**Test Plan – The Godfather Bot**

Outsourced*™* - Tod Jones, Dakota Methvin, Vince Seely

1. **Introduction**

*State the purpose of the plan and identify the scope of the plan in relation to your project plan. You may also include the resource and budget constraints, and scope of the testing effort.*

The intention of this plan is to provide a descriptive framework for the verification and validation processes we are using to correctly perform maintenance on our project.

Given the large (~50k lines) codebase, our scope could be anywhere from modified code unit tests to system-scale testing. For this project, we have each taken a different approach, so our scales are slightly different.

Vince has chosen to work on a wide-spread system change which will make the scope of his required testing larger than that of the other two developers. Dakota is adding a new functionality module so his scope is that module plus the hooks from the main project. Tod is refactoring some of the main code, so his scope is the linking of that class to the dependent classes.

Resources include the Visual Studio testing suite. Because our project is entirely .NET C#, we are able to test in-IDE without requiring an external testing suite. If we were doing more intensive maintenance, we might require more rigorous or capable tools. Such maintenance would be outside the scope of the project for this class, however.

Our time budget is two weeks, but split by our other responsibilities. We have met twice in this time and have worked from home another.

1. **Test Items**

*List the items you intend to test within the scope of this test plan. Essentially, something you will test, a list of what is to be tested.*

Since we each took a different approach to maintaining this project, we have different test items between developers. Enumerated below are our individual high-level test items:

**Dakota**

* R6Stats module (new code)
* DSharpPlus module integration (w.r.t. R6Stats module)

**Vince**

* TODO

**Tod**

* TODO

1. **Features to be tested**

*This is a listing of what is to be tested from the USERS viewpoint of what the system does.*

**Dakota**

* R6Stats command: get my statistics
* R6Stats command: get other player’s statistics
* Other R6Stats commands as implemented

**Vince**

* TODO

**Tod**

* TODO

1. **Approach**

*This is your overall test strategy for this test plan. Overall rules and processes should be identified.*

* *Are any special tools to be used and what are they?*
* *What levels of regression testing will be done and how much at each test level?*
* *Will regression testing be based on severity of defects detected?*

**Dakota**

* No external dependencies for testing
* Visual Studio testing suite for unit tests
* No regression testing possible (command pattern interface for new modules)

**Vince**

**Tod**

1. **Test Cases**
   1. **Unit Testing**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case No. | Test Case Description | Input | Expected Output/Result |
| R6\_U\_001 | Connect to Ubisoft API with known U:P | User:Password | UserID returned from API |
| R6\_U\_002 | Retrieve player statistics given existing API connection | API connection, username, statistics type | Player statistics by type requested |
| R6\_U\_003 | Display player statistics using DSharpPlus hook to Discord embed object | Player statistics, target channel | Embed object is displayed in target channel |

* 1. **Integration Testing**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case No. | Test Case Description | Classes/Methods Invoking Sequence | Expected Output/Result |
| R6\_I\_001 | DSharpPlus command pattern | DSharpPlus library via user input to channel | R6Stats command called via DSharpPlus |
|  |  |  |  |