Asia Pacific College

*School of Computing and Information Technology*

Quality – Test Plan

(Business Analytic Decision Support System for Realty Management)

Submitted to

**Mr. Joe Gene Quesada**

Instructor

In partial fulfilment of the requirements in

**Quality**

Submitted by

Cyberbullying Detection Using Support Vector Machine (SVM) algorithm Team

Faith Ballesteros

Paulo Miguel Capuz

Jason Juarez

Samantha Mallari

Eva Samillano

**Table of Contents**

[**1.** **Introduction** 3](#_Toc478926837)

[1.1 Purpose 3](#_Toc478926838)

[1.2 Scope 3](#_Toc478926839)

[1.3 References 3](#_Toc478926840)

[1.4 Document Terminology 3](#_Toc478926841)

[**2.** **Test Areas and Specifications** 4](#_Toc478926842)

[2.1 Features to be tested 4](#_Toc478926843)

[**3.** **Test Cycle Transition** 5](#_Toc478926844)

[3.1 Entry Criteria 5](#_Toc478926845)

[3.2 Exit Criteria 5](#_Toc478926846)

[3.3 Continuation Criteria 5](#_Toc478926847)

[3.4 Abnormal Termination 5](#_Toc478926848)

[**4.** **Project Milestones** 6](#_Toc478926849)

[**5.** **Test Deliverables and Execution** 7](#_Toc478926850)

[5.1 Test Deliverable 7](#_Toc478926851)

[5.2 Key Participants 7](#_Toc478926852)

[5.3 Bug Tracking 8](#_Toc478926853)

[5.4 Bug Isolation 8](#_Toc478926854)

[5.5 Test Reports 9](#_Toc478926855)

[5.6 Test Release Management 10](#_Toc478926856)

[**6.** **Environmental Needs** 11](#_Toc478926857)

[6.1 Hardware and Software 11](#_Toc478926858)

[6.2 Productivity and Support tools 11](#_Toc478926859)

[6.3 Test Environment Configuration 11](#_Toc478926860)

[**7.** **Planning Risks and Contingencies** 13](#_Toc478926861)

[**8.** **Approvals** 15](#_Toc478926862)

# **Introduction**

Business Analytic Decision Support System for Realty Management is an application that help the company Promax Realty Corporation (PRC) to automate paper-based system in their organization to avoid data loss, data redundancy, inconsistencies in their company workflow. The system will be able to manage the database record, calculate data and produce reports and graphs to produce reliable information to help decision maker to analyze data for future use.

## 1.1 Purpose

This Test Plan document supports the following objectives:

* Identify the project information about the system that should be tested
* To test the feasibility and performance of the system
* List the recommendation and describe the testing strategies to improving the system
* Show the result of the test activities
* To ensure the Functional and Design Requirements are well implemented and met the demand of the client
* To enhance the performance and interface of the system for user need purposes

## 1.2 Scope

* Test Plan is only for Business Analytic Decision Support System for Realty Management Group
* Testing are only focused on the Use Cases available in Test link (User Management, Contract Management and Property Management)

## 1.3 References

* Business Analytic (DSS) for Realty Management Documents (Software Requirement Specification, Statement of Work and Final Paper Submitted on Software Development)
* Knowledge of the Team Members to their project

## 1.4 Document Terminology

|  |  |
| --- | --- |
| PM | Project Manager |
| QC | Quality Control |
| TP | Test Plan |
| UC | Use Case |
| TC | Test Case/s |
| GUI | Graphical User Interface |
| DSS | Decision Support System |

# **Test Areas and Specifications**

## 2.1 Features to be tested

|  |  |  |  |
| --- | --- | --- | --- |
| **UC No.** | **UC Name** | **Test Type** | **Build No.** |
|  | | | |
| USER MANAGEMENT | | | |
| UC 01 | User Authentication | Functional, GUI, Performance | 001, 002 |
| UC 02 | Creating new user account | Functional, GUI, Performance | 001, 002 |
| UC 03 | Creating user types | Functional, GUI, Performance | 001, 002 |
| UC 04 | Update user account | Functional, GUI, Performance | 001, 002 |
| UC 05 | Change user password | Functional, GUI, Performance | 001, 002 |
| PROPERTY MANAGEMENT | | | |
| UC 06 | Creating unit types | Functional, GUI, Performance | 001, 002 |
| UC 07 | Change Access rights | Functional, GUI, Performance | 001, 002 |
| UC 08 | Edit company | Functional, GUI, Performance | 001, 002 |
| UC 09 | Update user type | Functional, GUI, Performance | 001, 002 |
| UC 10 | Create reservation contract | Functional, GUI, Performance | 001, 002 |
| CU 11 | Create units | Functional, GUI, Performance | 001, 002 |
| UC 12 | Create tenant | Functional, GUI, Performance | 001, 002 |
| UC 13 | Create Property | Functional, GUI, Performance | 001, 002 |
| CONTRACT MANAGEMENT | | | |
| UC 14 | Create reservation contract | Functional, GUI, Performance | 001, 002 |
| UC 15 | Generate invoice | Functional, GUI, Performance | 001, 002 |
| UC 16 | Create invoice for Miscellaneous payment | Functional, GUI, Performance | 001, 002 |
| UC 17 | Apply payment for Miscellaneous payment | Functional, GUI, Performance | 001, 002 |

