

COMMONWEALTH OF PENNSYLVANIA

DEPARTMENT OF Human Services

INFORMATION TECHNOLOGY STANDARD

Name Of Standard: Mobile Development	Number: STD-EASS010
Domain: Application	Category:
Date Issued: 08/19/2013	Issued By Direction Of: Cliff Van Scyoc, Dir of Div of Tech Engineering
Date Revised: 02/17/2016	

Abstract:

The purpose of this standard is to define the guidelines to be used to drive future mobile web application development. These guidelines should be applied based on the needs of the customer and have been defined to make the best use of DHS’s existing development technology, processes, and tools.

To better serve the citizenry and workforce, the Department of Human Services (DHS) has embarked on a strategic initiative to define and operationalize their mobile application architecture. The Department’s goal is to provide an opportunity for DHS to build a standard architecture and develop a set of guidelines that define the foundation for mobile based solutions to support potential citizen facing mobile web applications.

The following areas are to be addressed to improve mobile development:

- Design and UX/UI Architecture
 - Critical to understanding how to create user friendly, interactive and attractive mobile web applications.
- Application Development Architecture
 - The core technology for the creation of mobile web applications.
- Mobile Device Architecture
 - The mobile device architecture for deploying mobile web applications.
- Security Architecture
 - Critical controls and data security considerations.
- Testing, Performance, and Usage Metrics
 - Important testing and performance management considerations during and after application development.

General:

The goal of the process is to create a set of guidelines relevant in the development and deployment of mobile web applications. The purpose of this document is to state standards in

terms that can be applied as technology changes and will require review as technology and requirements change.

The key objectives are to articulate and document the architecture and development guidelines to enable and support the creation of citizen-facing DHS mobile web applications. Developing mobile web applications is not fundamentally different than developing websites and web applications designed for PCs. However, there are important differences. The guidelines herein are intended to describe the items that are not part of the established standards defined in the ALM Baseline (which should be followed where appropriate). The guidelines in this document focus on the mobile-specific topics and include mobile web environments, architectures, security, and standards.

Standard:

DESIGN AND UX/UI ARCHITECTURE

This section describes the activities, tools and processes required to create highly usable, interactive and attractive mobile web applications:

- Mockups, Wireframes, and Prototyping Tools - Tools to facilitate the creation of visualizations of a mobile web application’s screens
- Creative Design – Branding, color scheme, and iconography guidelines
- Accessibility – Creating mobile web applications that are accessible for users with disabilities

	Guideline	Rationale	Tools
Mockups, Wireframes, and Prototyping Tools	Design tools appropriate for the project should be selected. Adobe Creative Suite is an approved tool, but other tools may be used if approved by stakeholders.	Adobe Creative Suite licenses are already owned by DHS. Although other tools may be helpful they will require the purchase of new licenses.	The wire framing tool Balsamiq, the prototyping tool Axure, and the graphics design tools provided in Adobe’s Creative Suite.
Creative Design	Mobile web applications should follow best practices for mobile web design such as being touch friendly and highly visible on small screens. The standard DHS style guide for websites should be used where applicable when creating styles for mobile websites.	Screen size and input methods on mobile devices differ greatly from desktop websites and a different design should be created to increase usability on mobile devices. The appearance of mobile websites should be consistent with DHS’s desktop websites and branding.	Best practices for mobile web design are defined by the W3C in the Mobile Web Best Practices 1.0 document (available on their website). The following documents contain style and branding information for desktop websites that may also apply to mobile websites: <ul style="list-style-type: none"> • Commonwealth of Pennsylvania Style Guide • ITB-APP005 • STD-APP005A

	Guideline	Rationale	Tools
Accessibility	The current accessibility standards issued by the state should be applied as much as possible to mobile websites.	Accessibility standards for websites have been created by the Office of Administration and mobile websites are not exempt.	For Commonwealth of PA's Accessibility Policy see ITB-ACC001. For the W3C's mobile web accessibility guidelines see the website of the W3C Web Accessibility Initiative.

ARCHITECTURE OVERVIEW

The following section provides a high level view of the components that comprise the mobile web architecture. Figure 1 demonstrates the relationship between the current web architecture and the architecture to support mobile web applications.

