MIDDLE EAST TECHNICAL UNIVERSTIY

DEPARTMENT OF COMPUTER ENGINEERING

SOFTWARE TEST DOCUMENT

prepared by

Flaming Swallow

for project SINCAP

METU - Department of Computer Engineering

CENG 492 Senior Design Project

1.Introduction

1.1. Project Definition

We aim to develop an application that uses real world data for playing games like scavenger hunt which is one of many different types of games which can have one or more players who try to find hidden articles, locations or places by using a series of clues. The project will be worldwide. We will keep the GPS coordinates of the locations where the game will take place. The player will be able to collect clues by moving throughout the game location in real life. When the player goes to a game location, he/she will receive a notification by the game and will try to find a clue by looking through phone's/tablet's camera. The game has also has multi-player aspect as players will be able to interact with each other if they are close by. They will be playing a trivia game and the winner gets a bonus while loser gets a penalty. The game may have different modes like RPG, time attack, PvP. In RPG mode, there may be monsters obstructing the roads and players will need to beat them to open the roads or have to take a detour. In Time Attack mode, there will be no monsters, however players will try to get the best time. In PvP mode, players will be able to challenge their friends to a game ad only that group is going to play that scenario.

1.2. Document Identifier

Flaming Swallow prepared this document with the purpose of testing the “Sincap: An Augmented Reality Game” software. The document explains design of test cases and proceure with great detail so that any tester would be able to run and observe the outcomes. This document is the first version of STD.

1.3. Purpose and Scope

The purpose of this document is to provide the test cases of the “Sincap: An Augmented Reality Game” project. It defines the objective, scenario, expected outcomes and procedural requirements for each test case. It also includes a table showing which test case is realted to which one. The software will be tested using guidance of this document.

In the first section, the background information about “Sincap: An Augmented Reality” and roughly the aim of the document are provided. The second section is dedicated to the information about test cases. The third section is about test management. The fourth section is about functional tests of all system in a detailed manner and using table system. Finally, last two chapters, form main part of the document, firstly explain the testing tools and results of system testing part.

1.4. References

(Sincap) Software Requirements Specification
(Sincap) Software Design Descriptions
IEEE STD 829-2008

1.5. Definitions, Acronyms & Abbreviation

SDD	Software Devolopment Document
Sincap	A mobile augmented reality game
GPS	Global Positioning System
METU	Middle East technical University

1.6. Level in the Overall Sequence

Project itself, together in order to ensure that they work correctly as a system, is described.The hierarchy between different level testing methods such as unit testing, which checks the correctness of the individual modules of the software; or integration testing that verifies compatibility between software elements of the project, does not take part in this document. Only functional testing, which is defined as the testing of individual parts of the project which becomes the project itself, together in order to ensure that they work correctly as a system, is described.

1.7. Test Classes and Overall Test Conditions

In the next sections of the document, many test cases are given a place for checking the software in terms of different perspectives.

2. Details for System Test Plan

2.1. Features to Be Tested

Projects functional and non-functional properties will need to be tested in METU main campus where the game scenarioes are developed.While testing envrinonmental variables ,the previously specified input parameters will be used to observe the expected and unexpected behavior of the game that can be occured.

2.2. Features Not to Be Tested

During the application testing process, GPS connection and MobileData speed will not be tested since these attributes are out of our control.

2.3. Approach

Clear box testing, glass box testing and structural testing are used based upon tester knowledge of internal perspective and implemantation of “Sincap app”. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs.

2.4. Item Pass/Fail Criteria

For each tet case test outcomes have to be compared to the expected and predefined outcomes. The following section describes the pass/faill criteria for each of the items descrived in Test items.

1. 90% of test cases must pass
2. All test cases dealing with critacal functionality must pass
3. All medium and high severity defects must be fixed

4. Test coverage must be at least 90%

2.5. Test Deliverables

This document contains test procedure specifications that explain the steps for executing all test cases given in the following sections and the results of each test case. It is prepared basen on IEEE standard for Software and System Test Documentation.

3. Test Management

3.1. Planned Activities and Task

For executing test cases that specified in this document in detail, tester should know the game dynamics and requirements that to be found in earlier project documents. These documents is available on the website of Sincap. Tester also should know briefly informations about augmented reality based world and enviroments in which the game takes place.

3.2. Environment

Since our project is based on mobile platforms, a mobile phone with android os is a must to play the game. Since our application uses the camera and gps of the phone, phone should have camera and gps sensor. Because of connection between phone and server, Wi-Fi or mobile data is necessary.

4. Performance and Tools

Since the project is a mobile application and uses the server for getting map, locations and story, response time is fast enough for communication between server and applcation. However, application checks the user’s location for 15 secs.

5. Test Case Details

5.1. Introduction

This section embodies the detailed explanation for each test case accompanied by the objective, scenario, expected outcomes and procedural requirements along with the dependencies among test cases. All devices that are going to be used should be smart phones with android os 4.3(API level 18) or higher.

This section includes the information for all test cases we run on the project. For each test case test there are four fields; objective, scenario, expected outcomes and procecdural requirements.

Objective explains why the test is run on the project. Scenarios are what inuts are to be given to the program in order to see the behavior of the program. Expeted outome is what we desire to see after test is applied. There are no specific environmental needs, but there are special procedural requirements and they are explaind in their sections.

