Vijendra Singh

20220756@mywhitecliffe.com

Test Plan

Bowling Game

**Index**

|  |  |
| --- | --- |
| Introduction | 3 |
| Objectives | 4 |
| Approach | 4 |
| Roles and Responsibilities | 5 |
| Test Deliverables | 5 |
| Acceptance Criteria | 5 |
| Approval | 6 |
| Entry and Exit Criteria | 6 |
| Test Schedule | 6 |
| Test Strategy | 7 |
| Test Summary Report | 10 |

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Description** | **Version** | **Created By** | **Approved By** |
| 27th June 2023 | Test Plan Created | 0.1 | Vijendra Singh | Vivian Wang |
|  |  |  |  |  |

1. **Introduction:**
   1. **Purpose**:

The SRS (Software Requirement Specification) provides the requirement for “Bowling Game”. The purpose of this game is to evaluate the result of each player playing the game according to pins scored by them.

Total number of frames for each player is 10 and 2 balls are provided in each Frame.

If player knocked 10 pins in first ball, then it will b consider as Strike and player will be awarded with one Extra strike and bonus point.

If player knocked 10 pins in two balls in one single frame, then it will be consider as Spare and bonus point will be awarded.

Bonus points are calculated as total score of next two balls.

* 1. **Scope**:

This test mainly targets the functionality of code and validating the output as expected by the developer.

This test is limited to White box testing by the tester to perform testing on developer code and its functionality.

**Function to be tested:**

* Test Gutter Game
  + Roll 0 in all two balls in 10 frames.
  + Verify the score is 0.
* Test All One
  + Roll 1 in all two balls in 10 frames.
  + Verify the score is 20.
* Test One Spare
  + Roll 5 in all two balls in 1 frame.
  + Roll 3 in first ball in 2nd frame.
  + Roll 0 in all remaining balls in remaining frames.
  + Verify the score is 16.
* Test One Strike
  + Roll 10 in all first balls in 1 frame.
  + Roll 4 and 3 in two ball in 2nd frame.
  + Roll 0 in all 16 balls in remaining 8 frames.
  + Verify the score is 24.
* Test Perfect Game
  + Roll 10 in all first balls in all 10 frames.
  + Roll 10 in extra 2 ball in extra one frame.
  + Verify the score is 300.
* Test All Spare
  + Roll 5 in first and 2nd ball in all 10 frames.
  + Verify the score is 150.
* Test Roll pins for invalid range i.e. -1 and 11
  + Roll -1 for 10 rolls.
  + Roll 11 for next 10 rolls.
* Test Roll for invalid Character
  + Roll ‘a’ for 5 rolls.
  + Roll 2.0 for 5 rolls
  + Roll 4.6 for next 10 rolls.

1. **Objectives**:

Primary objective of this testing is to validate all functionality is working perfectly in Bowling Game mentioned in SRS document.

That will include all functional requirement as follows:

* Player will be awarded with one extra strike and bonus point for next two balls when hit Strike.
* Player will be awarded with bonus point for next two balls when hit Spare.
* In Case of perfect game player gets 300 points with one extra Strike.
* Player will get points accordingly if get one Strike and normal strike.
* Player will get points accordingly if get one Spare and normal strike.
* Player gets normal points when score normally i.e., neither score Spare nor Strike.
* For invalid range of pins player get 0 like in case of rolling -1 or 11 in rolls.
* For invalid character like str player get 0 and float will be converted to int.

1. **Approach** 
   1. **Test** **Environment**:

Stakeholders involved in this testing are as follows:

* Testing team
* Developer Team
  1. **Test Resources:**

Current version of application will be tested in below mentioned Environment:

Desktop Configuration :

* Processor: Intel Processor over and above core i3
* Ram: 8Gb or more
* Operating System: Window 8 or above

application needs to be tested:

* IDE: Visual Studio Code version 1.53.0.2 or above
* Python: version 3.0 or above

3.3 **Testing Tool:**

For test case creation, Tracking, management use Microsoft Excel and submit all report in pdf/Excel or word format.

