**System Test Plan**

**For**

**<Project Name>**

Written by:

Date:

Table of Contents

[1. Introduction 2](#_Toc225217702)

[1.1 Purpose 2](#_Toc225217703)

[1.2 Objectives 2](#_Toc225217704)

[2. Functional Scope 2](#_Toc225217705)

[3. Overall Strategy and Approach 2](#_Toc225217706)

[3.1 Testing Strategy 2](#_Toc225217707)

[3.2 System Testing Entrance Criteria 2](#_Toc225217708)

[3.3 Testing Types 2](#_Toc225217709)

[3.4 Suspension Criteria and Resumption Requirements 3](#_Toc225217710)

[3.5 Test Data 3](#_Toc225217711)

[4. Execution Plan 3](#_Toc225217712)

[4.1 Execution Plan 3](#_Toc225217713)

[5. Defect Reporting 3](#_Toc225217714)

[5.1 Defect Tracking 3](#_Toc225217715)

[5.2 Defect Reporting and Reports 3](#_Toc225217716)

[5.3 Defect Management Process 4](#_Toc225217717)

[5.4 Defect Severity Definitions 4](#_Toc225217718)

[6. Environment 4](#_Toc225217719)

[6.1 Environment 4](#_Toc225217720)

[7. Test Schedule 4](#_Toc225217721)

[8. Assumptions 4](#_Toc225217722)

[9. Risks and Contingencies 4](#_Toc225217723)

[10. Who to Call List 5](#_Toc225217724)

[11. Appendices 5](#_Toc225217725)

# Introduction

## Purpose

This document is a test plan for <Project Name> System Testing, produced by the System Testing team. It describes the testing strategy and approach to testing the team will use to verify that the application meets the established requirements of the business prior to release.

## Objectives

* Meets the requirements, specifications and the Business rules.
* Supports the intended business functions and achieves the required software standards.
* Satisfies the Entrance Criteria for User Acceptance Testing.

# Functional Scope

The Modules in the scope of testing for the <Project Name> System Testing are mentioned in the document attached in the following path :

# Overall Strategy and Approach

## Testing Strategy

<Project Name> System Testing will include testing of all functionalities that are in scope (Refer Functional Scope Section) identified. System testing activities will include the testing of new functionalities, modified functionalities, screen level validations, work flows, functionality access, testing of internal & external interfaces.

## System Testing Entrance Criteria

In order to start system testing, certain requirement must be met for testing readiness. The readiness can be classified into:

## Testing Types

### Usability Testing

User interface attributes, cosmetic presentation and content will be tested for accuracy and general usability. The goal of Usability Testing is to ensure that the User Interface is comfortable to use and provides the user with consistent and appropriate access and navigation through the functions of the application (e.g., access keys, consistent tab order, readable fonts etc.)

### Functional Testing

The objective of this test is to ensure that each element of the component meets the functional requirements of the business as outlined in the:

* Business / Functional Requirements
* Business rules or conditions
* Other functional documents produced during the course of the project i.e. resolution to issues/change requests/feedback

## Suspension Criteria and Resumption Requirements

This section will specify the criteria that will be used to suspend all or a portion of the testing activities on the items associated with this test plan.

### Suspension Criteria

Testing will be suspended if the incidents found will not allow further testing of the system/application under-test. If testing is halted, and changes are made to the hardware, software or database, it is up to the Testing Manager to determine whether the test plan will be re-executed or part of the plan will be re-executed.

### Resumption Requirements

Resumption of testing will be possible when the functionality that caused the suspension of testing has been retested successfully.

## Test Data

Test data requirements are drawn up based on the functional requirements that are due for testing. The testing team will identify test cases that can be grouped into test scenarios and detail the data required to complete the testing activities.

