1. **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 25/10/2018 | 1.0 | <details> | Alp Demirezen |
| 26/10/2018 | 1.1 | Added inspection and demonstration tests | Ataberk Ozek |
| 26/10/2018 | 1.2 | Added analysis and test sections | Mert Buyukiz |
|  |  |  |  |

TABLE OF CONTENTS

[1](#_gjdgxs) Revision History 1

[Introduction 3](#_30j0zll)

[1.1](#_1fob9te) Document overview 3

[1.2](#_3znysh7) References 3

[1.3](#_2et92p0) Conventions 3

[Tests preparations 4](#_tyjcwt)

[1.4](#_3dy6vkm) Hardware preparation 4

[1.5](#_1t3h5sf) Software preparation 4

[1.6](#_4d34og8) Other test preparation 4

[1.7](#_2s8eyo1) Safety, security and privacy precautions 4

[3 Tests descriptions 5](#_17dp8vu)

[1.8](#_3rdcrjn) REP Tests 5

# Introduction

## Document overview

This document is the software test plan of the XXX software development project. It contains the description of tests.

The structure of this template is simple compared to others. But don’t underestimate the effort to fill it. Tests descriptions may give you headaches!

## References

|  |  |  |
| --- | --- | --- |
| # | Document Identifier | Document Title |
| [SRS] | Real Estate Application-SRS-1 | Real Estate Application Software Requirements Specifications |
|  |  | Add your documents references.  One line per document.  There should be a reference to the SRS document at least; because each test must be linked to a requirement. |

## Conventions

Each test case will be presented in a table format as shown below:

|  |  |  |
| --- | --- | --- |
| **Test ID** | <ID of the test> |  |
| Test description | <A brief description of the test> |  |
| Verified Requirement | <ID numbers of requirements that are to be verified with this test.> | Verification method: <I,A,D,T> |
| Initial conditions | <The state of software before test> |  |
| Tests inputs | <Input data from any test tool, input files name and location, if any > |  |
| Data collection actions | <Recording and post processing of output data, if any > |  |
| Tests outputs | <Output data files, if any> |  |
| Assumptions and constraints | If any, may be limited access to a tool, license … |  |
| Expected results and criteria | <list of expected results> | <special criteria to be met by the result, if any> |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and evaluation criteria** |
| 1 | <actions listed row by row> | <expectation after each action if any> |
| 2 | … | … |

# Tests preparations

This section contains tasks and recommendations before executing tests.

## Hardware preparation

Cpu: i7 (6700HQ)

Ram: 16 GB DDR4

Gpu: GTX 1060 (6GB)

A computer including these hardware will be used while testing.

Describe platform configuration operations, like specific hardware to be used, physical network configuration.

You may add a diagram and steps to set up hardware for use.

## Software preparation

Describe software set-up and configuration operations, like simulators or software test tool to be used, logical network configuration.

You may add a diagram and steps to set up software for use.

## Other test preparation

None

Describe here any other specific task to do before tests.

## Safety, security and privacy precautions

Add here warnings or precautions about safety, security and privacy.

Think twice about it if tests are, for example, realized in a health care center.

In the context of the course project, you can just write here “None”

None

# 3 Tests descriptions

You may organize tests by groups or list them in alphabetical order.

The most effective way is to list them in chronological order of execution for each phase. Test engineers won’t waste time searching for tests in the document.

### 3.1 REP Tests

Section name may be:

* or category of test xxx
* or some other logic to group tests

Describe each test with the pattern below.

For most of tests, only a subset of fields in the table is used, mark N/A (non applicable) the unused fields.

|  |  |  |
| --- | --- | --- |
| **Test ID** | **T-REP-REQ-001** |  |
| Test description | Verify that user can see the user interface when clicked on the exe. |  |
| Verified Requirement | SRS-REQ-001 |  |
| Initial conditions | REP was not running | Inspection |
| Tests inputs | N/A |  |
| Data collection actions | N/A |  |
| Tests outputs | N/A |  |
| Assumptions and constraints | N/A |  |
| Expected results and criteria | User can see and interact with the user interface. |  |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and evaluation criteria** |
| 1 | Click REP.exe | REP is started |
| 2 | Click on the search bar | Search bar should have a cursor |

Verification Methods:

* Inspection (I): control or visual verification
  + Control of the physical implementation or the installation of a component. The control verifies that the implementation or the installation of a component is compliant with the requirements of diagrams.
  + Control of the documentation describing a component. The control verifies that the documentation is compliant with the requirements.
* Analysis (A): verification based upon analytical evidences
  + Verification of a functionality, performance or technical solution of a component by analyzing the data collected by tests in real conditions, by simulation of real conditions or by a analysis report.
  + Analysis of test data or of design data is used as appropriate to verify requirements.
  + The verification is based upon analytical evidences obtained by calculations, like modeling, simulation and forecasting.
  + Analysis is used when an acceptable level of confidence cannot be established by other methods or if analysis is the most cost-effective solution.
* Demonstration (D): verification of operational characteristics, without quantitative measurement
  + Verifying a requirement by demonstration implies that the required functionality specified by a requirement is complete.
  + Demonstration is used when quantitative measurement is not required for verification of the requirements
  + Demonstration includes the control of the technical solutions specified by the non-functional requirements.
* Test (T): verification of quantitative characteristics with quantitative measurement
  + Verifying a functionality, performance or technical solution of a component by executing testing scenarios in predefined, controlled and traceable testing conditions.
  + Tests require the use of special equipment, instrumentation, simulation techniques, or the application of established principles and procedures,
  + Data produced during tests is used to evaluate quantitative results and compare them with requirements.

