



MedTouch Quality Assurance Testing Methodology v1.0

CLIENT	MedTouch
PROJECT	Development Testing
DATE	June, 2010
PREPARED BY	John Arnold

24 Thorndike St
Cambridge, MA 02141
617.621.8670
www.medtouch.com

MEDTOUCH PROPRIETARY INFORMATION

The information contained herein is MedTouch Proprietary Information. External disclosure or use outside of MedTouch shall be made only in accordance with the terms of an appropriate Non-Disclosure Agreement. For more information contact: solutions@medtouch.com

Copyright © 2005-2010 MedTouch, LLC. All rights reserved.

Table of Contents

1	Software Testing Approach	3
1.1	Unit Testing	3
1.2	System Testing	3
1.3	User Acceptance Testing	3
1.4	Reference Documents.....	3
2	Quality Assurance	5
2.1	How can Software QA process be implemented without reducing productivity?	5
2.2	Determining the Number of Evaluators.....	5
2.3	The Benefits from Heuristic Evaluation	6
3	Best Practices	8
3.1	(Unit/System)Test Plan.....	8
3.2	QA Checklist.....	8
3.3	Open Issues Report.....	13
3.4	QA Audit.....	14
4	Testing Methodology.....	15
4.1	Software Testing Methodologies	15
4.2	Waterfall Model.....	15
4.3	Using the QA Checklist	15
5	Appendix A (Glossary)	17

1 Software Testing Approach

MedTouch utilizes a three tier approach to Software Testing:

1.1 Unit Testing

Unit Testing is integrated into the coding phase of application development and comprises the on-going testing of individual components and/or module functions. Unit Testing is usually conducted by the development team.

1.2 System Testing

System Testing is conducted on a complete, integrated system to evaluate compliance with specified requirements. System testing utilizes the end-user interface to simulate the production environment to validate communications between the various application modules. System Testing is generally completed by project stake holders (project managers etc) external to the development team to provide an outside perspective.

System testing can be further delineated into Functional and Non-Functional testing.

1.2.1 Functional vs Non-Functional Testing

Functional testing refers to tests that verify a specific action or function of the code. These are usually found in the requirements specification documentation, although some development methodologies work from use cases or user stories. Functional tests tend to answer the question of "can the user do this" or "does this particular feature work".

Non-functional testing refers to aspects of the software that may not be related to a specific function or user action, such as scalability or security. Non-functional testing tends to answer such questions as "how many people can log in at once", or "how easy is it to hack this software".

1.2.2 Regression testing

Regression Testing - re-testing after fixes or modifications of the software or its environment. It can be difficult to determine how much re-testing is needed, especially near the end of the development cycle.

1.3 User Acceptance Testing

User Acceptance Testing by the end client attempts to validate properties of the entire integrated system rather than the individual components and that the objects/functions being tested meet mutually agreed-upon requirements.

1.4 Reference Documents

The following documents are relevant to this design specification:

(Available on the J drive: J:\Templates\Project Folder TEMPLATE(ja)\5) Quality Assurance)

- QA Methodology: MedTouch__QA_Methodology.docx (this document)
- QA Best Practices: MedTouch_QA_Methodology_and_Best_Practices.docx
- Unit/System Test Plans: MedTouch_System_Test_Plan.xls
- QA Check List: MedTouch-website-QA-checklist-v1.xlsx
- Open Issues report: MedTouch_Open_Issues_Report.xls
- QA Audit: MedTouch_QA_audit_checklist_v1.xlsx

2 Quality Assurance

As defined by the online [Project Management glossary](#), Quality Assurance (QA) is "a planned and systematic pattern of all actions necessary to provide adequate confidence that the item or product conforms to established technical requirements."

Proceeding testing there is frequently on-going issue tracking, see the Open Issues Report section. This formal document would typically only be used for large projects &/or long development life cycles with many iterations &/or phased releases. MedTouch's issue tracking can optionally be coordinated utilizing established client tools and processes for assigning and tracking these issues &/or using the To-Do functions of Basecamp.

