as defaults.
- Make it possible to select items using navigation keys and/or numeric input.

5.5.1.3 What to test

AVOID_FREE_TEXT Machine Test: Check whether `input type="text"` and `textarea` are used.

Human Test: If one of them is used, check whether it can be replaced by a pre-determined entry.

PROVIDE_DEFAULTS Machine Test: Check if there is a pre-selected value in controls (selected or checked attribute set).

Human Test: If not, check if there could be sensible pre-selection in the context (e.g. most common choice).

DEFAULT_INPUT_MODE Machine Test: Send a request with a device known to support the `inputmode` attribute and if the response is in a language that supports this attribute, check that it is present on `input type="text"` and `textarea` elements.

5.5.1.4 References

This relates to WCAG 10.4.

5.5.2 Tab Order

[TAB_ORDER] Create a logical order through links, form controls and objects.

5.5.2.1 What it means

It is important that as the user navigates through the page the various fields and objects are presented in a logical order, especially as many of them will not be visible at the same time as the focus item.

5.5.2.2 How to do it

Use document order to control layout and tab order.

5.5.2.3 What to test

Machine Test: Check that there are no `tabindex` attributes or layout effects that affect the order of presentation.

If there are `tabindex` attributes check that all controls have a tab index and that they are used consistently.

Human Test: If there are either `tabindex` attributes or layout effects that might affect the order of presentation, then check that the order is usable.

5.5.3 Labels for Form Controls

[CONTROL_LABELLING] Label all form controls appropriately and explicitly associate labels with form controls.

[CONTROL_POSITION] Position labels so they lay out properly in relation to the form controls they refer to.

5.5.3.1 What it means

This means use the `label` element in HTML, or its equivalent in other languages. Make sure that where the label goes is consistent and close to the form control so re-flowing or adapting the content intelligently will always recognize label controls and keep them together.

5.5.3.2 What to test

Machine Test: Check if the `label` element is used in forms.

Human Test: Check whether the labels are properly positioned with regard to the controls.

5.5.3.3 References

This relates to WCAG 10.2 and 12.4.

6 Conformance and mobileOK

The Best Practice statements are intended to be capable of having conformance statements constructed around them in support of the mobileOK trustmark and for other purposes. Work on the mobileOK trustmark will develop specific recommended requirements for a trustmark, which will be based on some profile, or subset, of the Statements in this document.

As such, the mobileOK trustmark will serve as the main conformance claim for the Best Practices document.

All of the Best Practice statements have a fragment identifier to allow formal reference to them and allow the construction of compliance claims that refer to them.

6.1 Classes of Products

This specification applies to one class of product: content delivered to a mobile device, including the metadata transferred as part of the delivery protocol.

6.2 Extensibility

This specification may be compatible with other specifications that describe a different set of requirements for content, insofar as such requirements do not conflict with the current specification.

A Sources (Non-Normative)

The Best Practice statements have been assembled by the BPWG from a number of sources. Primary among those are:

- Various BPWG group meetings and discussions
- WCAG Guidelines 1.0 [WCAG]
- iMode Guidelines [iMode]
- Opera's "Making Small Devices Look Great" [Opera]
- Openwave Guidelines [OpenWave]
- Nokia's Series 60 XHTML-MP Guidelines [Nokia-MP]
- Browsing on Mobile Phones, Nokia [Nokia-VR]
- Little Spring Design [LSD]

While the Best Practice statements have mainly been assembled by secondary research, the sources for that research have in many cases been assembled from primary research. In addition, group members' contributions are to some extent informed by primary research carried out by their company.

B Related Reading (Non-Normative)

Readers interested in the topic of this document will find a variety of other publications of interest. As noted in the Scope paragraph above, topics such as internationalization and accessibility have been addressed separately by the W3C and have not been covered here.

The Character Model for the World Wide Web and other materials prepared by the W3C Internationalization (i18n) Activity cover important interoperability drivers for content prepared for the One Web and the mobile-services arena.

The Web Accessibility Initiative has prepared a variety of Guidelines and Techniques that likewise bear on the preparation and processing of content in and for the Web.

Section 3.2 Background to Adaptation above introduced the idea of content adaptation. Readers who contemplate implementing server-side adaptation will be interested in the ongoing work of the Device Independence Activity.