# **Test Cycle Transition**

## 3.1 Entry Criteria

* Requirements are documented, reviewed, and approved (Iteration TP and TC
* Build release is complete and ready and
* The build is ready for testing

## 3.2 Exit Criteria

* TC execution completed
* TC completed with the same percentage of failed result
* All defects are not fixed
* Same bugs found after regression testing

## 3.3 Continuation Criteria

* PM request to execute test
* Bugs are not fixed

## 3.4 Abnormal Termination

* Blocker and/or critical bug found that would cause the termination of executing the test
* Incomplete deliverables that would waste time to execute test wrong configured   
   environment
* PM request to terminate the test
* No changes has been made into the system

# **Project Milestones**

|  |  |  |
| --- | --- | --- |
| **Milestone Task** | **Date Start** | **Date End** |
| System Test Planning  (Use Cases for our System) | February 27, 2017 | March 01, 2017 |
| System Test Development – Iteration 1  (Test Cases for other group System) | March 06, 2017 | March 09, 2017 |
| System Test Execution – Iteration 1 | March 10, 2017 | March 31, 2017 |
| System Test Evaluation – Iteration 1 | March 31, 2017 | March 31, 2017 |
| System Test Development – Iteration 2  (Test Cases for other group System) | March 31, 2017 | April 01, 2017 |
| System Test Execution – Iteration 2 | April 01, 2017 | April 01, 2017 |
| System Test Evaluation – Iteration 2 | April 01, 2017 | April 01, 2017 |
| Review of the Test Cases | April 01, 2017 | April 02, 2017 |
| Create Test Plan | February 27, 2017 | April 02, 2017 |