In addition to the Test Plan, our Copy writing & Creative departments in conjunction with a QA auditor will need to be familiar w/ the client's Style Guide. The QA Checklist is used to confirm formatting (Header font/size, confirm link formatting etc) is in line with the client specific style guide and our internal coding standards and best practices.

2.1 How can Software QA process be implemented without reducing productivity? ¹

By implementing QA processes slowly over time, using consensus to reach agreement on processes, focusing on processes that align tightly with organizational goals, and adjusting, experimenting and refactoring as an organization matures, productivity can be improved instead of stifled. Problem prevention will lessen the need for problem detection, panics and burn-out will decrease, and there will be improved focus and less wasted effort.

At the same time, attempts should be made to keep processes simple and efficient, avoid a 'Process Police' mentality, minimize paperwork, promote computer-based processes and automated tracking and reporting, minimize time required in meetings, and promote training as part of the QA process. However, no one - especially talented technical types - likes rules or bureaucracy, and in the short run things may slow down a bit. A typical scenario would be that more days of planning, reviews, and inspections will be needed, but less time will be required for late-night bug-fixing and handling of irate customers.

2.2 Determining the Number of Evaluators²

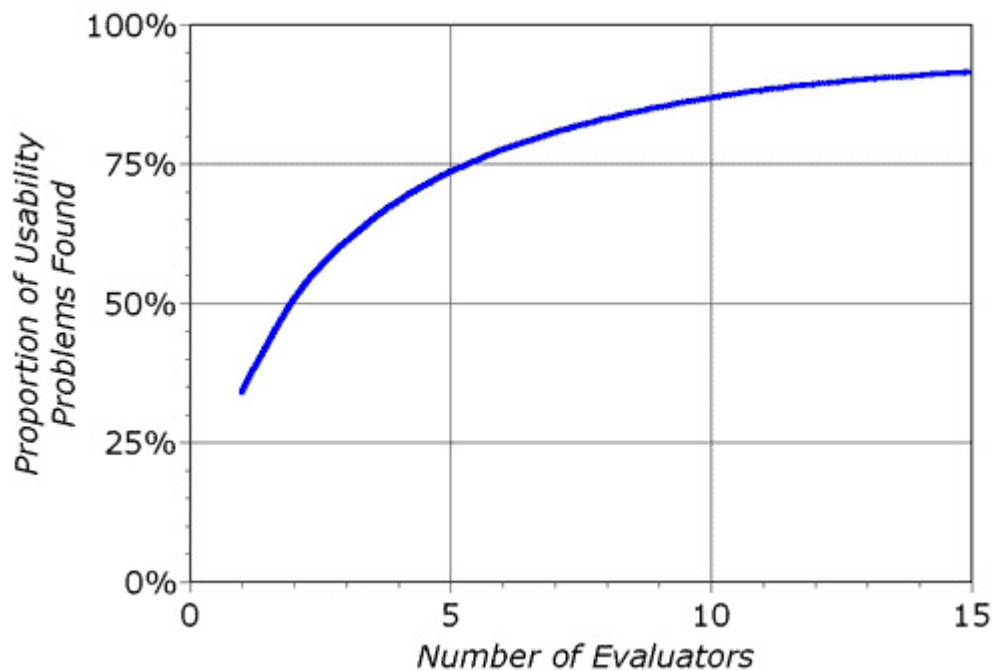
In principle, individual evaluators can perform a heuristic evaluation of a user interface on their own, but the experience from several projects indicates that fairly poor results are achieved when relying on single evaluators. Averaged over six of my projects, single evaluators found only 35 percent of the usability problems in the interfaces. However, since different evaluators tend to find different problems, it is possible to achieve substantially better performance by aggregating the

¹ Reference: Software QA and Testing Less-Frequently-Asked-Questions
http://www.softwareqatest.com/qat_lfaq1.html#LFQA1_6

² Reference: How to Conduct a Heuristic Evaluation
http://www.useit.com/papers/heuristic/heuristic_evaluation.html

evaluations from several evaluators. Figure 2 shows the proportion of usability problems found as more and more evaluators are added.

