Medi Finder

Planning: Master Test Plan

Test Scope

The test plan is designed to cover the following:

1. White Box Basis Path and Data Flow testing
    a. This will be done to test the alternative paths data can take within the different endpoints
2. Black Box testing of all the endpoints
    a. All the backend endpoints shall be tested for accepting the expected input data, behaving properly(i.e. Valid data to success and exception handling), and generating expected output correctly)
    b. The inputs passed will also include edge cases and alternative cases to perform exhaustive black box testing and see if any boundary value analysis needs to be performed
3. The test plan will eventually cover all high-level methods and the individual units composing those methods in the backend endpoints

Test Approach and Objectives

Pre-conditions:

1. Each backend endpoint of the ASP.NET Core application will be treated as an individual unit. Due to the structure of the codebase, testing the controllers will ensure that all subsequent methods(units) called are working properly
2. Driver classes are created for each type of unit test, which will then pass data(including edge cases) to the test unit under different conditions. Stubs are used wherever possible/necessary

Test Name	Description and Objective	Comment
Integration Test	Performed on subunits that form clusters to check all functionality end-to-end	May not be necessary for every endpoint in our use case due to the specific type of structure of code achieved
Regression Test	Performed after any modification to a unit/endpoint to make sure any untouched functions didn't get adversely impacted by	Tests are adjusted accordingly during the duration of project

	project related work	
Validation Test	A series of tests to be performed on the deployed system(Alpha testing) to make sure the stakeholder requirements are met	Undertaken after the aforementioned tests are completed and the product is in a suitable state
Beta Test	To be considered at a later stage to gauge user experience reviews and identify missed test cases	
Security Test	Script injections to test data security	
Performance Test	Performed using Chrome Dev Tools and identify key areas of performance improvement	
Interface Test	Performed on front-end by a group of users to test content, navigation and other front-end components	

Resources(tools, system, and databases)

Tools	Description	Version
System		
Database		

Test Roles and Responsibilities

Resource	Role	Responsibility
Bayan Assali	Tester	● Execute test cases, logs, defects, and document test results
Evan Lim	Project Manager	● Provide technical direction ● Acquire testing resources ● Assign tasks ● Communicate results to stakeholders
Hasin Shabbir	Test Designer	● Evaluate the effectiveness of test effort ● Ensure test environment and assets are installed, managed and maintained
Kartikey Singhal	Test Designer	● Document test plan ● Identify, document and prioritize test cases
Muaath Asali	Tester	● Execute test cases, logs, defects, and document test results

Test Milestones

Task	Date(DD/MM/YYYY)
Setup Test Environment	11/11/2022
Complete Unit Testing	03/12/2022
Complete Integration Testing	05/12/2022
Complete Regression Testing	05/12/2022
Complete Validation Testing	05/12/2022
Complete Beta Testing	05/12/2022

Complete Security Testing	05/12/2022
Complete Performance Testing	06/12/2022
Complete Interface Testing	06/12/2022