At a high level, the DHS Mobile Web Architecture can be described in the context of the chosen subject matter areas including **Design UX/UI, Application Architecture, Device Support, Security and Testing, Performance and Usage Metrics.**

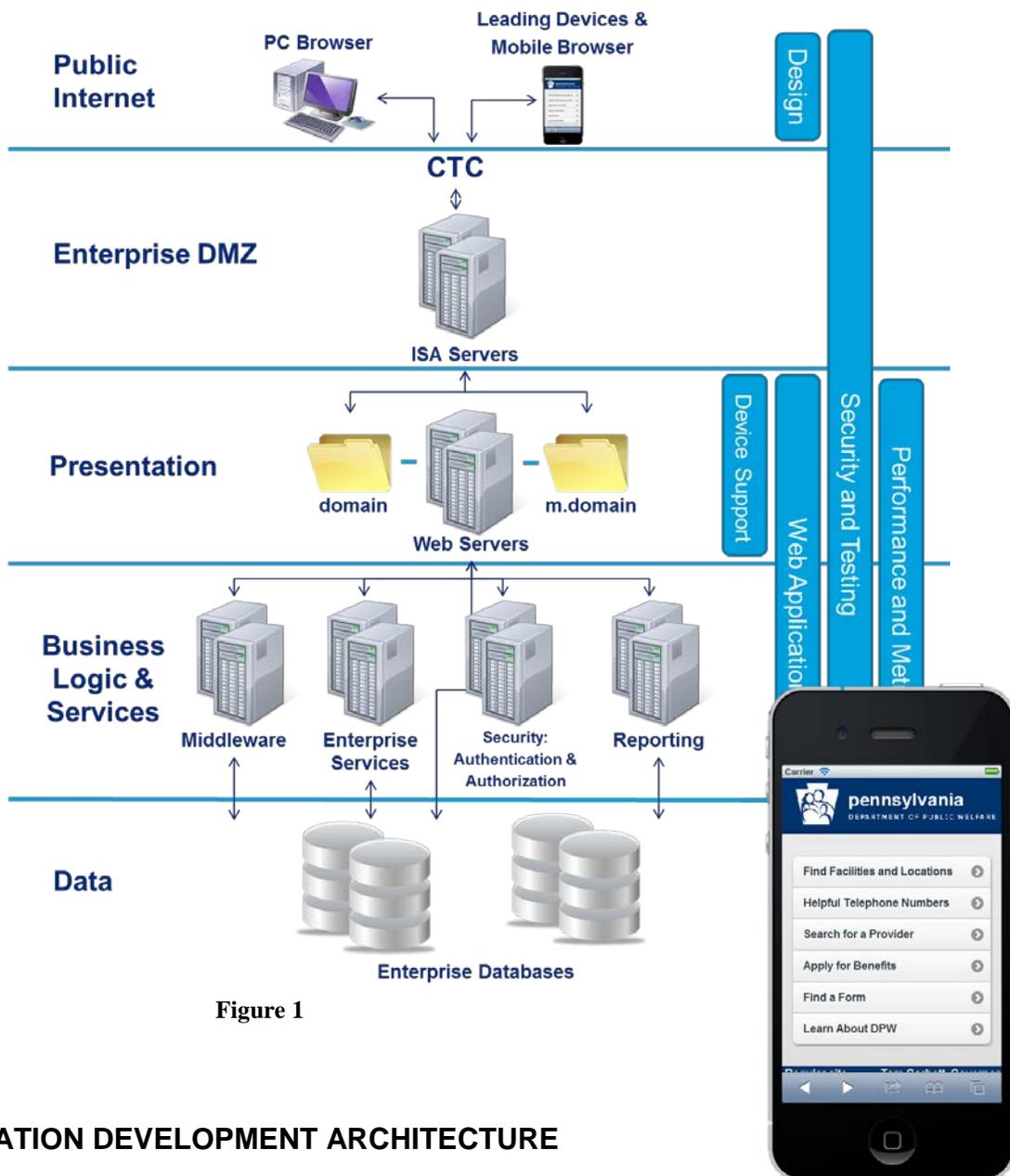


Figure 1

APPLICATION DEVELOPMENT ARCHITECTURE

Figure 2: Mobile websites offer greater portability and connectivity.