2.2.1 Number of Evaluators Graph



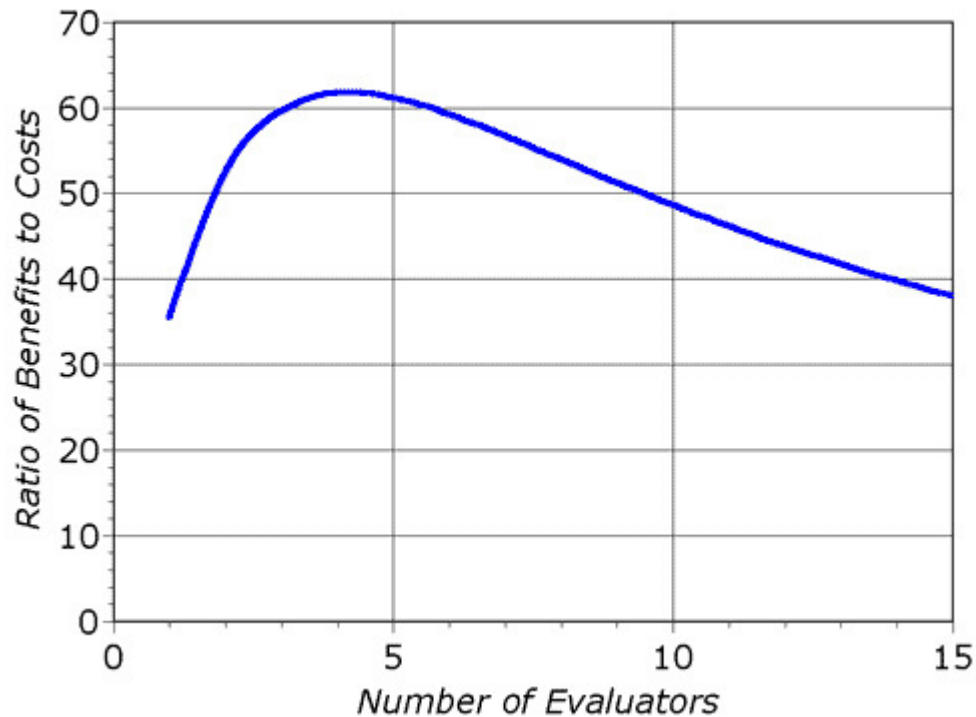
Curve showing the proportion of usability problems in an interface found by heuristic evaluation using various numbers of evaluators. The curve represents the average of six case studies of heuristic evaluation.

2.3 The Benefits from Heuristic Evaluation ³

The benefits from heuristic evaluation are mainly due to the finding of usability problems, though some continuing education benefits may be realized to the extent that the evaluators increase their understanding of usability by comparing their own evaluation reports with those of other evaluators.

³ Reference: How to Conduct a Heuristic Evaluation
http://www.useit.com/papers/heuristic/heuristic_evaluation.html

2.3.1 Number of Evaluators Graph



Curve showing how many times the benefits are greater than the costs for heuristic evaluation of a sample project using the assumptions discussed in the text. The optimal number of evaluators in this example is four, with benefits that are 62 times greater than the costs.

...The curve shows that the optimal number of evaluators in this example is four, confirming the general observation that heuristic evaluation seems to work best with three to five evaluators. In the example, a heuristic evaluation with four evaluators would cost \$6,400 and would find usability problems worth \$395,000.

3 Best Practices

There are many variations of development methodologies & industry best practices. Most of them are very similar; it's really more a matter of selecting one & adhering to it. Adherence to a single methodology is also something a formalized QA process addresses. As MedTouch continues to refine our development processes the team will bring their differing methodologies into line with one another. Adhering to approved Coding Standards provide a baseline to formalize MedTouch development standards.

3.1 (Unit/System)Test Plan

A Test Plan can be prepared to describe every user interaction with the web site and the expected results as a pass/fail element. i.e. Action: Enter the home page URL www.xxx.com; Expected Result: The home page loads. Pass / Fail

Most web sites are comprised of similar components, to minimize the need for lengthy, manually prepared Test Plans; we have prepared a QA Checklist to streamline the testing process.

