**Doctors on Demand**

Test Plan Document

April 02, 2019

Team: Ayana Perry, Jinal Bangur, John Saterfil, Tianhou Sun

# Testing Plan

We are to write a test plan for a radically Doctor on Demand application that includes smart phone applications. The smart phone application allow customers/users to login or sign up and look for a doctor online at their own convenience.

The test plan covers all aspects of testing. It describes the development approach, e.g. plan driven or agile, and how that impacts the testing. It describes the different phases of testing from unit through system test including usability and security testing. The test plan also explains the tradeoffs of your decisions and identify the risks.

## 

## Application Overview

DoD is being developed to allow patients to talk to a Doctor whenever, regardless of whether you have medical insurance coverage or not. Online Medical Care 24/7 is the future for treating patients.

### 

### Introduction

**• Describe the overall system and high-level goals**

o DoD is being developed to allow patients to talk to a Doctor whenever, regardless of whether you have medical insurance coverage or not. Medical Care 24/7 is the future for treating patients.

o The concept for the Doctors on Demand App came about from patients who expressed other alternatives to Urgent Care facilities, Emergency Rooms and Hospital appointments for minor concerns and questions.

o The Doctors on Demand App has been developed by a team who wanted to make an impact on the medical industry as we know it. The DoD App goal is to reach anyone who does not want to wait days to visit a doctor or must book an in-person appointment just to speak with their primary care physician. The focus for now is on the national level with the intent to expand internationally in the future. So, come and sign up and enjoy a new fresh idea to modern medicine as we know it.

**• Testing strategy Executive Summary - a very high-level overview**

o Test Strategy is a critical step in making a Test Plan. A Test Strategy document is a high-level document. Testing strategy supports DevOps and continuous testing. And continuous testing is important to improving product quality.

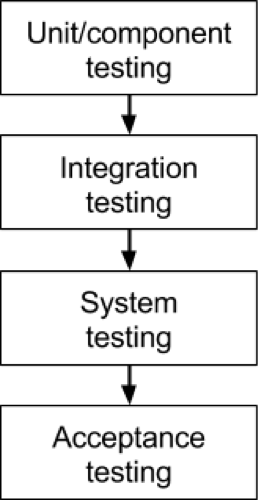
o In this development environment, testing needs to happen early and often. So, instead of waiting for development to be finished before testing begins, testing happens continuously as features are added.

o Tests are prioritized just like user stories. Testers aim to get through as many tests as they can in an iteration. Adding automated testing tools can help testers get through more of the testing backlog.



**• What development approach are you using? Waterfall? Plan driven? Agile?**

o Development approach that we are using is Agile.



**• How does development approach impact testing?**

o Agile testing is well suited for small projects.

o As testing begins at the start of the project, errors can be fixed in the middle of the project.

o There is very less documentation required for agile testing.

o In this approach, every iteration has its own testing phase. The regression tests can be run every time new functions or logic are released.

o In agile testing shippable features of the product are delivered to the customer at the end of an iteration.

o Testers and developers work closely in Agile testing.

o User acceptance is performed at the end of every sprint.

o The testers need to set up communication with developers to analyze requirements and planning

**2. Testing scope**

**• What are you testing?**

o The components of the system to be tested (hardware, software, middleware, etc.) are defined as "in scope"

o As the software requirement specs, the project DoD only focus on testing all the functions and external interface of application DoD (in scope testing)

o In-scope items: Functional Testing, Api Testing

**• What are you not testing?**

o The components of the system that will not be tested also need to be clearly defined as being "out of scope."

o Nonfunctional testing such as stress or logical database currently will not be tested

o Out of scope items: Database Testing, hardware & any other external interfaces

**• Describe the criteria you used for prioritizing tests.**

Test case prioritization is an extremely important element of software testing. It allows testers to test highest priority test cases, which helps them resolve most critical issues earlier in the testing phase. However, it is essential for the testers to follow several factors while executing test case prioritization.

These factors ensure that proper testing is executed in proper order and validates that all discrepancies are detected during the process of testing. The aspects that are needed to be considered while prioritizing test cases are:

o The tester should consider the usage frequency of a function or the probability of failure in software use.

o The failure risks of the test cases should be measured.

o Visibility of a failure for the end user is a further criterion for prioritization of the test cases.

o Test cases can be chosen as per the priority of the requirements of all the stakeholders.

o Quality characteristics may have different importance to the customer or client.

o Prioritization can also be done from the perspective of the development or system architecture.

o Areas with complex coding and with past histories of problems should be considered.