This section describes how mobile web applications are rendered and served to mobile devices and any third party services, external to the Commonwealth that provide additional functionality and capabilities (mapping, social media, etc.). It also provides guidance on how mobile web applications should be tested to ensure that consistent results are achieved across a diverse set of browsers and devices.

This section provides guidelines for the following:

- Mobile Page Rendering Approach – Approaches to use for page rendering
- Device Detection – Detecting mobile devices so the correct pages will load on the device
- Server Side Application Framework – Specifies a server side application framework for mobile web applications
- Data Interchange Format – Specifies the data interchange format for mobile web applications
- JavaScript Framework – Specifies the JavaScript framework for mobile web applications
- HTML5 Features – Process for use and review of certain HTML5 features
- Third Party Services – Process of evaluating and using web services offered by third parties external to the Commonwealth

Guidelines

	Guideline	Rationale	Tools
Mobile Page Rendering Approach	The preferred page rendering approach is either a server based page rendering architecture, a rich client architecture, or a combination of the two depending on the needs of the application.	A server based page rendering architecture which renders a complete page on the server has the advantage of being a familiar architecture. A rich client architecture which uses AJAX calls from the client to load and display data can be more responsive, reduce the amount of data passed between client and server, and make use of existing web services. These two architectures can be combined by having some data being rendered into the page on the server but other data being loaded with AJAX calls.	The frameworks selected by DHS in these guidelines can support any of the rendering approaches mentioned.
Device Detection	Servers should detect requests from mobile devices and automatically redirect users to a mobile version of the website (links should be provided to return to the non-mobile version if desired). Server device detection that serves different content to different devices or client side feature detection with JavaScript may be required in order to provide the desired user experience on some mobile web applications. The need for this functionality should be evaluated on an application specific basis.	Redirection for mobile devices is an industry best practice which insures the best experience is provided to mobile users. However, mobile users should be allowed to easily switch to the full version of the site if the features they are looking for are not available on the mobile version. Differences in browser support for HTML5 and hardware difference between devices may require custom code in client side scripts or device specific files to be delivered to the browser in order properly implement features on all supported devices.	Two .NET libraries useful for server side device detection and redirection are: 51Degrees.mobi and WURFL. Modernizr is a JavaScript library that is used to do client side feature detection.

	Guideline	Rationale	Tools
Server Side Application Framework	The preferred server side application framework for mobile web applications is ASP.NET MVC4. If complex web services are required, the Windows Component Framework (WCF) may be required.	ASP.NET MVC4 provides a more flexible architecture for mobile web applications than the .NET Web Forms architecture since it is capable of easily rendering both web pages and JSON services.	ASP.NET MVC4 web applications are developed in Visual Studio 2010. MVC4 requires a separate download from Microsoft to be installed.
Data Interchange Format	The preferred data interchange format for mobile services is JSON.	JSON integrates with JavaScript natively and is the most compact of the standard data interchange formats.	Support for JSON built into Visual Studio and all web browsers that support JavaScript.
JavaScript Framework	The preferred JavaScript framework for mobile web applications is jQuery/jQuery Mobile. If approved by stakeholders, secondary JavaScript frameworks may be used based on project requirements.	jQuery and jQuery Mobile are market leading JavaScript frameworks which are comparatively easy to learn. This framework also supports the widest variety of mobile browsers and has the best support for graceful degradation. Support for these frameworks is available through a partnership with Microsoft.	jQuery Mobile is open source, supported by Microsoft, and requires a separate download from the jQuery Foundation's website. Several alternative frameworks were evaluated including Dojo Mobile and Sencha Touch.
HTML5 Features	The checklist of applicable HTML5 features in Appendix A should be submitted and reviewed for each mobile web application developed.	HTML5 support varies across mobile browsers, features like local storage have security implications, and use of features like geo location can have privacy implications. Reviewing an HTML5 feature checklist of proposed features for an application allows stakeholders to address concerns at an early stage and mitigate risk.	Appendix A provides a checklist based on the HTML5 specification as of May 2012. The features encompassed by HTML5 are evolving and is managed by the W3C HTML Working Group. The latest standards are available on their website.
Third Party Services	If integration with third party services is desired, a document justifying their use should be produced and approved by stakeholders.	In some situations integration with third party services could provide some tangible advantages. However, there are also security concerns and potential issues if the third party has a service outage. Therefore, any third party services should be carefully reviewed by stakeholders before they are implemented.	Examples of common third party services used in mobile applications are mapping APIs and social media interaction.