3.1.1 Test Plan Screenshot

MedTouch		Module Test Plan: Calendar of Events - UI		6/23/2010 Page 1 of 4	
Item	Description	Expected Result	Pass/Fail	Notes	
		o Module home page editor displays the following components: o Page Header o Introduction paragraph (editable content text area) o Class/Event titles (links) o Mini-Calendar o Search - Keywords			
1	Enter module URL (ex: www.medtouchit.com/prototype/calendar/)	o Search - Date Range (from / to)			
2	Click on the Mini-calendar Back (arrow) button	Minicalendar month descends by one month (ex: Dec. to Nov)			
3	Click on the Mini-calendar Forward (arrow) button	Minicalendar month ascends by one month (ex: Nov to Dec)			
		o Links to the calendar search results page for the selected date o Displays the following Event components: o Date o Title o Time o Location o Sign Up (submit button)			
4	Select a high-lighted Mini-calendar Date	Note: The selected criteria may include multiple results.			
5	Select Back on the browser to return to module home page	Module home page loads.			
6	Select the Search button to submit, do not enter a Keyword	Links to the (Calendar) Search Results page and displays all Events for the next thirty (30) days.			
7	Select Back on the browser to return to module home page	Note: The selected criteria may include multiple results.			
8	Select the Search - Keywords field, enter special characters (SHIFT 1 through 4), select the Search button to submit	Module home page loads.			
9	Select Back on the browser to return to module home page	Links to the (Calendar) Search Results page for the selected Keyword.			
10	Select the Search - Keywords field, enter "Class", select the Search button to submit	Note: The results should display a No Records Found message.			
11	Select Back on the browser to return to module home page	Module home page loads.			
12	Select the Search - Keywords field, enter "Class", select the Search button to submit	Links to the (Calendar) Search Results page for the selected Keyword.			
13	Select the Search - Date Range From field, select the current date, select the Search button to submit	Note: The selected criteria may include multiple results.			
14	Select the Search - Date Range To field, select a future date, select the Search button to submit	Links to the (Calendar) Search Results page for the selected Date Range.			
15	Select Back on the browser to return to module home page	Note: The selected criteria may include multiple results.			
		Module home page loads.			

MedTouch Module Test Plan - Calendar of Events - UI
Completed by: _____
Print Name: _____

3.2 QA Checklist

Every website developed by MedTouch is subject to strict quality guidelines that must be satisfied before handover to client. This ensures the finished site meets all the required accessibility, usability, design, and technical "best practice" that we strive for.

The QA Checklist has been developed to assure these best practices are implemented in a consistent manner by streamlining the process of validating the web site meets the defined criteria in the following key areas:

- ARCHITECTURE AND NAMING CONVENTIONS
- DESIGN
- NAVIGATION
- PROGRAMMING; CSS, JAVASCRIPT AND OTHER CODE
- STANDARDS COMPLIANCE
- BROWSER COMPATIBILITY
- IMAGES
- MULTIMEDIA
- CONTENT
- EXTRA FEATURES
- MODULE CHECK

Additional details can be added to subsequent tabs allowing for full site QA to be compiled in a single document.

3.2.1 QA Checklist Screenshots



WEBSITE DEVELOPMENT QA CHECKLIST

Every website developed by MedTouch is subject to strict quality guidelines that must be satisfied before handover to client. These guidelines include required accessibility, usability, design, and technical "best practice" that we strive for.