1. **Roles and Responsibilities**

|  |  |
| --- | --- |
| **Project Manager** | Will be Responsible for providing overall guidance and support for this Test.  Will be the Primary contact for development and Quality Assurance. |
| **Business Analyst** | Responsible for gathering CRS (Customer Requirement Specifications) from the client and convert them into SRS (Software Requirement Specifications) for development and testing. |
| **Test Lead:** | Responsible for managing the test process, test planning, coordination, and reporting. |
| **Tester** | Responsible for test case design, execution, defect tracking, and test documentation. |
| **Developer** | Responsible for resolving defects and error identified during this test. |

1. **Test Deliverables**

The following documents will be provided during the test:

* 1. Test Plan
  2. Test Scenarios
  3. Test Cases
  4. Defect Reports
  5. Test Summary Report

1. **Acceptance Criteria**

Approval for the test will be based on following process:

* + Successful completion of all identified test scenarios and test cases.
  + Verifying all requirement as per client requirement with SRS.
  + After completion above process Project Manager Approval will be required

1. **Approval:**

|  |  |  |
| --- | --- | --- |
|  | **Project Manager** | **Test Lead** |
| **Name** |  |  |
| **Date** |  |  |
| **Signature** |  |  |

1. **Entry and Exit Criteria**

Entry Criteria:

* All test hardware, tool and software installed as per the testing environment.
* All documentation mentioned in Test Deliverables is available for testing.
* User has understood all testing requirement.
* User has knowledge of code functionality.

Exit Criteria:

* All test case scenario successfully executed.
* Defects and error resolved by developer.
* No high priority or server bugs are left.
* All test performed as per the schedule.

1. **Test Schedule:**

**Date: 27th June 2023**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Name** | **Start Time** | **Finish Time** | **Comments** |
| Test Planning | 10:00 | 12:00 |  |
| Functional Testing 1 | 13:00 | 13:30 |  |
| Functional Testing 2 | 13:30 | 14:00 |  |
| Functional Testing 3 | 14:00 | 14:30 |  |
| Functional Testing 4 | 14:30 | 15:00 |  |
| Functional Testing 5 | 15:00 | 15:30 |  |
| Functional Testing 6 | 15:30 | 16:00 |  |
|  |  |  |  |

1. **Test Strategy:**

Users’ role in test process:

User has understood all requirement specified in SRS by client.

Preparing test Case:

Tester will prepare test case based on the following functional

Functional Functionality:

* Test Gutter Game
* Test All One
* Test One Spare
* Test One Strike
* Test Perfect Game
* Test All Spare
* Test invalid range of pins
* Test invalid character in rolls

Test Methodology:

This test is involved manual testing by the user to test functional functionality.

* White box testing:

White box testing methodology is applied in this test plan to test the code and its functionality provided by developer.

Test case is designed to test the code’s logic and data flows.

Test Case and test Data:

* Test Gutter Game
  + Roll 0 in all two balls in 10 frames.
  + Verify the score is 0.
* Test All One
  + Roll 1 in all two balls in 10 frames.
  + Verify the score is 20.
* Test One Spare
  + Roll 5 in all two balls in 1 frame.
  + Roll 3 in first ball in 2nd frame.
  + Roll 0 in all remaining balls in remaining frames.
  + Verify the score is 16.
* Test One Strike
  + Roll 10 in all first balls in 1 frame.
  + Roll 4 and 3 in two ball in 2nd frame.
  + Roll 0 in all 16 balls in remaining 8 frames.
  + Verify the score is 24.
* Test Perfect Game
  + Roll 10 in all first balls in all 10 frames.
  + Roll 10 in extra 2 ball in extra one frame.
  + Verify the score is 300.
* Test All Spare
  + Roll 5 in first and 2nd ball in all 10 frames.
  + Verify the score is 150.
* Test Roll pins for invalid range i.e. -1 and 11
  + Roll -1 for 10 rolls.
  + Roll 11 for next 10 rolls.
* Test Roll for invalid Character
  + Roll ‘a’ for 5 rolls.
  + Roll 2.0 for 5 rolls
  + Roll 4.6 for next 10 rolls.

Testing Type:

White Box Testing (Functional Testing)

Bug Severity and Priority Level:

|  |  |
| --- | --- |
| Severity: | Critical  High  Medium  Minor |
| Priority: | P1  P2  P3  P4  P5 |

Creating Test Data:

Test data will be created as specified.

Executing Test Case:

Users/Tester will execute test as per test schedule, on based of test scenario mentioned in test plan.

Actual Test result (fail/pass) will be updated on test case document and defect and error will be logged and reported.

Team lead will update all bugs to developer for resolution.

Retesting and regression Testing:

Retesting for fixed bugs will be done by tester and regression testing of all features will be executed again as per the process.

Delivery:

Once all bugs are fixed, report will be sent to client for final decision.

**TEST SUMMARY REPORT**

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Description** | **Version** | **Created By** | **Approved By** |
| 27th June 2023 | Test Summary Report | 0.1 | Vijendra Singh | Vivian Wang |
|  |  |  |  |  |

**GitHub Link:** [**https://github.com/vijend795/FinalAssignment\_20220756.git**](https://github.com/vijend795/FinalAssignment_20220756.git)

**Summary:**

Test case for all below mentioned functionality was successfully executed on 27th June 2023.

