# Take Me Out to the Ballgame

**Test Plan: CS1D Project 2**

**Table of Contents**

#### Introduction

* 1. Purpose
  2. Team Members

#### Scope

* 1. Overall Scope

#### Test Approach

* 1. Overall Test Strategy
  2. Features Affected in User’s Perspective
     1. Login interfaces
     2. Logout
     3. Adding a Souvenir to a Stadium
     4. Removing a Souvenir from a Stadium
     5. Purchasing Souvenirs while in a trip
     6. View the MST
     7. View Existing Stadium Info
     8. Admin Edit Existing Team Data
     9. Admin Import New Team
  3. Entry Criteria
  4. Exit Criteria
  5. Suspension Criteria

#### Test Environment

1. **Deliverables / Approval Process** 1.
2. **Reference**

1.

**Introduction**

### Purpose

* + - The purpose of this test plan is to thoroughly test all of the user functionalities whether the user is an Admin or just a Normal User. Each type of user will have access to unique abilities so each user is only able to access their unique functionalities.
    - Purpose ­ to achieve a complete, exhaustive and thorough analysis of the code infrastructure to ensure the most robust execution with minimal bugs and utmost accuracy.

This paper will document Multiple test cases Security infrastructure Code integrity

Elegant, simple and clear documentation

### Team Members

|  |  |
| --- | --- |
| Team Members | Role |
| Daniel Reisch | Scrum Master |
| Randy | Product Owner |
| Sean | Developer |
| Gabe Kutasi  Kai Duty | Developer  Developer |

* 1. Objective
* Thoroughly test user functionality whether the user is an Admin or a normal user who does not have access to Admin privileges.
* Black box test - Test functionality without any knowledge of internal code
* Regression test - Finding defects after a major code change has occurred
* Unit Test - Verify the functionality of a specific section of code

## Scope

### Overall Scope

This Test Plan describes the integration and system tests that will be conducted on the project two. The tests will be held on viewing unique stadium information, having the ability to sort information given, have the ability to take a trip to visit different stadiums in the shortest possible manner, and Admin abilities to manipulate existing data.

It is assumed that unit testing already provided thorough black box testing, extensive coverage of source code, and testing of all module interfaces.

The purpose of assembling the test plan was to test feasibility and performance of the selected architecture. It is critical that all system and subsystem interfaces be tested as well as system performance at this early stage. Testing of system functionality and features will not be conducted on the prototype.

The interfaces between the following subsystems will be tested:

* + 1. Login interfaces
    2. Logout
    3. Adding a Souvenir to a Stadium
    4. Removing a Souvenir from a Stadium
    5. Purchasing Souvenirs while in a trip
    6. View the MST
    7. View Existing Stadium Info
    8. Admin Edit Existing Team Data
    9. Admin Import New Team

The external interfaces to the following devices will be tested:

1. Login interfaces
2. Logout interfaces

## Test Approach

1. Changing screens from products

### Test Strategy

The Test Strategy presents the recommended approach to the testing of the software applications. The previous section on Test Requirements described *what* will be tested; this describes *how* it will be tested.

The main considerations for the test strategy are the techniques to be used and the criterion for knowing when the testing is completed.

In addition to the considerations provided for each test below, testing should only be executed using

1. Black box testing the code allowing the right requirements for the project
2. Sending error messages to all the user to understand that they are not in putting the right data
3. Weekly updates to the code and keeping track of all the code being tested

### Features Affected in User’s Perspective

* + 1. View Team’s Data on Command

The strategy will be to black box the ability to view all the available team’s data. This will allow the user to determine if a specific stadium is worth traveling to.

### Sort Team’s Data On Command

The strategy will be to black box the ability to sort available team’s data whether it’s sorting by alphabetical order, sorting by each stadiums seat

capacity, or sort by each stadium’s opening date. Doing and sort will not manipulate existing data.

### Filter Team’s Data On Command

The strategy will be to black box the ability to filter available team’s data whether it’s filtering by team leagues (American League or National League), or filter by the typology of the stadium’s,

### Login to Admin

The strategy will be a black box test to test the login to only take in the username and password we gave it not allowing anyone to login into the admin page if the wrong username or password is entered it will not allow them to pass once entered correctly the page will change to the admin page

### Buying a Product

The strategy will be to black box test to test buying a product function allowing them to find the product they want to buy. Once found they will be able to place an order and the amount of orders they want. Having them only to order a right amount to their company

### Viewing the Minimum Spanning Tree

The strategy will be to black box the functionality of the user to view the Minimum Spanning Tree that was calculated and displayed to the GUI for easy access to the user. The overall MST should not manipulate existing data and will have the possibility to change if data is manipulated by the Admin

### Admin Edit Existing Team Data

The strategy will be to black box the functionality of the Admin to be able to edit an existing Team’s data if needed.

### Admin Edit Import New Team

The strategy will be to black box the functionality for an Admin to import a new team with it’s corresponding information needed.

### Exit criteria

The exit criteria will be tested and determined that the project application has been satisfactorily completed before exiting the system test stage and clarifying the application as complete.

### Entry criteria

The entry criteria will be testing when all group members are up to date on the project and to allow all the users to understand the project and to understand all the backlog code

### Suspension Criteria

Application is manually closed by customer which will result in no data being affected

## Test Environment

The listing below identifies those items (use cases, functional requirements,

non­functional requirements) that have been identified as targets for testing. This list represents *what* will be tested.

* + 1. To know C++
    2. User standing QT
    3. Functions of QT
    4. Computer for testing

## Deliverables / Approval Process

### 5.1. Test Deliverables

Application Testing ­ Working Pages Login to Admin

Buying a Product

Viewing All Team’s Available Data Sort Avaliable Team Data on Command

Filter Avaliable Team Data on Command View MST

Admin Edit Existing Team Data Admin Add/Delete Souvenirs Admin Import New Team

### 5.2 Schedule

The testing will be done every day

## References

Applicable references are:

* 1. UML Modeling Class Diagram
  2. State diagram
  3. Use Case #1
  4. Use Case #2
  5. Use Case #3