GENERAL INFORMATION

Project Name: Massena

Page/Section: Full Site QA

HISTORY

Reviewed By

John Arnold

Date

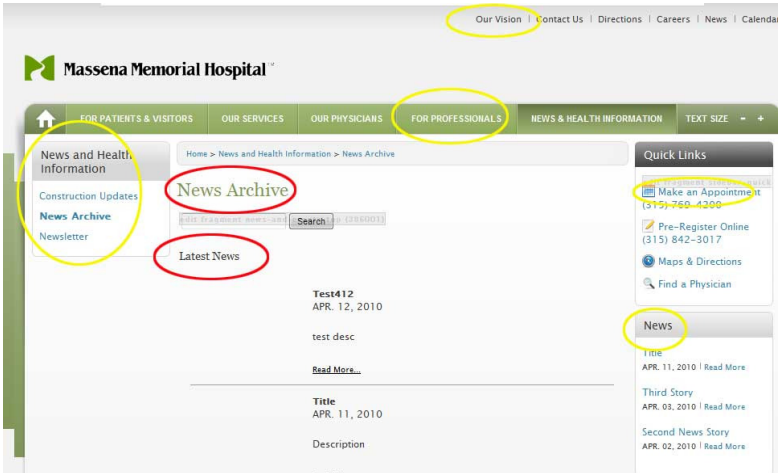
4/19/2010

Reviewed By

Notes

Full site QA

http://mas

REVIEW CRITERIA	YES	NO	
ARCHITECTURE AND NAMING CONVENTIONS			
Recommendation: Is all URL Naming Consistent		X	Our Physi Hospital) Hospital)
Recommendation: Is all URL Case Consistent	X		Ex. all lowe
Recommendation: Are the page URL names consistent with the page?	X		
Deleted all extraneous files	NA		
DESIGN			
Headlines color and size correct?	X	?	Page Hea are Arial
			
Sub-headlines color and size correct?	X	?	See Head
Intro copy color and size correct?	X	?	Modules -
Body copy color and size correct?	X		
Link color and size correct?	X		
Are all clickable areas clickable? (images, feature titles, more links...)		X	See addit
Are images the correct size, alignment and dimension?	X		
NAVIGATION			
Have all links been tested to make sure they work?		X	"Make an design?"
Do all links pointing to external sites open a new window?	NA		Staging site
Are the quick link dropdowns working and going to the correct page?		X	Staging s
PROGRAMMING; CSS, JAVASCRIPT AND OTHER CODE			
Have all code been tested for programming or script errors?	NA		
Are all error and boundary conditions properly handled with user friendly messaging?	X		

Is the enlarge text function working?	X		
Is the print page function working?	X		
Are the CSS styles working on all of the modules?	X		
STANDARDS COMPLIANCE			
Did we validate the HTML/XHTML code using W3C Markup Validation Service?	NA		
Did we validate the CSS code using W3C CSS Validation Service?	NA		
Have all pages passed XHTML and CSS Validation?	NA		
BROWSER COMPATIBILITY			
Did we check the website in the browsers and platforms as defined in the agreed proposal?		X	Initial testi
IMAGES			
Do all images have ALT tags?		X	
Recommendation: Are all image names using the correct format?	NA		
Recommendation: Do the image names make sense where other people know what they are?	NA		
Does the client logo link back to the homepage?	X		
Recommendation: Add the ALT attribute to all images?	NA		
Recommendation: Appropriate file type for your images?	NA		
Did we use plain text instead of images for important content and navigation?	X		
MULTIMEDIA			
Does the Flash piece link to the correct pages?	NA		
Has the Flash been implemented using the SWF Object?	NA		
If Flash has been implemented with the SWF Object is there alternate content?	NA		
Did we make sure that music and video clips don't start playing automatically?	NA		
Did we make sure that music and video clips can be turned off at any time?	NA		
Did we inform the user about the size and length of your music and video clips?	NA		
CONTENT			
Did we proofread your content and SEO the copy as defined in the agreed proposal?	NA		
EXTRA FEATURES			
Did we make an rss feed? (for news items)	NA		Optional
Is the health library integrated correctly? (feeds and design)	NA		Optional
Is the internal search engine indexing the site?	NA		Optional
Are all of the fragments in place and editable?		X	Editable, spreadsheet
Is Site by MedTouch on every footer?	X		Optional