# **Test Deliverables and Execution**

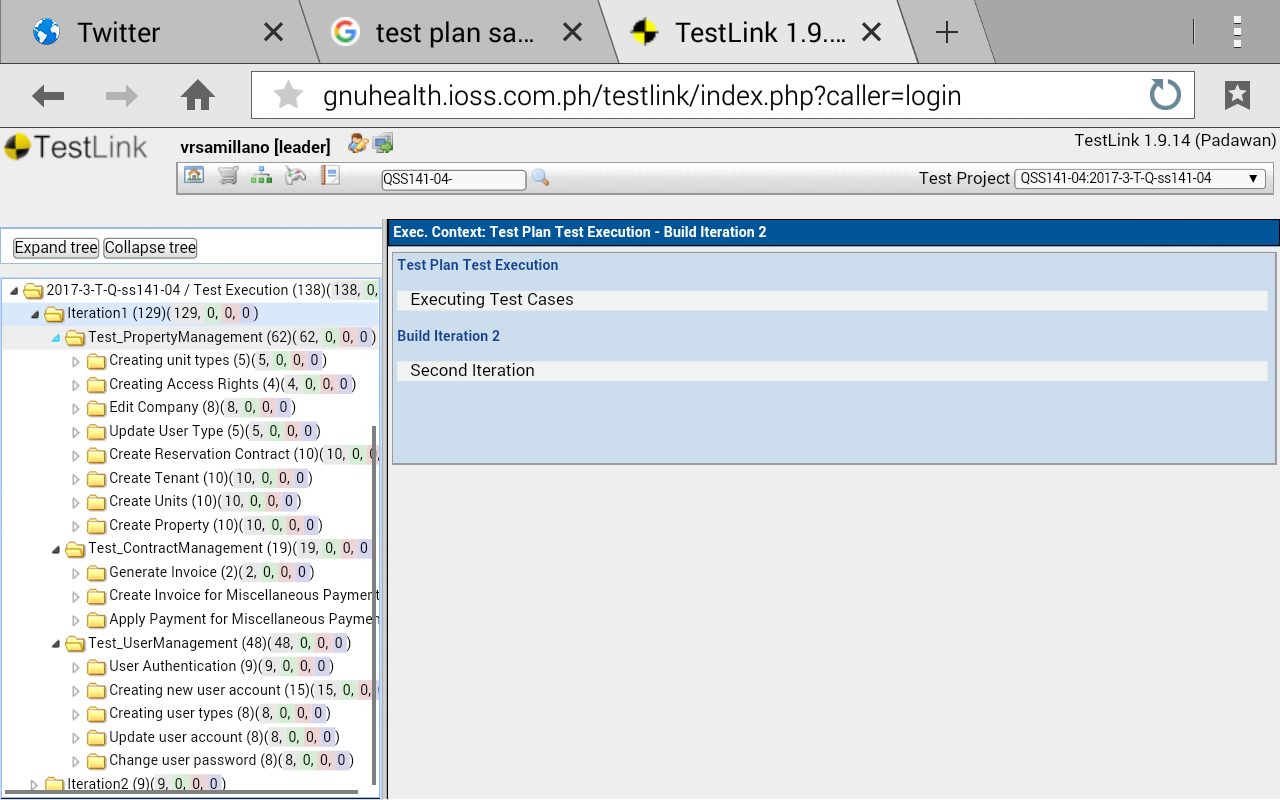
## 5.1 Test Deliverable

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Date Start** | **Date End** |
| Create Use Cases  (will be tested by other group) | February 27, 2017 | March 01, 2017 |
| Develop Test Plan | February 27, 2017 | April 02, 2017 |
| Update and Review Test Plan | March 03, 2017 | April 02, 2017 |
| **BUILD 001** | | |
| Test Cases(Iteration1) | March 06, 2017 | March 09, 2017 |
| Execute TC – Build 001 | March 10, 2017 | March 31, 2017 |
| Test Log/Reports | March 31, 2017 | March 31, 2017 |
| **BUILD 002** | | |
| Test Cases(Iteration2) |  |  |
| Execute TC – Build 002 | March 31, 2017 | April 01, 2017 |
| Test Log/Reports | April 01, 2017 | April 01, 2017 |
| Review/Update all the Test Cases | April 01, 2017 | April 02, 2017 |

## 5.2 Key Participants

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Note** |
| Samantha Mallari | Project Manager | Property Management 58-84 test cases |
| Faith Ballesteros | Technical Writer/UI/UX Designer | User Management and Property Management 36-57 test cases |
| Eva Samillano | Software Analyst/Programmer | User Management 1-35 test cases |
| Paulo Capuz | Programmer | Contract Management 115-129 test cases |
| Jason Juarez | Software Quality Assurance Engineer/Software Tester | Property Management 85-114 test cases |

## 5.3 Bug Tracking



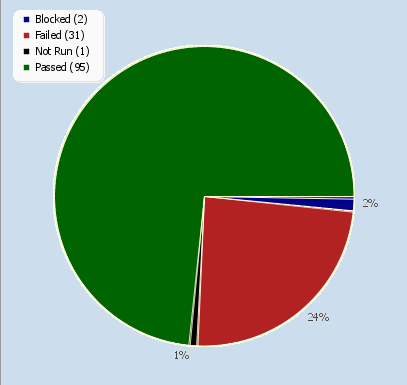
## 5.4 Bug Isolation

|  |  |
| --- | --- |
| **Status** | **Resolution** |
| **Open**  The issue is open and ready for the assignees to start work on it. | **Fixed(Default)**  A fix for the issue is checked and tested |
| **In progress**    This issue is being actively worked on at the moment by the assignees. | **Won’t Fix**  The problem described is an issue which will never be fixed. |
| **Reopened**    This issue is already address but no action has been taken to correct the issue | **Duplicate**  The problem is a duplicate of an existing issue |
| **Resolved**  A resolution has been taken, and it is awaiting verification by the programmer. From here issues are either reopened, or are closed. | **Incomplete**  The problem is the information is not completely indicated |
| **Closed**  The resolution for the issue is re-schedule for the continuation of the project. | **Cannot Reproduce**    Failure cannot resolve because lack of information to produce a solution. |
|  |  |

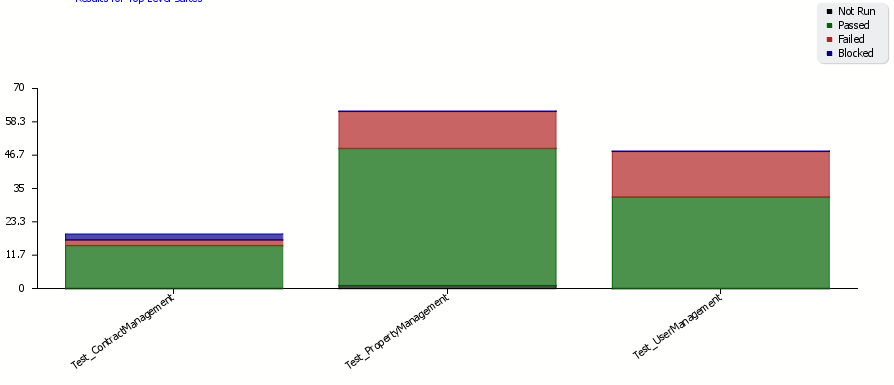
|  |  |
| --- | --- |
| **Priority Levels** | **Impact of Bug** |
| **Blocker** | Impact of a bug |
| **Critical** | Blocks development and/or testing work, production could not run. |
| **Major** | Crashes, loss of data, severe memory leak |
| **Minor** | Major loss of function |
| **Trivial** | Minor loss of function, or other problem where easy workaround is present. |