MOBILE DEVICE ARCHITECTURE

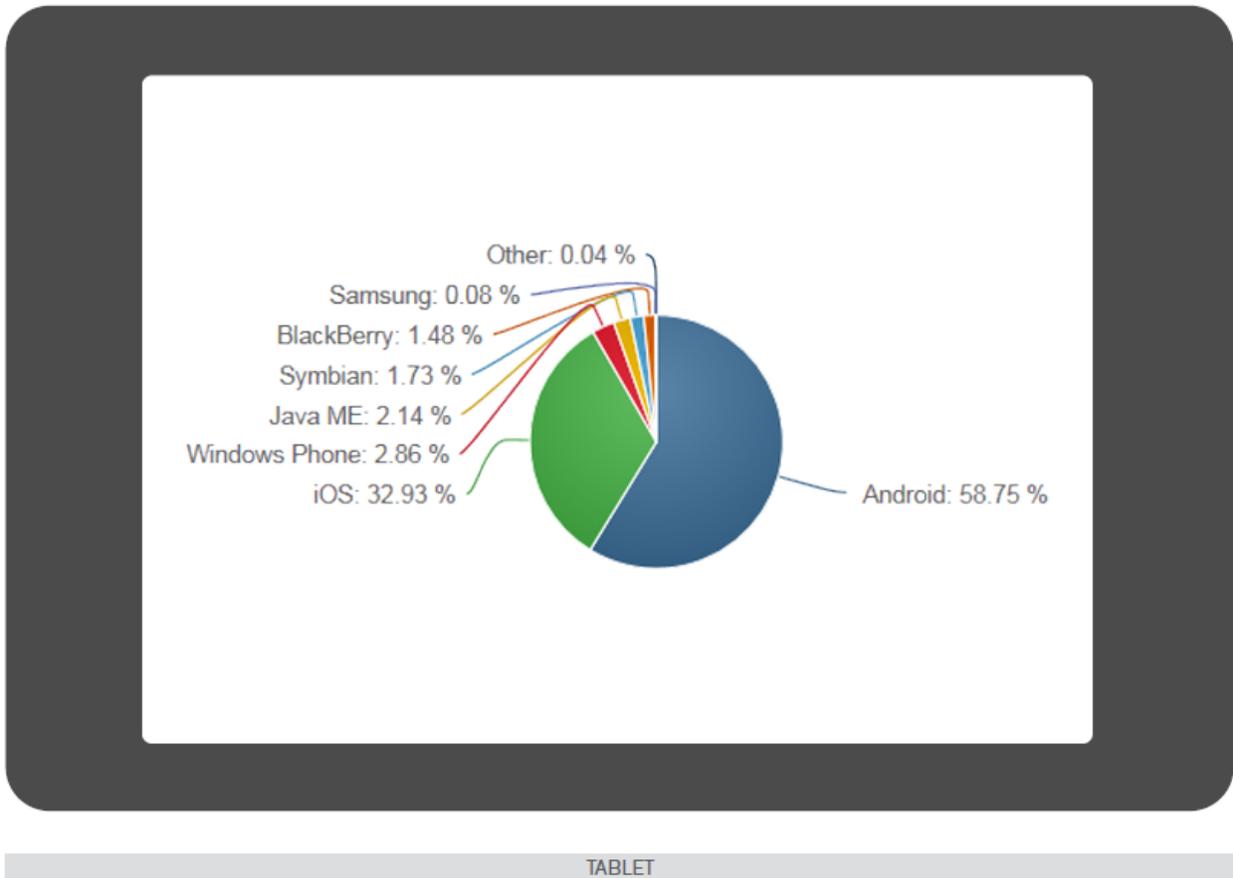
This section describes the systems and devices required for end to end solutions. Devices selected for support are based on the current trends in the market. As shown in Figure 3 below, statistics over the last 4 years indicates a majority of the smartphone market is held by Android devices, followed by iOS devices.

Period	Android	iOS	Windows Phone	BlackBerry OS	Others
2015Q2	82.8%	13.9%	2.6%	0.3%	0.4%
2014Q2	84.8%	11.6%	2.5%	0.5%	0.7%
2013Q2	79.8%	12.9%	3.4%	2.8%	1.2%
2012Q2	69.3%	16.6%	3.1%	4.9%	6.1%

Source: IDC, Aug 2015

Figure 3: Mobile Device Traffic

The tablet market share is rapidly changing. As shown in the Figure 4, the Android tablet is rapidly taking over market share.



