Project name: Airline Booking

Test engineer: Sakshi Kumari

Date: 1/165/2023

Prepared by: Scholars

Reviewed by: Nageshwara Rao

**1)Test objective or Aim:**

The objective of the test is to verify that the functionality of **AIR TICKET BOOKING APPLICATION** and search flights functionality works according to the specifications.

Final product of the test includes:

* Production ready software
* A set of stable test script that can be reused for the functional and UAT test execution.

**2) Scope of testing:**

|  |  |  |
| --- | --- | --- |
| Module | Application | Description |
| Registration | Customer | Customer able to register with their details |
| Search flight | Customer | Customer able to search the flights |
| Book the ticket | Customer | Customer able to book the flight |
| Cancel flights | Customer | Customer is able to cancel the booked flights |
| Admin | Admin | Admin is able to add new flights |
| Main menu | Customer | Customer is able to choose which module they need. |

1. within the scope:

* Functional Testing
* Unit Testing
* Smoke Testing
* Sanity Testing

1. Out of scope:

These feature are not to be tested because they are not included in the software requirement specs

* Performance Testing

**3) Test strategy:**

a) Levels of testing:

System Testing: It is end to end testing where testing environment is similar to production environment. We just test the end feature. We do not check data flow or do functional testing and all.

User acceptance Testing:Here, they use the software for business for particular time and checks whether the software can handle the real time scenarios. Completely done by end users.

Application Under Test (AUT):

 After designing and coding section of development cycle, when the application(build) comes under testing then at that time application state is under test. The benefits of testing include **preventing bugs, reducing development costs and improving performance**. 

 b) Types of Testing

* Black Box Testing:

Technique of software which examines the functionality of software without out looking into internal structure or coding. Checks whether function is expected output /not*.*

* Functional Testing:

Here, in this testing we the verify the functionality of the search flights.

* Smoke testing:

Testing the basic or critical feature of an application, before doing thorough one round of rigorous testing. In this, we can only focus in positive flow of application and enter only valid data, not the invalid data.

In smoke testing, we verify every build is testable or not;

* Sanity Testing:

Generally, sanity testing is performed on stable builds and it is also known as a variant of regression testing. It is a checkpoint to assess if testing for the build can proceed or not. 

* Exploratory Testing:

We will explore the application in all possible ways and understand the flow of the application, preparing a test document and then testing the application. We test that application is functioning as per the available list of buses and their exact recorded timings with all possible conditions

c) Test Design Techniques:

* Boundary Value Analysis(BVA)
  + We are going to check the boundaries values of the minimum and values
  + maximum(Minimum-1) (Minimum+1) and (Maximum value-1) (Maximum+1)
* Equivalence Class Partitioning(ECP)
  + - Partition the data into various classes and we can select according to class then test.
    - It reduces the number of test cases and saves time for testing
* Error Guessing.
  + Technique used to find bugs in a software application based on testers prior experience.
  + In guessing errors when we don't follow specific rules.
  + Depends on Tester analytical skills and experience.

  d) Configuration Management tool:

* Git (Code Configuration Management) (Code CM)
* SVN- Document Configuration.

   e) Terminology

* Test Plan
* Test Case
* Test Scenario
* Defect Log
* RTM-Requirements Traceability Matrix

    f) Area planned for Automation:

Automation is out of scope.   
   g) List of automation tools

 Automation is out of scope.  
**4) Exit and Entry criteria**

     A) Entry Criteria

The entry criteria are need to be done after the code implement is performed. Complete or Partially testable code is available.

* Requirements are defined and approved.
* Availability of sufficient Test data
* Test cases are developed and reviewed.
* Test environment is ready.

    B) Exit Criteria

* 100% test scripts executed.
* Pass rate is equal to 95%.
* No critical defects left.
* 95% of medium severity defects were closed.
* Remaining bugs were fixed.

**5) Test deliverables:**

|  |  |  |
| --- | --- | --- |
| Before Testing | During Testing | After Testing |
| Test Plan Document | Test Tool | Test Results & Reports |
| Test Case Document | Test Data | Defect Reports |
| Test Design Document | RTM | Installation Guidelines |

**6) Roles and Responsibility**

|  |  |
| --- | --- |
| Roles | Responsibilities |
| Test Manager | 1. Manage the whole Project. 2. Define Project direction 3. Risk Management. |
| Test Engineer | 1. Write the test cases. 2. Execute the test cases. 3. Report the defects. 4. Identifying the test design techniques. 5. Identifying the severity of the defect. |
| Senior QA | 1. Taking in-charge of Quality assurance. 2. Confirms whether the testing process is meeting specified requirements. |
| Developer in test | 1. Implementation of test cases 2. Writing automation scripts |

**7) Risks and mitigation**

  a) Risk and contingency

1. <Customer>will endeavor to meet prerequisites indicated by.
2. <Customer>will ensure a full set of suitable and protected test data is available.
3. <Tester>will indicate what is required and will verify suitability of test data.

b) Risk and mitigation

1. Meet outstanding prerequisites
2. Redefine test data
3. Review test plan and modify components (that is, scripts)
4. Restore data and restart

**8) Schedule**

|  |  |  |
| --- | --- | --- |
| Task | Members | Estimate effort |
| Create the test specification | Test designer | 170 man-hour |
| Perform test execution | Tester, Tester administrator | 80 man-hour |
| Test report | Tester | 10 man-hour |
| Test delivery | Test administrator | 20 man-hour |
| Total |  | 280 man-hour |

**9) Hiring and Training**

* Minimum 2 years of experience in manual and automation testing.
* Programming knowledge in Java and Python.
* Data Base Knowledge preferred.
* 3 weeks of training should be given under the domain and application.
* Manual Testing
* Application
* Automation

**10) Test Environment**

|  |  |  |
| --- | --- | --- |
| No. | Resources | Descriptions |
| **1.** | Server | Need a Database server. |
| **2.** | Test tool | Develop a Test tool which can auto generate the test result to the predefined form and automated test execution |
| **3.** | Network | Setup a LAN Gigabit and 1 internet line with the speed at least 5 Mb/s |
| **4.** | Computer | At least 4 computer run Windows 7, Ram 2GB, CPU 3.4GHZ,Chrome version-27 & above. |
| **5** | MS Excel | Test case preparation, test case execution, defect management, test reporting, checklist of test, |

**11) Assumptions**

* Exploratory Testing would be carried out once the build is ready for testing
* Performance testing is not considered for this estimation.
* Test case design activities will be performed by QA Group
* Test environment and preparation activities will be owned by Dev Team
* Development team will provide Defect fix plans based on the Defect meetings during each cycle to plan.

**12) Approval Information**

* Project Manager: reviews the content of the Test Plan, Test Strategy and Test estimates signs off on it.
* Test Manager: Reviews the test cases, test conditions and Test data, test report
* The Names and Titles of all persons who must approve this plan.

**Signature:**

* Name: Sebastain.
* Role:   Team leader.
* Date:16-01-2023

**13) Test Metrics**

* Passes Test Cases Percentage: **(No. of Passes Tc/No. of Executed Tc) \*100**
* Failed Test Cases Percentage:**(No. of Failed Tc/ No. of Executed Tc) \*100**
* Accepted Defect Percentage:**(No. of Accepted defects/Defects Reported) \*100**
* Defects Deferred Percentage: **(Defects Deferred/Defects Reported) \*100**