## 5.5 Test Reports

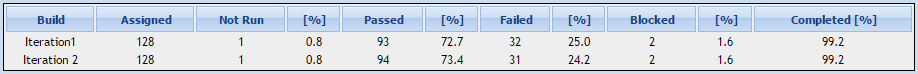
Evaluated test result after the test case execution in every iteration will be shown below.



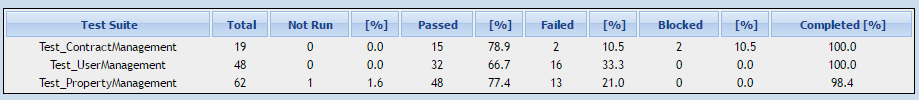
*Figure1: Pie Chart for Iteration1 & Iteration2 consisting of 129 Test Cases*

**

*Figure2: Test Result per Use Cases*

**

*Figure3: Overall Build Status per Iteration*

**

*Figure4: Test Result by top level Test Suites*

This result will be present to the Project Manager and Software Quality Assurance Engineer of the other group.

Note: Other Test Report are organized through excel in GitHub

(Test Result Flat and Test Result Matrix) <https://github.com/vrsamillano/Quality_SystemCode>

## 5.6 Test Release Management

 Test release is handled by Project Manager, Software Analyst, Programmer and Software Quality Assurance Engineer where ever build a sanity test will be conducted to prevent further more error, and bugs that might be occur.

# **Environmental Needs**

## 6.1 Hardware and Software

|  |  |  |
| --- | --- | --- |
| **Resource** | **Configuration** | **Installed OS, Software** |
| **Test Server** |  | Windows 10 – Operating System (64 bit) |
|  |  | Laravel Framework |
|  | Apache Web Server,  MySQL Database | XAMPP |
| **Client Server** | (Standard)  +RAM: 2GB | Windows7 (64 bit)  Chrome and Mozilla Firefox |

## 6.2 Productivity and Support tools

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool’s Purpose** | **Tool Name** | **Vendor or in-house** | **Version** |
| **Bug Tracking**  **Management** | TestLink | Open Source | 1.9.2 |
| **Test Management** | TestLink | Open Source | 1.9.2 |
| **Documentation** | MS Word, MS excel | Microsoft Office | 2016 |
| **Revision Control System** | Planner   (Office 365),  GitHub | Microsoft    Tom-Preston Werner | GitHub Enterprise2.9 |

## 6.3 Test Environment Configuration

|  |  |  |
| --- | --- | --- |
| **Configuration Name** | **Description** | **Implemented in Physical Configuration** |
| **Test Server** | Install test server at system admin test server | Open any device where Application are installed |
| **Testlink** | Open Source Software Application | Log-in |
| **GitHub** | Open Source Software Application | Log-in |

# **Planning Risks and Contingencies**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risks** | **Mitigation** | **Contingency**  **(risks is realized)** | **Probability** | **Impact** |
| **SCHEDULE**  Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the UAT schedule start date. | * Testing team should control the preparation tasks (in advance) * Make an early communication to the people involved in testing. | * Optimize the test schedule and define overtime strategy at the beginning | High | High |
| **DEFECTS**  Defects are found at a late stage of the cycle, those are changes that discovered late due to unclear specifications. | * Management plan is in place to ensure prompt communication and fixing of issues. | * Document defects found * Report found defects to the programmer * Apply all changes made by the programmer * Document possible result | Medium | High |
| **SCOPE**  Scope are not completely defined | * Scope in their paper is well defined but the changes in the functionality are not yet done so might there will be added or deleted process. | * Revised document after correction has been made * Execute TC after correcting and defining the document | Medium | Medium |
| **Lack of resources for testing**  Non-availability of Independent Test environment and accessibility | * Prepare other computer for testing environment * Send application Codes into GitHub for testing * Install software into tester computer/laptop for testing | * Testing time is based on the availability of the computer/Laptop * Implementing Static Testing/Verbal Testing | Medium | High |

# **Approvals**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Signature** | **Date** |
| Samantha Mallari | Project Manager |  |  |
| Faith Ballesteros | Technical Writer/UI/UX Designer |  |  |
| Eva Samillano | Software Analyst/Programmer |  |  |
| Paulo Capuz | Programmer |  |  |
| Jason Juarez | Software Quality Assurance Engineer/Software Tester |  |  |