C Acknowledgements (Non-Normative)

The editors would like to thank members of the BPWG for contributions of various kinds. The editors would also like to thank contributors to the public list, and contributors of Last Call comments whose comments have been taken into account in the creation of this document.

The editors acknowledge significant written contributions from:

- Greg Aaron, Afilias Limited
- Daniel Appelquist, Vodafone (*Mobile Web Best Practices Working Group Chair*)
- Phil Archer, ICRA
- Rotan Hanrahan, MobileAware
- Dominique Hazaël-Massieux, W3C/ERCIM (*W3C Team Contact*)
- Philipp Hoschka, W3C/ERCIM (*W3C Team Contact*)
- Richard T. Kennedy, The Boeing Company
- Rhys Lewis, Volantis Systems Ltd
- Luca Passani, Openwave Systems Inc.
- Giles Payne, NTT DoCoMo
- James Pearce, Argo Interactive Ltd
- Kai-Dietrich Scheppe, Deutsche Telekom AG, T-Com, Geschäftsbereich T-Online
- Andrea Trasatti, W3C Invited Expert
- Paul Walsh, Segala M Test

D References (Non-Normative)

D.1 MWI References

Scope
Scope of Mobile Web Best Practices, Phil Archer, Ed Mitukiewicz, Editors, W3C Working Group Note, 20 December 2005 (See http://www.w3.org/TR/2005/NOTE-mobile-bp-scope-20051220/)

Techniques
Mobile Web Techniques for Best Practices [in development] (See http://www.w3.org/2005/MWI/BPWG/techs/TechniquesIntro)

mobileOK
mobileOK Basic Tests 1.0, Sean Owen, Jo Rabin (eds), W3C Working Draft, 10 June 2008 (See http://www.w3.org/TR/2008/WD-mobileOK-basic10-tests-20080610/)

D.2 Sources

WCAG
Web Content Accessibility Guidelines 1.0, W Chisholm, I. Jacobs, G Vanderheiden, Editors, W3C Recommendation, 5 May 1999. (See http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/)

WAIGlossary
WAI (Printable) Glossary, Katie Haritos-Shea, Charles McCathieNevile, Internal Working Draft, 1 March 2003 (See http://www.w3.org/WAI/GL/Glossary/)

iMode
iMode (See http://www.iMode.nl/pdf/download/How_to_create_an_i-mode_site_1_3.pdf)

Opera
Opera's Making Small Devices Look Great (See http://my.opera.com/community/dev/device/)

OpenWave
Openwave (See http://developer.openwave.com/dvl/support/documentation/guides_and_references/best_practices_in_xhtml_design/index.htm)

Nokia-MP
Nokia Guidelines for XHTML-MP on Series 60 (See http://sw.nokia.com/id/4f7b6805-47d7-4914-885c-6ef2b487adf6/Series_60_Platform_Designing_XHTML_MP_Content_v1_4_en.pdf)

Nokia-VR
Browsing on Mobile Phones, paper by Virpi Roto, Nokia (See http://www.research.att.com/~rjana/WF12_Paper1.pdf)

LSD
Little Spring Design (See http://patterns.littlespringsdesign.com)

D.3 Device Independence

DIP
Device Independence Principles, R. Gimson, Editor, W3C Working Group Note, 1 September 2003 (See http://www.w3.org/TR/2003/NOTE-di-princ-20030901/)

DCODI
Delivery Context Overview for Device Independence, , R. Gimson, R. Lewis, S. Sathish, Editors, W3C Working Group Note, 20 March 2006 (See http://www.w3.org/TR/2006/NOTE-di-dco-20060320/)

DIGLOSS
Glossary of Terms for Device Independence, R. Lewis, Editor, W3C Working Draft (work in progress), 18 January 2005 (See http://www.w3.org/TR/2005/WD-di-gloss-20050118/)

D.4 Web, Protocols and Languages

WebArch
Architecture of the World Wide Web, Volume One, N. Walsh, I. Jacobs, Editors, W3C Recommendation, 15 December 2004 (See http://www.w3.org/TR/2004/REC-webarch-20041215/)

XML
Extensible Markup Language (XML) 1.0 (Fourth Edition), Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, François Yergeau, Editors, W3C Recommendation 16 August 2006, edited in place 29 September 2006 (See http://www.w3.org/TR/2006/REC-xml-20060816)

