

# Test Plan

# Twitter User Analyser

---

**Submitted By:**

**IIT2016106 - Aswin VB**

**IIT2016105 - Aswanth K**

**IIT2016133 - Utpal Aman**

**BIM2016501 - Druval CR**

**IRM2016006 - Sarath Nandimandalam**



## Table Of Contents

1. Objective
2. Scope
3. References
4. Functions to be tested
5. Testing Process Overview
6. Test Strategy
7. Testing Types
  - Black box Testing
  - GUI Testing
  - Integration Testing
  - Functional Testing
  - System Testing
  - Performance Testing
8. Test Schedule
9. Control Procedure
10. Entry Criteria
11. Suspension Criteria
12. Exit Criteria

## 1. Introduction

The aim of this document is to list the various test scenarios that have been taken care of while our application was under development. Our application - Twitter User Analyser, has two parts Analysing a particular user and Classifying different Twitter users based on Classification algorithms, we have included the test plan for both the phases in this document.

## 2. Objective

The objective of the Test plan is to define the various Testing strategies and testing tools used for the complete Testing life cycle of this project.

**The testing life cycle of our application includes 4 main strategies :**

- **Unit testing:** Unit testing starts at the center and each unit is implemented in the source code.
- **Integration testing:** An integration testing focuses on the construction and design of the software.
- **Validation testing:** Check all the requirements like functional, behavioral and performance requirements are validated against the construction software.
- **System testing:** System testing confirms all system elements and performance are tested entirely.

## 3. Scope

The document mainly targets the GUI testing and validating data in report output as per Requirements Specifications provided by Client. The SRS that was prepared initially is checked during the testing phase to make sure if the developed software meets the needs of the client.


The main functions to be tested are usually mentioned as a functiona

## Functions to be used

Our application has 2 functions to be tested

### Analysis of a twitter user

To test this feature the sample twitter username needs to be entered and the algorithm which



has been implemented will find out in which category the user belongs to ( anomalous, suspected or non-anomalous ).

### Classification

A dataset of twitter usernames are passed on to the function which finds out the different parameter values ( a,b,c,d,e, FAL, type ) which is stored in a CSV file and once all the usernames are computed by the algorithm, by using different classification algorithms like KNN, Naive Bayes, Decision Tree, Random Forest, and SVM classifiers the data is classified and the accuracy of each classifier is compared.

## 4. References

References include the specifications mentioned in the SRS prepared by the analyst, the SRS prepared consists of the client's requirements and also includes the functional and the non-functional requirements for the application.

## 5. Testing Process Overview

### Test Process

Test process followed by QA will be categorized into 2 ways:

- Process to be followed when sufficient time is available for QA
- Process to be followed when sufficient time is not available for QA

### Understanding Requirements:

- Requirement specifications are prepared considering all the scenarios.
- Understanding of requirements is done.
- Raised queries will be re-verified.
- Response to queries are taken and the requirements are altered accordingly.

### Preparing Test Cases:

#### - Invalid usernames :

When invalid usernames are entered, the algorithm will not be able to fetch the tweets which in turn will tell us it is an invalid user and with the help of exception handling the program will continue without terminating.

#### **- Invalid URLs while checking WOT and URL ranks:**

If the URLs which are present in a tweet are invalid, to find the WOT and URL ranks GET request needs to be sent which in turn will raise an exception if the URL is invalid

#### **- Less number of data in the dataset to be classified**

Since the classification of data is meaningful only if there is ample amount of data to be classified, so if there is less number of data in the classification dataset the classification algorithm will raise an error. To avoid this error, before the data is passed into the classifier the size of the dataset is checked and if there is ample amount of data then it is passed else a message will be displayed saying less number of data.

#### **Reviewing test cases:**

Once all the test cases are listed, it is time to test the application and check its coverage accordingly the suggestions or improvements will be re-worked and re-checked using these test cases.

#### **Creating Test Data:**

Test data was created based on the scope of the current version of the application, and is extended further on when newer updated version of the application is developed.

#### **Executing Test Cases:**

The test cases are executed keeping an idea what the desired outcome needs to be and accordingly, develop the application catering to all the needs and covering all the cases that the application can run into.


## **6. Test Strategy**

### **Testing types**

#### **Black box testing:**

It is sometimes called behavioral testing or Partition testing. This kind of testing focuses on the functional requirements of the software. It enables one to derive sets of input conditions that will fully exercise all functional requirements for a program.

#### **GUI Testing:**



GUI testing will include testing the UI part of the report. It covers users Report format, look and feel, error messages, spelling mistakes, GUI guideline violations.

**Integration Testing:**

Integration testing is a systematic technique for constructing the program structure while conducting the test to uncover errors associated with interacting. In Report, integration testing includes the testing Report from the respective location(s).

**Functional Testing:**

Functional testing is carried out in order to find out the unexpected behavior of the report. The characteristic of functional testings are to provide correctness, reliability, testability and accuracy of the report output/data.

**System Testing:**

System testing of software is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

**Performance Testing:**

Performance testing will be done by Client

• **User acceptance testing:**

The purpose behind user acceptance testing is to confirm that the system is developed according to the specified user requirements and is ready for operational use.

Acceptance testing is carried out at two levels..

## 7. Test Schedule

**Planning Phase:**

High-level test planning activities, which include preliminary development of Master QA Plan (this document, QA schedule. At this Milestone, the high level planning should be completed. Some of the deliverables are: Project Plan, Program function specifications.

**Design Phase:**

Reviewing the requirements and design documents. As the design documents are completed, it is encouraged to start working on the Test Plan document and test



design planning

The scenarios, expected results, data sets, test procedures, scripts are generated.

**Code Complete-Function:**

White Box Test Plan, test cases and other QA documents related to test execution for each feature or component such as test scenarios, expected results, data sets, test procedures, scripts and applicable testing tools.

**Feature Complete:**

All bugs verified and QA documentation is finalized. One should assess that Binary Tree features are ready for Beta regression and have started their preliminary Test Summary Reports.

**Regression Test:**

Complete regression test execution of complete system and update Test Summary Reports for regression.

## 8. Entry Criteria

- The whole source code must be unit tested H/W and S/W should be in place
- QA resources have completely understood the requirements
- QA resources have sound knowledge of functionality in Reports

## 9. Suspension Criteria

- If build contains many serious defects which seriously or limit testing progress.
- Significant change in requirements suggested.
- Software/Hardware problems
- Assigned resources are not available when needed by test team.



## 10. Resumption Criteria

- Resumption will only occur when the problem(s) that caused the suspension have been resolved

## 11. Exit Criteria

- No defects over a period of time or less testing efforts
- All the high priority/severity test cases has been executed
- Deliverables are ready