**3. Testing approach**

**• What are the key factors?**

o Test early and often

o Divide the project into small functional modules.

o Effectuate test case development.

o Identify the right metrics for assessing test results.

o Custom test environment

o Focus on task allocation

o Emphasize on effective bug reporting

• What are the key risks?

o Lack of personnel resources when testing is to begin.

o Lack of availability of required hardware, software, data or tools.

o Late delivery of the software, hardware or tools.

o Delays in training on the application and/or tools.

o Changes to the original requirements or designs.

o Complexities involved in testing the applications

|  |  |
| --- | --- |
| **Risk** | **Mitigation** |
| Team member lack the required skills for website testing. | Plan **training course** to skill up your members |
| The project schedule is too tight; it's hard to complete this project on time | Set **Test Priority** for each of the test activity. |
| Test Manager has poor management skill | Plan **leadership training** for manager |
| A lack of cooperation negatively affects your employees' productivity | **Encourage** each team member in his task **and inspire** them to greater efforts. |
| Wrong budget estimate and cost overruns | Establish the **scope** before beginning work, pay a lot of attention to project planning and constantly track and measure the progress |

**• What are the success criteria?**

o Test early and often

o Divide the project into small functional modules.

o Effectuate test case development.

o Identify the right metrics for assessing test results.

o Custom test environment

o Focus on task allocation

o Emphasize on effective bug reporting

o Peer programming

**• What are the contingency plans?**

Requirements definition will be complete by January 29, 2019, and, if the requirements change after that date, the following actions will be taken:

o The test schedule and development schedule will move out an appropriate number of days. This rarely occurs, as most projects tend to have fixed delivery dates.

o The number of tests performed will be reduced.

o The number of acceptable defects will be increased.

o Resources will be added to the test team.

o The test team will work overtime (this could affect team morale).

o The scope of the plan may be changed.

o There may be some optimization of resources. This should be avoided, if possible, for obvious reasons.

**• What is the item pass/fail criteria?**

The success criteria of the tests in the test plan are defined as when the tester test the API it should pass if not notified to the developer about the same. We can encounter three situations while executing the test cases – normal, suspension, resumption.

Let’s look at the item pass/fail criteria for our DoD application:

o Suspension Criteria: Any situation which impedes the ability to continue testing or value in performing testing lead to suspend testing activities.

o Resumption Criteria: When the problem that caused the suspension had been resolved, testing activities can be resumed.

o Approval Criteria: An item will be considered as ‘Pass’ if it meets the ‘Expected Outcome’ defined in the corresponding test case

**• What are entry criteria?**

o development phase has been finished

o requirements have been defined and approved

o test design and tests plan have been created

o test environment has been set up

o all necessary resources are available.

**• What is exit criteria?**

It specifies the criteria that denote a successful completion of a test phase. The exit criteria are the targeted results of the test and are necessary before proceeding to the next phase of development. Example: 95% of all critical test cases must pass.

Some methods of defining exit criteria are by specifying a targeted run rate and pass rate.

o Run rate is ratio between number test cases executed/total test cases of test specification. For example, the test specification has total 120 TCs, but the tester only executed 100 TCs, So the run rate is 100/120 = 0.83 (83%)

o Pass rate is ratio between numbers test cases passed / test cases executed. For example, in above 100 TCs executed, there’re 80 TCs that passed, so the pass rate is 80/100 = 0.8 (80%)

o failed test cases are not related to crucial functionality

o tests results have been accepted

o critical defects have been fixed.

This data can be retrieved in Test Metric documents.

o Run rate is mandatory to be 100% unless a clear reason is given.

o Pass rate is dependent on project scope but achieving high pass rate is a goal.

o So, incase run rate is 90% then exit criteria is not passed as mandatory run rate is 100%

**• What are your testing criteria and checkpoints?**

TBD

**• What are the test deliverables?**

o Pass/Success rate.

o Failure Rate

o Issues

o Risks

o Solutions

**• What is the testing budget?**

o Testing budget is not decided’

o As we are using Open Source Software Tools for UAT/DEV automation testing

o Manual testing requires no cost

**• What tools are you going to use?**

o Unit testing tools?

o Configuration management tools?

o Bug tracking tools? - GitHub

**• What is your automation strategy?**

TBD

**• Which types of testing are you performing? What are the methodologies and techniques for each?**

o Requirement reviews? - Yes

o Design reviews? - Yes

o Unit testing? - TBD

o OATS? - No

o Integration testing? - Yes

o Usability testing? - Yes

o Performance testing? - Yes

o Reliability testing? - Yes

o System testing? - Yes

o Alpha/beta tests? - Yes

**• Describe your test platform:**

o We have a fully function test platform on GCP

• **Describe how you will measure the progress of testing? What reports and what metrics will you use? Include sample reports in your document**.

TBD

**• How will you determine when you're ready to ship the system?**

TBD

**4. Schedule**

• As the project started on 22nd Jan 2019 we will be delivering it by May 7, 2019 as a deadline.

**5. Approvals**

• Prof. Richard Kempinski