XHTML-Basic
XHTML™ Basic 1.1, Shane McCarron, Masayasu Ishikawa, Editors, W3C Recommendation, 29 July 2008 (See http://www.w3.org/TR/2008/REC-xhtml-basic-20080729/)

CSS
Cascading Style Sheets (CSS1) Level 1 Specification, Håkon Wium Lie, Bert Bos, Editors, W3C Recommendation, 11 Jan 1999 (See http://www.w3.org/TR/1999/REC-CSS1-19990111)

CSS2
Cascading Style Sheets, level 2 CSS2 Specification, Bert Bos, Håkon Wium Lie, Chris Lilley, Ian Jacobs, Editors, W3C Recommendation, 12 May 1998 (See http://www.w3.org/TR/1998/REC-CSS2-19980512/)

HTTP1.0
Hypertext Transfer Protocol -- HTTP/1.0 Request for Comments: 1945, T. Berners-Lee, R. Fielding, H. Frystyk, May 1996 (See http://www.w3.org/Protocols/rfc1945/rfc1945)

HTTP1.1
Hypertext Transfer Protocol -- HTTP/1.1 Request for Comments: 2616, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999 (See http://www.w3.org/Protocols/rfc2616/rfc2616.html)

UTF-8
UTF-8, a transformation format of ISO 10646 Request for Comments: 3629, F. Yergeau, November 2003 (See http://www.ietf.org/rfc/rfc3629.txt)

CHARSET1
Tutorial: Character sets & encodings in XHTML, HTML and CSS (See http://www.w3.org/International/tutorials/tutorial-char-enc/)

CHARSET2
FAQ: Setting encoding in Web authoring applications (See http://www.w3.org/International/questions/qa-setting-encoding-in-applications)

D.5 Other References

UAPROF
Open Mobile Alliance OMA-TS-UAProf-V2_0-20060206-A User Agent Profile Approved Version 2.0 – 06 Feb 2006 (See http://www.openmobilealliance.org/technical/release_program/docs/UAProf/V2_0-20060206-A/OMA-TS-UAProf-V2_0-20060206-A.pdf)

TOMCAT
Tomcat FAQ How do I get a customized error page? (See http://tomcat.apache.org/faq/misc.html#q6)

APACHE
Apache Core Features ErrorDocument directive (See http://httpd.apache.org/docs/1.3/mod/core.html#errordocument)

IIS
IIS Custom HTTP Error Messages (See http://msdn.microsoft.com/library/default.asp?url=/library/en-us/iissdk/html/ee7a8c53-f9bc-4cd4-b954-e32066105cf1.asp)

T-MOB
T-Mobile International - Position Statement for W3C Mobile Web Initiative Version: 1.0 18 Dec 2004 (See http://www.w3.org/2004/10/MWIWS-papers/W3C_Mobile_Web_Initiative_-_T-Mobile_Position_Statement-final.pdf)

MF
Study by Segala M Test on Scrolling vs. Pagination (See http://www.mobilefriendly.org)

https://web.archive.org/web/20110704113245/http://www.casualgameguides.com/games/guide/game_guide.cfm/Farm_Mania-Walkthrough/Cheats-1/game_id-3628/

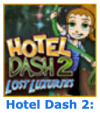
INTERNET ARCHIVE WayBackMachine
http://www.casualgameguides.com/games/guide/game_guide.cfm/Farm_Mania-Walkthrough/Cheats-1/game_id-3628/ Go
9 captures 4 Jul 2011 - 27 Sep 20
JUN JUL 04 AUG 2010 2011 2012 About this capture

Related Games

People who liked this game also enjoyed these games


Island Tribe 2


The Timebuilders: Caveman's Prophecy


Hotel Dash 2: Lost Luxuries


Chocolatier 3: Decadence by Design


Youda Farmer 3: Seasons

Top Games In This Franchise


Farm Mania: Hot Vacation


Farm Mania 2


Farm Mania

Walkthrough and Cheats

Farm Mania Walkthrough and Cheats

[courtesy of Jasper]

Farm Mania

As you'll learn from the comic strip style story, you are Anna, a young lady who gives up a summer partying to become the girl next door helping out her Grandpa on the farm. Grow and sell plants, animals and goods through 60 levels to help get your Grandpa's farm out of debt.

