Quality Assurance Plan

Software Development

Final Project - Baggage Recovery App

Wei Liu
Ziwei Dai
Lixin Chen

TABLE OF CONTENTS

1. Quality Assurance 3

2. Unit Testing 3

3. Integration Testing 4

4. Customer Trial Period 4

1. Quality Assurance

This section describes policies and procedures that will be used to meet QA program objectives which are also the strategies the Baggage Manager Application will use to ensure the quality of the project. The strategies cover the whole phases of the development of this project, the detailed or specific operations for quality assurance include unit testing, integration testing, customer trial period, etc.

2. Unit Testing

Unit tests are used to test a minimal functional unit, such as a single method (giving a class with a specified state, then calling the x method of that class, and finally checking if the state is as expected). Unit tests should focus on a particular function (for example, calling a pop method on an empty stack throws an InvalidOperationException). Unit tests should all be executed in memory, which means that the test code and the code being tested should not:

- Call into other methods
- Access to the network
- Access the database
- Use the file system
- Create new thread

The preferred way to write tests in Django is using the unit test module built into the Python standard library. This is covered in detail in the Writing and running tests document of Python Documentation.

The strategy the team adopted was to carry out unit test after the implement of each method and every single class and a documentation of the testing result should be continuously integrated into a complete test report.

3. Integration Testing

Integration testing checks the combination of code units and test units to see if they are correct. This can be done inside the system or a combination of multiple systems. Another difference from unit testing is that integration testing can use threads, access databases, or ensure that all code and different environments will work correctly.

After every continuous integration, the team should carry at least one integration test. The industry best practice about the continuous integration is at least one continuous integration per day, in best cases several times a day.

The continuous integration should be conducted in several steps:

- Keep all source code in a source code. repository
- Pick up the software and it changes and starts building the checked in the project.
- If the build is a failure, the team will be notified accordingly otherwise it is tested for any defects.
- If the test fails, the developers are notified, otherwise, the build is ready to be deployed.

4. Customer Trial Period

In the project plan, after the development phase, the application would be delivered to the market department and would have a 5-month customer trial period. During this period, the marketing department will continue to receive feedback from customers and pass it on to the development team. The development team will respond to customer feedback (including feature requirement, bug report and other feedbacks) and continuously improve the product.