Examples of tests methods:

Inspection:

* Verify that the color of background is blue,
* Verify that the user manual has the CE mark on its cover
* Verify that the PC has 4Gb memory
* Verify that firmware version on electronic card is 1.0.1

Demonstration

* Verify that when the user closes the window, a confirmation message appears
* Verify that the file is saved in the output directory
* Verify that the result is shown
* Verify that if a value is out of range, a warning is displayed

Analysis:

* Verify that the statistical distribution of results of xxx algorithm is a Gaussian with mean=x and stdev=y, when input data are blah blah
* Verify that the linear regression of results of xxx algorithm is a line for which value is 1 on the y-axis, at zero on the x-axis,

Test:

* Verify that a file of 1Gb is processed in less than 3s
* Verify that the response time of the server is 15ms with 20 simultaneous requests

Demonstration

|  |  |  |
| --- | --- | --- |
| **Test ID** | **T-REP-REQ-002** |  |
| Test desc. | Verify that REP software can find the appropriate result with the given inputs. |  |
| Verif. Req. | SRS-REQ-001, | Demonstration |
| Init. Cond | REP Software is started and idle |  |
| Tests inputs | City, Province, number of rooms |  |
| Data collection | N/A |  |
| Tests outputs | N/A |  |
| Assum & constr | N/A |  |
| Expected results and criteria | Results are correct. |  |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and eval crit** |
| 1 | Enter the data to the desired location. | White bars allow users to write the data that needed. |
| 2 | Click search button. | Search algorithm starts. |
| 3 | Verify that the results are correct with respect to the inputs. | Search algorithm finds the correct houses. |

Demonstration

|  |  |  |
| --- | --- | --- |
| **Test ID** | **T-SRS-REQ-003** |  |
| Test desc. | Verify that theREP software allows to choose on protocol in the list of recorded protocols |  |
| Verif. Req. | SRS-REQ-002 | Demonstration |
| Init. Cond. | XXX Software is started and idle |  |
| Tests inputs | N/A |  |
| Data collection | N/A |  |
| Tests outputs | N/A |  |
| Assum & constr | N/A |  |
| Expected results and criteria | Chosen protocol is selected | See last step for criteria |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and eval crit** |
| 1 | Open the list of protocols | The list of protocol is displayed |
| 2 | Select a protocol | The protocol summary is displayed |
| 3 | Validate the choice | The protocol is selected. The name of the protocol is displayed at the top of the screen. |

Analysis

|  |  |  |
| --- | --- | --- |
| **Test ID** | **T-SRS-REQ-003** |  |
| Test desc. | Verify that the REP software computes the real estate results with search algorithm |  |
| Verif. Req. | SRS-REQ-003, | Analysis |
| Init. Cond. | REP Software is started and set in “Rental” mode |  |
| Tests inputs | Script search of criteria to generate data | See procedure search of criteria on how to use script |
| Data collection | Log file xxx-rental.log and post-processor script |  |
| Tests outputs | Output file rent-processor script and graph:rent-post-process.txt and rent-post-process.png | See procedure rental on how to use script |
| Assum & constr | Do not run data generator script with more than 1000 loops, or post process is too long |  |
| Expected results and criteria | xxx-rent-process.txt and xxx-rent-process.png | The post process graph is a gausian |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and eval crit** |
| 1 | Set log mode in “verbose” and restart software | Software is started, log file name xxx-verbose.log is created in xxx directory |
| 2 | Run xxx data generator script | Data binary file generated |
| 3 | Open data file with software | Software processes the file and a msg is displayed when finished |
| 4 | Run xxx post process and wait end of post processing | xxx-post-process.txt and xxx-post-process.png are generated |
| 5 | Open files | The post process graph is a gausian  The post process values are:  Chi2 = … Mean = … stdev= … |

Test

|  |  |  |
| --- | --- | --- |
| **Test ID** | **T-SRS-REQ-001** |  |
| Test desc. | Verify that the xxx software receives data from yyy |  |
| Verif. Req. | SRS-REQ-001, | Test |
| Init. Cond. | XXX Software is started  YYY simulator is started | See procedure yyy on how to use YYY simulator |
| Tests inputs | N/A |  |
| Data collection | N/A |  |
| Tests outputs | N/A |  |
| Assum & constr | YYY simulator work only in the range of zzz |  |
| Expected results and criteria | Data received and processed | See below |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and eval crit** |
| 1 | Send data with YYY simulator | Simulator displays msg “data sent” |
| 2 | Open xxx window of xxx software | Sent data are displayed, with date-time of reception |
| 3 | Press run button | Sent data are processed, result is displayed.  Value of result is … |

Test

|  |  |  |
| --- | --- | --- |
| **Test ID** | **T-SRS-REQ-004** |  |
| Test desc. | Verify that a user displays result in less than one minute |  |
| Verif. Req. | SRS-REQ-003, | Test |
| Init. Cond. | REP Software is started |  |
| Tests inputs | N/A |  |
| Data collection | Stopwatch |  |
| Tests outputs | N/A |  |
| Assum & constr | User should know how search mechanism work |  |
| Expected results and criteria | Results displayed in less than twenty second |  |
| **Test procedure** |  |  |
| **Step number** | **Operator actions** | **Expected result and eval crit** |
| 1 | For 3 users, run the main use scenario 3 times. | Give separate information for each user |
| 2 | Compute the mean of top searched houses | Give suggestions to user about specific informations |