You will not be bored for long when you play this game. By the time level 6 hits, you're knee deep in...weeds. Where did you think I was going with that one? Arcade mode is timed, casual mode is not. The goals are the same in either mode, but if you want a challenge, arcade mode is the way to go. Have fun, remember it's only a game and if you aren't thankful for not working on a farm, you will be when you are done playing.

Hints and tips:

- If playing Arcade Mode, it's a good idea to keep up your Casual Mode levels. If you get stuck on an Arcade Mode, you can take your time in Casual Mode until you figure out what you need to do. Unfortunately unlocking a new level in one mode does not unlock it in the other.
- The faster you finish a level, the more bonus cash you earn.
- You can sometimes earn more cash by selling more goods instead of going for the bonus points. Take a look at the clock (top left) to see how much time is left. If you have time, sell some extra goods and still get a gold star.
- You can send the cart to town anytime you like by clicking the red arrow to the right.
- Buy power-ups early. Having a faster hoe and better equipment will make things much easier for you.
- Yeah, he's old, but that's no excuse for Grandpa to be standing around not working. If Anna is hoeing, make sure Grandpa is getting water ready or preparing food for one of your animals.
- Speaking of animals, there are two schools of thoughts. If an animal is not needed to reach a goal, you can either sell them or ignore them. If you sell them, you get fast money, but will have to raise them again from a baby. If you ignore them, they get mad at you. You might find a special/homemade gift inside one of your work boots, but nothing bad will happen and you'll have full grown animals for future levels.
- Keep an eye out for rabbits and crows. They will eat your crops quickly. Don't let them get close. If you see them, stop what you're doing and click them away.
- The hidden sheep stages are fun, but can drive you crazy. Remember they can be laying sideways, upside down or just a few curls can be sticking up. If you're not sure, save your click. Too many wrong clicks and you get a 15 second penalty.
- You can replay a level, but you don't get extra cash unless improve upon your last performance on that level.
- Just like in real life, goods cost different amounts. It takes 4 pumpkins to equal 1 basket of eggs. Remember this when you have to raise a high amount in your goals. Also, if you have to let something go bad, let it be the item that is worth the least. This is not a foolproof plan. Sometimes it's more beneficial to save the good that takes the longest to grow. Especially if you need that item to complete a goal.

Farm Mania Cheats and Walkthrough

Content(s) of this game guide may not be copied or published on any other site without permission from Casual Game Guides. Please feel free to share the link to this guide with others.
Any resemblance to other guide(s) is purely coincidental. ©CasualGameGuides.com 2006 - 2011

Deutsch: **Farm Mania loesung**
Español: **Farm Mania Tutorial y Trucos**



New Games


Mystery of the Missing Brigantine


Herofy


Tales From The Dragon Mountain: The Strix


The Microbie Story


Sandra Fleming Chronicles: Crystal Skulls


Midnight Mysteries 3: Devil on the Mississippi


The Fall Trilogy Chapter 3: Revelation


Grim Facade: Mystery of Venice Collector's Edition


Munchies' Lunch


Val'Gor: The Beginning

» More New Games

Popular



Golden Trails 2: The Lost Legacy Collector's Edition





Island Tribe 2



Virtual Villagers 5: New Believers

» More Popular Games

Farm Mania Links

Walkthrough | Forum / Help | About | Play Online | Free Trial | Buy Now

Get this Game for only $6.99!

Privacy Policy | About / Contact | Ad Inventory

A Product of Engenuous Software, © 1996-2011

https://web.archive.org/web/20110503225857/http://techcrunch.com/2011/03/31/facebook-now-has-250-million-mobile-users-and-a-new-unified-mobile-website-to-match/

INTERNET ARCHIVE Wayback Machine
http://techcrunch.com/2011/03/31/facebook-now-has-250-million-mobile-users-and-a-new-unified-mobile-website-to-match/ Go
213 captures
2 Apr 2011 - 16 Jun 2
APR MAY JUN
2010 03 2011 2012
About this capture

Get your early bird Disrupt NYC tickets before May 5th at 12:00am PST!