Source: NetMarketShare

Figure 4: Tablet Market Share

Figure 5 shows a breakdown of mobile traffic that visited DHS’s websites during January 2016. It reinforces the need for testing on a variety of devices.

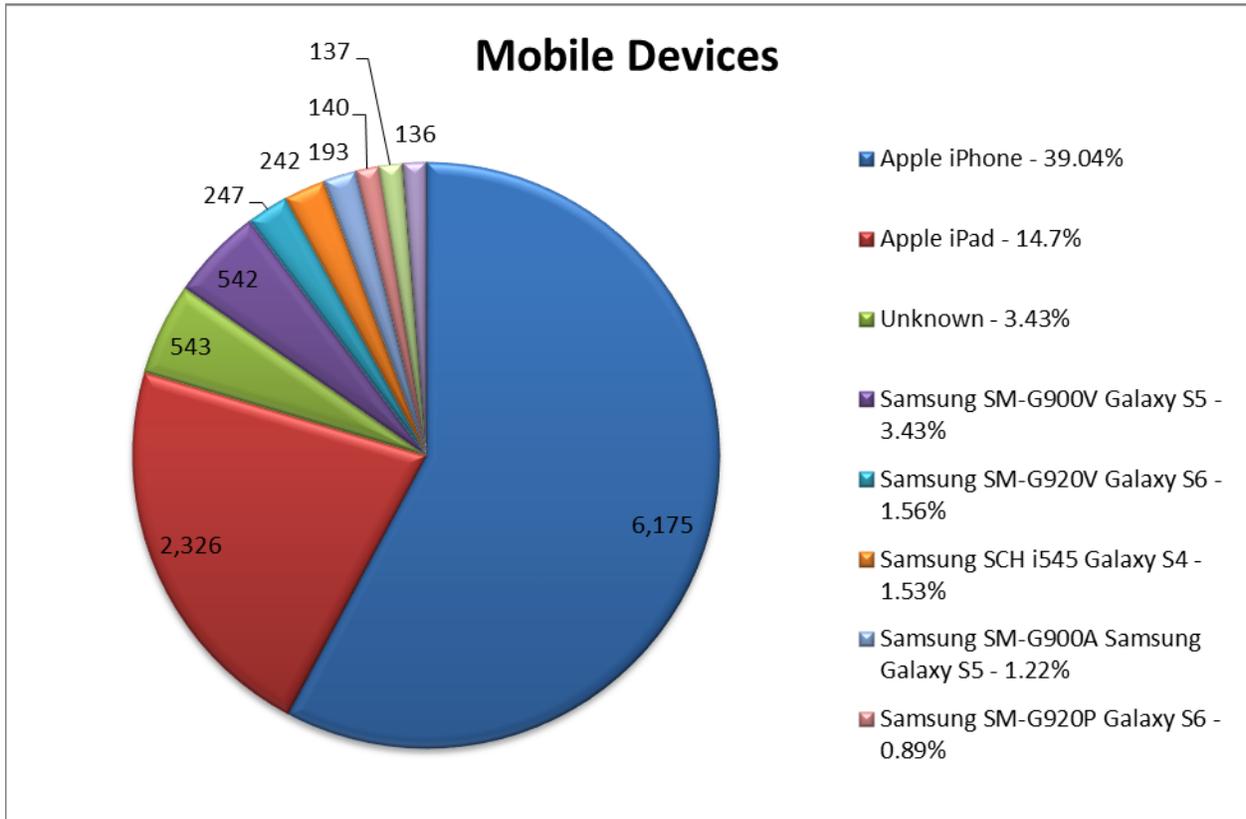


Figure 5: Mobile Traffic to DHS Sites, January 2016

This section provides guidelines for the following:

- Device Support – Which devices should be supported by mobile web applications

	Guideline	Rationale	Tools
Device Support	Mobile web applications should be fully functional on a set of the top devices found to be accessing DHS's mobile web applications.	DHS recognizes that mobile device landscape changes rapidly and market leading devices and operating systems can be obsolete in less than a year. The top device categories accessing DHS's public websites are a moving target that closely conforms to the devices citizens are actually using, making it a suitable guide for determining device support.	As of May 2012 the following devices are market leaders across several categories and encapsulate a range of operating systems and screens sizes. Apple iPhone Apple iPad Samsung Galaxy S5 Samsung Galaxy S6 Testing on these primary devices will provide coverage for the most common devices on the market at the time of this document's creation. These devices need to be re-evaluated periodically to keep pace with current market trends.