1. Test Gutter Game

Code has TypeError for roll functions, rolls is a list array to store all pins and can’t be called as method. instead of roll is the correct method to called.

Error raised and rectified by developer.

Recursion test done and found AttributeError for wrong method name stickeScroe, correct name of method is ‘strikeScore’

Error raised and rectified by developer.

Recursion test done and found AssertionError: result not matched and asked to change the method of assertion to get the actual result value.

Error raised and rectified by developer.

Recursion test done and found AssertionError: actual result is 10 and expected result is 0

Error raised and Code refactored.

Recursion test done and asserted result is 0 as expected

1. Test All One

Code has AssertionError: result not matched and asked to change the method of assertion to get the actual result value.

Error raised and rectified by developer.

Recursion test done and found AssertionError: actual result is 12 and expected result is 20

Error raised and Code refactored.

Recursion test done and asserted result is 20 as expected

1. Test One Spare

Code has TypeError for roll functions, rolls is a list array to store all pins and can’t be called as method. instead of roll is the correct method to called.

Error raised and rectified by developer.

Recursion test done and found AssertionError: result not matched and asked to change the method of assertion to get the actual result value.

Error raised and rectified by developer.

Recursion test done and found AssertionError: actual result is 18 and expected result is 16

Error raised and Code refactored.

Recursion test done and asserted result is 16 as expected

1. Test One Strike

Code has TypeError for roll functions, rolls is a list array to store all pins and can’t be called as method. instead of roll is the correct method to called.

Error raised and rectified by developer.

Recursion test done and found AssertionError: result not matched and asked to change the method of assertion to get the actual result value.

Error raised and rectified by developer.

Recursion test done and found AssertionError: actual result is 17 and expected result is 24

Error raised and Code refactored.

Recursion test done and asserted result is 24 as expected

1. Test Perfect Game

Code has AssertionError: result not matched and asked to change the method of assertion to get the actual result value.

Error raised and rectified by developer.

Recursion test done and found AssertionError: actual result is 30 and expected result is 300

Error raised and Code refactored.

Recursion test done and asserted result is 300 as expected

1. Test All Spare

Code has AssertionError: result not matched and asked to change the method of assertion to get the actual result value.

Error raised and rectified by developer.

Recursion test done and found AssertionError: actual result is 20 and expected result is 150

Error raised and Code refactored.

Recursion test done and asserted result is 150 as expected

1. Test Invalid range

Test result achieved to be 0 as expected.

1. Test Invalid range

Test result achieved to be 50 as expected.

**Refactoring Code**:

Refactoring -1: In the main code method ‘score’ , if statement has error and it was calling the frameIndex method again in if Statement also isStrike method was missing from if condition. Rectify the code and introduce isStrike method in if statement with isSpare method.

Refactoring -2: in the main code method ‘score’. Return statement for result is within for loop statement, result value will be updated with the loop. Get the return statement outside the for loop statement.

Refactoring -3: its a syntax error for method name ‘StrikeScore’. Method name needs to be start from small letter. change the method name to strikeScore.

Refactoring -4: in the main code method roll, insert condition for pins value will be 0 is less than 0 or greater than 10.

Refactoring -4: in the main code method roll, insert condition for pins character is not int or float then pins value will be 0 and float will be converted to int.

After refactoring the code all test result passed and function is working perfectly fine

**Variances**

There is no variance while testing all eight-functionality mentioned in test plan. Application is working as expected without any error.

**Comprehensive Assessment:**

All test passed and code is working as expected.

.

**Conclusion:**

Based on test results all eight functionalities are working correctly as expected.

Next Steps:

* + 1. Method to validate the data input format can be created.
    2. Work on GUI interface.
    3. Test again and Remove error or bug, if any.

**Summary of Results:**

Total 8 test was performed.

* Test Gutter Game
* Test All One
* Test One Spare
* Test One Strike
* Test Perfect Game
* Test All Spare
* Test invalid range of rolls value
* Test invalid character of rolls value

Result of all 8 test is passed.

**Summary of Activities**

This testing activity include white box testing only. Where we have use functionality test for Bowling game functions. Test has been conducted as per the testing environment and schedule. Test was started on 27th June 2023 at 13:00 hours and all eight-test finished by 16:30 hours.

We found all tests are passed and all functionality are working as expected.

**Approvals**

|  |  |  |
| --- | --- | --- |
|  | **Project Manager** | **Test Lead** |
| **Name** |  |  |
| **Date** |  |  |
| **Signature** |  |  |