Test Plan

Bowlstrike.py

# The Product

The project is developing a 10 pin bowling game prototype that can be used to teach a variety of subjects, it is made by a company specialised in educational software.

**Rules of Play**

Each game of bowling consists of ten frames. In each frame, the bowler will have two chances to knock down as many pins as possible with their bowling ball. In games with more than one bowler, as is common, every bowler will take their frame in a predetermined order before the next frame begins. If a bowler knocks down all ten pins with their first ball, he is awarded a strike. If the bowler knocks down all 10 pins with the two balls of a frame, it is known as a spare. Bonus points are awarded for both a strike and a spare. The bonus points awarded depend on what is scored in the next 2 balls (for a strike) or 1 ball (for a spare). If the bowler knocks down all 10 pins in the tenth frame, the bowler is allowed to throw 3 balls for that frame. This allows for a potential of 12 strikes in a single game, and a maximum score of 300 points, a perfect game.

# Test Strategy

## Testing Scope

In Scope:

* Code formatting
* Functions
* Prototype

Out of Scope:

* Graphic UI
* Database
* Input data

## Testing Type

Which Testing Types should be focused for testing?

* Back-end Testing
* Component Testing
* Functional Testing

Which Testing Types should be ignored?

* Graphical User Interface Testing
* Compatibility Testing
* Performance Testing

## Test Logistics

Testing and Bug fixing are going to be performed by Vitor Cazella, during the period of a week.

# Testing Objective

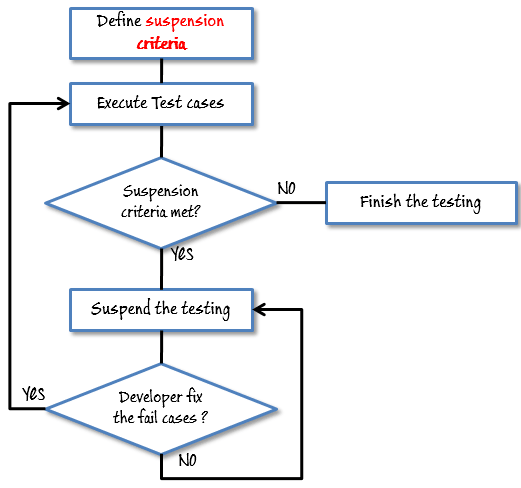
List all the software features (functionality, performance, GUI…) which may need to test.

Define the target or the goal of the test based on above features

# Test Criteria

Suspension Criteria

Test Plan Example: If your team members report that there are 40% of test cases failed, you should suspend testing until the development team fixes all the failed cases.



Exit Criteria

Some methods of defining exit criteria are by specifying a targeted run rate and pass rate.

# Resource Planning

Human resource

|  |  |  |
| --- | --- | --- |
| No. | Member | Tasks |
| 1. | Test Manager | Manage the whole project  Define project directions  Acquire appropriate resources |

System Resource

|  |  |  |
| --- | --- | --- |
| No. | Resources | Descriptions |
| 1. | Server | Install the web application under test  This includes a separate web server, database server, and application server if applicable |
| 2. | Test tool | The testing tool is to automate the testing, simulate the user operation, generate the test results  There are tons of test tools you can use for this project such as Selenium, QTP…etc. |
| 3. | Network | You need a Network include LAN and Internet to simulate the real business and user environment |
| 4. | Computer | The PC which users often use to connect the web server |

# Test Environment

A testing environment is a setup of software and hardware on which the testing team is going to execute test cases.

# Schedule & Estimation

|  |  |  |
| --- | --- | --- |
| Task | Members | Estimate effort |
| Create the test specification | Test Designer | 170 man-hour |
| Perform Test Execution | Tester, Test Administrator | 80 man-hour |
| Test Report | Tester | 10 man-hour |
| Test Delivery |  | 20 man-hour |
| Total |  | 280 man-hour |

### Test Deliverables

Test deliverables are provided **before** testing phase.

* Test plans document.
* Test cases documents
* Test Design specifications.

Test deliverables are provided **during** the testing

* Test Scripts
* Simulators.
* Test Data
* Test Traceability Matrix
* Error logs and execution logs.

Test deliverables are provided **after** the testing cycles is over.

* **Test Results/reports**
* Defect Report
* Installation/ Test procedures guidelines
* **Release notes**