Are all social media features linking to the correct pages?	NA		Optional
Are third party integrations working correctly? (email, job boards, iframes)	NA		Optional
Are all maps displaying correctly and linking to directions?		X	See addit
Are social bookmarking options on all required pages? (ShareIt and AddThis)	NA		Optional
Are all of the feeds working? (events, news, health library)	NA		Optional
MODULE CHECK			
Find a Doctor	X		See forms
Pre-Registration	X		See forms
Bill Pay	X		See forms
Contact Us	X		See forms
Events Calendar	X		See forms
Ways to Give	NA		See forms
Add additional modules	X		See forms
FORMS			
Did we make a friendly 'thank you' page with a confirmation email?	X		Editable,
Did we tell the user what to expect after clicking the submit button?	NA		
Did we place an asterisk when a field is compulsory?	X		
Does all field validation function as defined in the agreed proposal?	X		
Did we keep the standard look of input fields that is generated by the browser?	X		
Did we create a logical order of asking information?	X		
Did we make sure that users can fill in the entire form using the TAB key?	X		
Did we check if your forms also work with JavaScript turned off?	NA		Not teste
MISC			
Google Analytics Code Installed?		X	
Google, Yahoo and Microsoft Webmaster Code Installed?	NA		
Has credit card authorization been confirmed? (PayPal, Authorize.net)		X	In proces
Are the correct security certificates in place?		X	
Have 301 and other relative redirects been established or communicated to the client?	NA		Not teste
Do all pages have Meta Descriptions?		X	
Have all pages that we added been spell checked?	NA		Not teste
Are all <titles> consistent with their pages?		X	See ARCH
*Optional - based on clients contract and final approved design		QA findings:	

3.2.2 Test Plan - Subsequent Tabs Screenshot

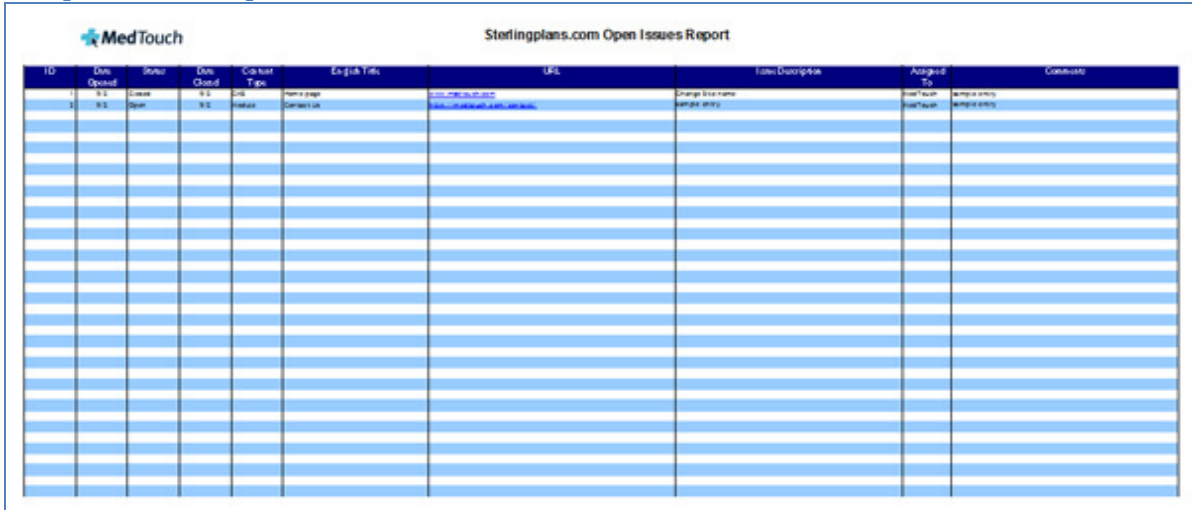
	A	B	C	D	E	F	G	H	I	J
1										
2	Subpages									
3										
4		For Patients & Visitors			Issues:					
5										
6			Cafeteria		Copy is FPO					
7			Directory Listing		Copy is FPO					
8			Gift Shop		Copy is FPO					
9			How To Find		Copy is FPO					
10			Select Dept		Dept's are FPO					
11					Image is FPO					
12					Copy is FPO					
13			Location Key		Links are inactive					
14			View Larger Map		Link is inactive					
15			Volunteering		Copy is FPO					
16			Patient Bill of Rights		Copy is FPO					
17			Patient Records		Copy is FPO	(listed on Footer tab)				
18			Stork Report		Copy is FPO					
19										
20										
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										

Ready | v1.1 | Home | TopNavSubpages | Footer | For Patients | ForPatientSubpages | OurSvcs | OurSvcsSubpages

3.3 Open Issues Report

Open Issues are typically generated from the User Acceptance Testing and the Open Issues Report is a "Bug report" shared with the customer to track issue resolution status. The report is similar in design to the generic Test Plan outlined above and generally implemented on large scale projects with multiple complex components and phased releases. Issue tracking via Basecamp To-Do list is generally sufficient for the majority of projects.