SECURITY ARCHITECTURE

The security control guidelines have been defined to make the best use of DHS's existing security technology, processes and tools in the ALM Baseline.

- A copy of the Mobile Application Security Standards can be found here:
http://myDHS/cs/groups/webcontent/documents/communication/p_031986.pdf.
- A copy of the Mobile Device Standard can be found here
http://myDHS/cs/groups/webcontent/documents/communication/p_031987.pdf.

TESTING, PERFORMANCE, AND USAGE METRICS

This section describes the tools and methods used to test mobile web applications during and after the development, validate the performance of mobile applications, and the metrics and tools be used to assist in the process. Where the stated guidelines do not address a specific area, it is either described in the ALM Baseline or the topic is not part of the scope of this standard.

This section provides guidelines for the following:

- Performance – Guidelines to minimize the amount of data transferred to mobile devices and maintain the minimum acceptable levels of performance for mobile web applications
- Usage Metrics – Specifies the tool used for metrics gathering and mobile web application usage analytics
- Load Testing – Tools used for load testing mobile web applications
- Functional Testing – Guidelines and options for testing the functionality of mobile web applications
- Device Testing – Testing mobile web applications on devices

	Guideline	Rationale	Tools
Performance	Mobile websites should have GZIP compression enabled, JavaScript files should be minified, and HTTP headers should be as small as possible.	Mobile devices can be limited by slow connection speeds or high cost of bandwidth so the specified practices should be applied to minimize the size of all content transmitted to and from the web server.	GZIP is supported by IIS, the web server currently used by DHS. The following minification tools are all solid options that can be integrated into the Visual Studio/TFS build process: <ul style="list-style-type: none"> • YUI Compressor • Microsoft AJAX Minifier • Google Closure Compiler
Usage Metrics	Google Analytics should be used to gather usage statistics for all mobile web applications.	Google Analytics is a free tool already being used by DHS and many other enterprises. No need is seen for the specific features of competing tools.	The following tools were evaluated: <ul style="list-style-type: none"> • Google Analytics • Site Catalyst • Coremetrics • Percent Mobile
Load Testing	DHSs standard load testing tool should be used to load test all mobile applications.	Load testing new websites is a DHS standard and industry best practice.	NeoLoad is the standard load testing tool for DHS as of May 2012 and it supports load testing of mobile sites.

Functional Testing	<p>Mobile web application functionality should be tested on devices (see device testing) but testing in browser emulators that emulate a broader range of devices than the supported devices is an industry trend. Use of this practice should be evaluated on a per project basis.</p> <p>Testing pages with W3C's mobileOk testing tool is also an industry trend that should be evaluated on a per project basis.</p>	<p>The large number of devices and speed with which new devices, operating systems, and mobile browsers are introduced makes testing web applications on the devices used by a significant portion of users prohibitively expensive and time consuming without using emulators.</p>	<p>The following emulators and testing tools were discussed:</p> <ul style="list-style-type: none"> • Keynote MITE • Perfecto Mobile • Compuware Gomez • CrossBrowserTesting • W3C mobileOk • mobiReady • Google PageSpeed
Device Testing	<p>All mobile web applications should be tested on the current list of supported devices.</p>	<p>It is an industry best practice to test on physical devices to verify that pages render correctly and features are fully functional.</p>	<p>The mobile web application should be loaded on all of the supported devices. Adobe Shadow was discussed as a way to make this process easier.</p>

Exemptions from this Standard:

There will be no exemptions to this standard.

Refresh Schedule:

All standards and referenced documentation identified in this standard will be subject to review and possible revision annually or upon request by the DHS Information Technology Standards Team.

Standard Revision Log:

Change Date	Version	Change Description	Author and Organization
02/17/2016	1.1	Reviewed and updated for accuracy	Bradley Deetz/Laura Fry DEA