5.2. Test Cases

5.2.1. Register Test

Objective:	Asserting a user can register to the system
Scenario:	1. User taps the register button in the first screen 2. User enters his/her credentials 3. User taps the register button
Expected Outcome(s):	1. System saves user to the system 2. Android application logs in the user
Procedural Requirements:	1. No user should be logged in on that device at that time 2. User should at least fill email, password, nickname fields

5.2.2. Login Test

Objective:	Asserting a user can login to the system
Scenario:	1. User enters his/her email and password 2. User taps the login button
Expected Outcome(s):	1. System saves users login info 2. Android application logs in the user
Procedural Requirements:	1. No user should be logged in on that device at that time

5.2.3. Choose Game Mode Test

Objective:	Asserting a user can choose between available game modes
Scenario:	1. User taps modes button 2. Users taps an element listed in a table
Expected Outcome(s):	1. Android app sends list mode request to the web server 2. Server responds with available modes 3. Android app saves chosen mode information inside user information
Procedural Requirements:	1. User should be logged in to the system

5.2.4. Choose Game Region Test

Objective:	Asserting a user can choose between available game regions

Scenario:	1. User taps regions button 2. User taps an element listed in a table
Expected Outcome(s):	1. Android app sends list regions request to the web server 2. Server responds with available regions 3. Android app saves chosen region information inside user information
Procedural Requirements:	1. User should be logged in to the system

5.2.5. See Scoreboard Test

Objective:	Asserting a user can see the overall scoreboard of registered users
Scenario:	1. User taps scoreboard button
Expected Outcome(s):	1. Android app sends list scoreboard request to the web server 2. Server responds with part of the scoreboard where user is currently resides
Procedural Requirements:	1. User should be logged in to the system

5.2.6. Change Settings Test

Objective:	Asserting a user can change the settings of the game
Scenario:	1. User taps the settings button 2. User makes changes in the screen
Expected Outcome(s):	1. Android app shows the current settings 2. Android app saves the current settings inside user information

Procedural Requirements:	1. User should be logged in to the system

5.2.7. Start Game Test

Objective:	Asserting a user can start a game
Scenario:	1. User taps a game listed in a table
Expected Outcome(s):	1. Android app starts new window with a map 2. Android app shows the borders of the placemarks of that region saved in the database 3. Android app shows the first clue to the user
Procedural Requirements:	1. User should be logged in to the system 2. User should choose a region 3. User should choose a mode 4. User should choose a game

5.2.8. Find Location Test

Objective:	Asserting the Android app can find user’s location accurately
Scenario:	1. User taps check location button 2. User checks if his/her real location is accurate
Expected Outcome(s):	1. Android app estimates user’s location 2. Android app shows users location
Procedural Requirements:	1. User should be logged in to the system 2. User should start a game

5.2.9. Show Object On Camera Test

Objective:	Asserting the Android app can show an object over camera feed
Scenario:	1. User taps show hint button
Expected Outcome(s):	1. Android app opens the camera 2. Android app puts an object to the feed 3. When user changes the orientation of the phone Android app rotates the object
Procedural Requirements:	1. User should be logged in to the system 2. User should start a game

5.2.10. Advance to the Next Stage Test

Objective:	Asserting a game can advance to the next stage
Scenario:	1. User taps check place button
Expected Outcome(s):	1. Android app finds users location 2. If user is in the correct location, Android app shows the next clue to the user
Procedural Requirements:	1. User should be logged in to the system 2. User should start a game 3. User should be in the correct location

5.2.11. Create Game Test

Objective:	Asserting a user can create a game
Scenario:	1. User visits game’s create game webpage 2. User enters game credentials

	3. User clicks submit button
Expected Outcome(s):	1. Server shows available region and mode information 2. Server saves the new game in the database
Procedural Requirements:	1. User should at least enter name and region fields

5.2.12. Add Story Piece Test

Objective:	Asserting a user can insert story pieces to a game
Scenario:	1. User visits game’s create piece webpage 2. User enters story credentials 3. User clicks submit button
Expected Outcome(s):	1. Server shows available locations, games information 2. Server saves the new story piece to the database
Procedural Requirements:	1. User should at least enter title, game, location, clue fields 2. User should enter valid location, game fields

6. System Test Report Details

6.1. Overview of Test Result

In this document, the acquired results for each test case are evaluated with respect to pass/fail criterias that are also defined in the document.

Test Case	Results
5.2.1	Passed
5.2.2	Passed
5.2.3	Passed
5.2.4	Passed
5.2.5	Passed
5.2.6	Passed
5.2.7	Passed
5.2.8	Passed
5.2.9	Passed
5.2.10	Passed
5.2.11	Passed
5.2.12	Passed
5.2.13	Passed
5.2.14	Passed
5.2.15	Passed
5.2.16	Passed

5.2.17	Passed

Detailed Test Results

6.2. Rationale For Decisions

The test cases were formed to see the behavior of the code in prone parameters and inputs. The test cases were tested number of times and behavior of the system was observed using our predefined evaluation parameters.

6.3. Conclusion

For conclusion, the results of all tests are successful , but this is not necessarily means that there is no bugs or no defects in our system. We could not perform some of the planned test due to missing implementations.