3.3.1 Open Issues Report Screenshot



ID	Date Opened	Status	Date Closed	Col user	English Title	URL	Issue Description	Assigned To	Comments
1	3/1	Open	3/1	Hester	Sterlingplans.com	http://sterlingplans.com	Sterlingplans.com	Hester	Sterlingplans.com

3.4 QA Audit

The QA Audit form provides an internal review document to verify the over-all QA process has been properly implemented and covers the following topics.

- Project Kickoff
 - Is there evidence of a project kickoff meeting in the project folder (i.e., meeting agenda, meeting minutes)?
- Project Plan
 - Has a project plan been generated?
 - Has the project plan been approved by the appropriate level of management?
- Release Contents
 - Has a release contents list been generated and placed in the project folder?
- Estimation Worksheet
 - Has an estimation worksheet been completed and placed in the project folder?
- Project Schedule
 - Has the project schedule been initiated and placed in the project folder?
- Requirements Review
 - Was a Requirements Review conducted for this project?
 - Has a checklist/minutes for the review been placed in the project folder?
- Requirements Specification Document
 - Has the RSD been completed or updated and placed in the project folder?
 - Was a peer review held for this document?
 - Has the document been approved/signed and placed under CM control?
- Design Review
 - Was a Design Review conducted for this project?
 - Has a checklist/minutes for the review been placed in the project folder?

4 Testing Methodology

4.1 Software Testing Methodologies

As noted above, there are many used in software development and testing. Following are some of the most commonly used methodologies:

- Waterfall model
- V model
- Spiral model
- RUP
- Agile model
- RAD

4.2 Waterfall Model⁴

The waterfall model utilized by MedTouch adopts a 'top down' approach regardless of whether it is being used for software development or testing. The basic steps involved in this software testing methodology are:

- Requirement analysis
- Test case design
- Test case implementation
- Testing, debugging and validating the code or product
- Deployment and maintenance

4.3 Using the QA Checklist

The QA Checklist provides a starting point for site testing and documenting the results. The primary testing functions should include functional validation of all components in both normal and abnormal conditions. QA should intentionally include testing for bugs, form validation and proper error handling.

The most important aspect of QA testing is being methodical. Develop a method that works for you that can easily be replicated page after page.

Typical web page QA:

- Check for obvious errors before going on
- Start at the top & work left to right, top to bottom down the page
 - Page title
 - URL naming conventions
 - Validate all navigation elements (secondary & primary)
 - Logo placement/link

⁴ Reference: Software Testing Methodologies
<http://www.buzzle.com/articles/software-testing-methodologies.html>

- Page structure/template selection
- Text colors, fonts, sizes
- Link validation (click them!)
- Image alt tags
- Validate all Footer links

- Form validation
 - Submit the form blank to verify form validation
 - Submit the form with valid information
 - Submit the form with invalid information
 - Confirm validation error handling
 - Is the message correct?
 - Is the message in User Friendly language?
 - Omit required fields
 - Enter text in numeric fields (phone)
 - Enter special characters in all fields (hold shift & swipe the number keys)
 - Validate date/time formats

- Administrative functionality
 - Validate Text Fragments are editable

- Module Testing
 - Test Module Functionality independently from web page content testing
 - Test Module Administrative functionality independently

5 Appendix A (Glossary)

Name	Definition
Unit Testing	Unit Testing is integrated into the coding phase of application development and comprises the on-going testing of individual components and/or module functions.
System Testing	System Testing is conducted on a complete, integrated system to evaluate compliance with specified requirements.
Functional Testing	Functional testing refers to tests that verify a specific action or function of the code.
Non-functional Testing	Non-functional testing refers to aspects of the software that may not be related to a specific function or user action, such as scalability or security.
User Acceptance Testing	User Acceptance Testing by the end client attempts to validate properties of the entire integrated system.
Regression Testing	Regression Testing - re-testing after fixes or modifications of the software or its environment.
Heuristic	A heuristic is a "rule of thumb", an educated guess, an intuitive judgment or simply common sense.