SE 3XA3: Test Plan
Title of Project

Team #31, R-DB V2
Jason Tsui tsuij8
Student 2 name and macid
Student 3 name and macid

October 25, 2018

Contents

List of Tables

List of Figures

Table 1: **Revision History**

Date	**Version**	**Notes**
Oct 25, 2018	1.0	Section 1
Date 2	1.1	Notes

1 General Information

1.1 Purpose

This document is a Test Plan document. This document will discuss and describe the testing, validation, and verification procedures to implement a discord chat bot, R-DB V2. As of document creation, the project is still undergoing major development and the test case procedures and descriptions disscussed are subject to change.

1.2 Scope

This project is a python based server hosted chat bot, which is able to take user input and output into the discord community server. The scope of testing for this program will cover user input, Discord API integration, and expected outputs.

1.3 Acronyms, Abbreviations, and Symbols

Table 2: **Table of Abbreviations**

Abbreviation	**Definition**
API	Aplication Programming Interface
VoIP	Voice over Internet Protocol

1.4 Overview of Document

R-DB V2 will be following the logical structure for testing purposes. Unit testing will primarily focus on handling user input and program logic. Integration testing will be devoted to integrating the bot onto the Discord API platform. System testing will be be done by observing if chat bot commands are correct. Acceptance testing will be done by a 3rd party user to evaluate non-functional requirements.

Table 3: **Table of Definitions**

Term	**Definition**
Discord	Freeware VoIP application. VoIP service which connects people and creates voice chat communities
VoIP	Methodology for communicating over the internet
R-DB V2	Name of project
Black Box Testing	Functional testing by probing program with inputs. Testing is concered with output of program
White Box Testing	Structural testing by understanding how program processing occurs. Testing is concered with soundess of program
Pytest	Python testing framework
Red Discord Bot	Original program which project is based off of

2 Plan

2.1 Software Description

2.2 Test Team

2.3 Automated Testing Approach

2.4 Testing Tools

2.5 Testing Schedule

See Gantt Chart at the following url ...

3 System Test Description

3.1 Tests for Functional Requirements

3.1.1 Area of Testing1

Title for Test

1. test-id1

Type: Functional, Dynamic, Manual, Static etc.

Initial State:

Input:

Output:

How test will be performed:

2. test-id2

   Type: Functional, Dynamic, Manual, Static etc.

   Initial State:

   Input:

   Output:

   How test will be performed:

3.1.2 Area of Testing2

...

3.2 Tests for Nonfunctional Requirements

3.2.1 Area of Testing1

Title for Test

1. test-id1

   Type:

   Initial State:

   Input/Condition:

   Output/Result:

   How test will be performed:

2. test-id2

   Type: Functional, Dynamic, Manual, Static etc.

Initial State:

Input:

Output:

How test will be performed:

3.2.2 Area of Testing2

...

3.3 Traceability Between Test Cases and Requirements

4 Tests for Proof of Concept

4.1 Area of Testing1

Title for Test

1. test-id1

   Type: Functional, Dynamic, Manual, Static etc.

   Initial State:

   Input:

   Output:

   How test will be performed:

2. test-id2

   Type: Functional, Dynamic, Manual, Static etc.

   Initial State:

   Input:

   Output:

   How test will be performed:

4.2 Area of Testing2

...

5 Comparison to Existing Implementation

6 Unit Testing Plan

6.1 Unit testing of internal functions

6.2 Unit testing of output files

7 Appendix

This is where you can place additional information.

7.1 Symbolic Parameters

The definition of the test cases will call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance.

7.2 Usability Survey Questions?

This is a section that would be appropriate for some teams.