Got a tip? Building a startup? Tell us

Facebook Now Has 250 Million Mobile Users (And A New Unified Mobile Website To Match)

Jason Kincaid
Mar 31, 2011

Digg
Comments



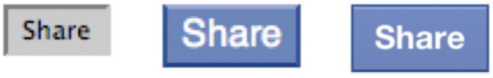
For years now, Facebook has offered quite a few ways to access the site from mobile devices, and they've proven immensely popular. In February 2010 the site had **100 million** mobile users per month — today it's announcing that it's up to 250 million. To coincide with the growth stats, Facebook has some other news to announce about mobile.

We hear most often about Facebook's native applications available for iPhone, Android, WebOS, and other app platforms, but a significant majority of mobile users actually access the site from the web. And up until now Facebook's web setup has been a bit complicated: users on smartphones like Android and iPhones have been directed to touch.facebook.com, which is optimized for large touchscreens. And users accessing the site from a featurephones (and there are a lot of them) would be directed to the more plain-looking m.facebook.com. Today, that's changing: Facebook is merging touch.facebook.com and m.facebook.com into the same site.

But don't worry — you touchscreen users aren't about to be downgraded to the other version. Facebook Head of Mobile Products Erick Tseng explains that up until now having two versions of Facebook's mobile website has led to issues, because the site's engineers would have to rebuild the same features twice, leading the two sites to rarely reach feature parity. Now Facebook has launched a new framework based on XHP, Javelin, and WURFL that uses the same underlying codebase, while tweaking UI elements on the fly depending on what device you're accessing the site from.

In other words, Facebook's engineers only have to implement a feature once on their backend, and if your phone supports it, it'll show up. In addition to paying attention to a phone's hardware capabilities (if your phone doesn't support location, you wont' be able to use Places), Facebook is also optimizing graphics on the fly — you can see three different versions of the Share button in the graphic below.

The new site started rolling out last night, and will be live for everyone in the next few weeks.



Facebook get widget minimize

Website: facebook.com
Location: Palo Alto, California, United States
Founded: February 1, 2004
Funding: $2.34B

Facebook is the world's largest social network, with over 500 million users.

Facebook was founded by... Learn More

Information provided by CrunchBase

Digg

Previous Post Next Post

Advertisement

Most Popular
Now Commented Facebook

Google Dissolves Search Group Internally, Now Called "Knowledge"
Stipple Opens The Kimono To Reveal A Product Tagging Platform With Massive Potential
Netflix CEO Reed Hastings: In Ten Years, "We Will All Have A Gigabit To The Home"
Bill Gates On Nuclear Energy: Compared To Coal, It Is Still Safer In Terms Of People "Killed Per Kil
Twitter To Buy TweetDeck For $40 Million - $50 Million



Featured Posts


VENTURE CAPITAL
Chris Sacca: The Train Has Left The Station On Early Stage Valuations »


GREENTECH
Bill Gates On Nuclear Energy: Compared To Coal, It Is Still Safer In Terms Of People "Killed Per Kilowatt Hour" »


THUNDERBOLT
With Big Graphics, HD Cameras, And Triple Monitor Support, iMacs Are More Pro-Like »


TAGGING
Stipple Opens The Kimono To Reveal A Product Tagging Platform With Massive Potential »


EXITS
Twitter To Buy TweetDeck For $40 Million - $50 Million »

Now Commented Facebook

- Google Dissolves Search Group Internally, Now Called "Knowledge"
- Stipple Opens The Kimono To Reveal A Product Tagging Platform With Massive Potential
- Netflix CEO Reed Hastings: In Ten Years, "We Will All Have A Gigabit To The Home"
- Bill Gates On Nuclear Energy: Compared To Coal, It Is Still Safer In Terms Of People "Killed Per Kil
- Twitter To Buy TweetDeck For $40 Million – $50 Million

Related Topics

Facebook

Advertisement

Leave Comments

Advertisement

The CrunchBoard

Jobs Services

Javascript Guru
Chart.io

Director, Ad Platform Tachnology
Confidential

User Operations Analyst
TrialPay

Gilt Groupe Engineer
Gilt Groupe

Software Engineer / Web Developer
Bug Labs

See all Post

Powered by Personforce

404 Not Found

nginx

© 2011 TechCrunch