Urban Hunt

Test Plan

# Project Overview

Urban Hunt will allow users to connect with others in their area to play games in real time based on the GPS location of their mobile phone. Many different game modes will be provided, revolving around users interacting with each others’ location data. Some example game modes for the application are: Capture the Flag, GPS Tag, scavenger hunts, “Where’s Waldo”, races, and team relays. Many of our anticipated users would be interested in fitness, however the finished product may include less physically intensive games based on the GPS location.

# Unit Test Cases

We will use Python scripts to implement tests on the server. We are using Android Java tests from the application itself.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Description** | **P/F** | **Notes** |
| 1 | Test Opening Socket | P | Localhost connection |
| 2 | Test Connecting to Server | P | Server connection |
| 3 | Test Sending Data to Server | P |  |
| 4 | Test Receiving Data from Server | P |  |
| 4 | Test Updating Marker | P | Needs LatLng param. |
| 5 | Test GPS System | P |  |
| 6 | Test Add Player | P | Overwrites player name “Test” |

# System Test Cases

Our system test cases will test each feature as it moves from the test branch into master. Once a feature set has been moved from dev into test, system test cases will be written to test those features, and once those system test cases pass the feature may be moved into master. The tests will be run by whichever member of the team is maintaining the feature.

1. Create a new Game

Description/Objective: This test should verify that users can create a new game.

Requirements: 1.1 - Display a screen that allows users to host a new game

Pre-Test: Install the app on a phone / emulator.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Action** | **Expected Result** | **P/F** | **Notes** |
| 1 | Launch the App | Launch screen comes up | P |  |
| 2 | Click “Start New Game” | New Game Screen Comes up | P |  |
| 3 | Fill out Data | Data is saved in App. | P | Saved in Game Class |
| 4 | Click “Submit” | Map launches and game starts. | P |  |

1. Join a Game

Description/Objective: Verify that users can join an existing game

Requirements: 1.2 - Allow users to join an existing game.

Pre-Test: Have another instance of the app running a game.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Action** | **Expected Result** | **P/F** | **Notes** |
| 1 | Launch the App | Launch Screen comes up. | P |  |
| 2 | Click “Join Game” | Join game screen comes up | P |  |
| 3 | Fill out data | Data is saved in the app, game code is verified on server. | F | Server does not verify game code. |
| 4. | Click Submit | Map launches and game is run. | P |  |

1. Test GPS

Description/Objective: Verify that the GPS functions.

Requirements: 2.5, 3.2

Pre-Test: App Installed and Server Running

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Action** | **Expected Result** | **P/F** | **Notes** |
| 1 | Create / Join a game | Map is displayed | P |  |
| 2 | Turn on Location Services | Marker updates to current position | P | Location Services may be enabled by default. |
| 3 | Move GPS Sensor | Marker continues to update | P |  |

1. Test Player Data Store

Description/Objective: Test to see if the Server is storing the data players.

Requirements: 1.3, 2.2, 2.4, 3.2, 3.3, 3.4

Pre-Test: App installed, server running in debug mode.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Action** | **Expected Result** | **P/F** | **Notes** |
| 1 | Create/Join a game | Game starts | P |  |
| 2 | Send location data to server | Socket opens, data is sent. | P | This step happens automatically. |
| 3 | Verify sent data on server. | Server will print out data received. | P |  |
| 4 | Verify data sent from server | Server will print out locations of other players. | P |  |

1. Test Game Rules

Description/Objective: Verify that users do not break the rules.

Requirements: 2.1, 2.3, 2.6

Pre-Test: App installed and server running.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Action** | **Expected Result** | **P/F** | **Notes** |
| 1 | Start a new game | Map displays and game starts | P | Currently only one game mode is available. |
| 2 | Add players to game | Other players appear on map | P |  |
| 3 | Move two users into close proximity | Collision event is called | F |  |
| 4. | Tag all players | Game ends | F |  |

1. Test Server Storage

Description/Objective: Verify that the server is properly storing game data

Requirements: 3.1, 3.2, 3.3, 3.4, 3.5

Pre-Test: Run the server in debug mode.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Action** | **Expected Result** | **P/F** | **Notes** |
| 1 | Start a game on the app. | Game starts | P |  |
| 2 | Add other players to game. | Users appear in app. | P |  |
| 3 | Check Server data stream. | Entries should appear for each player, team, and game currently being played. | P |  |