# Execution Plan

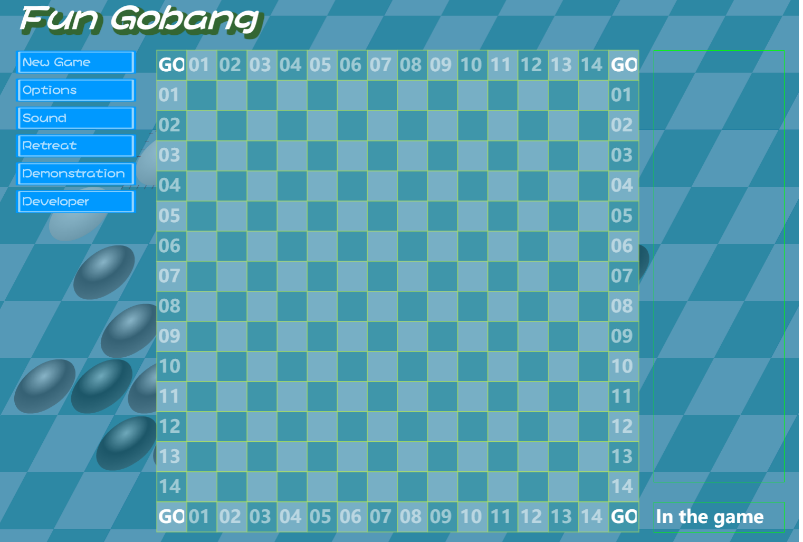
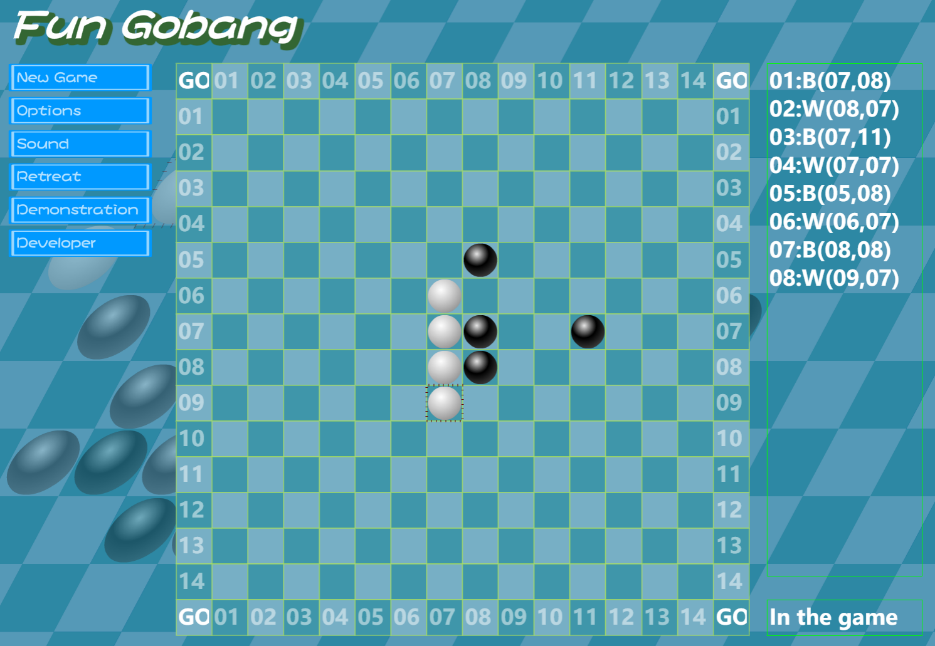
## Execution Plan

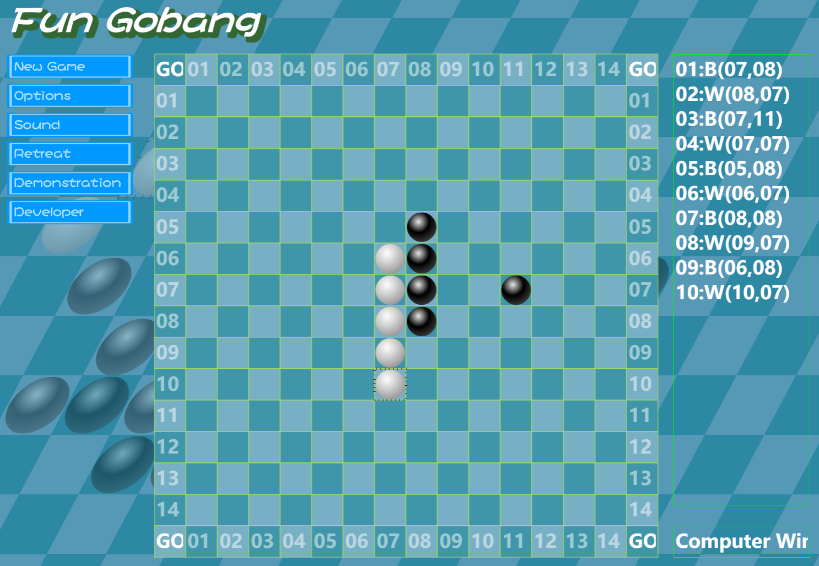
The execution plan will detail the test cases to be executed. The Execution plan will be put together to ensure that all the requirements are covered. The execution plan will be designed to accommodate some changes if necessary, if testing is incomplete on any day. All the test cases of the projects under test in this release are arranged in a logical order depending upon their inter dependency.

**4.2.2 Winning Judgment Test Case**

4.2.2.1 Judge\_Test\_001

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Winning\_Judge | | | | **Test Case ID** | | Judge\_001 | |
| **Test Case Description** | | Verify whether system can identify an unbroken row of five stones vertically. And finish the game. | | | | **Test Priority** | | High | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. Don’t save game. | |
| Test Execution Steps | | | | | | | | | |
| **S.NO** | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 14:00 AM]:  Launch successful |
| 2 | Place pieces that make unbroken row of four stones vertically | | Click crosses on the chessboard to place pieces | Pieces can be placed on the chessboard. | Pieces can be placed on the chessboard. | | Pass | | [Beijing 06/08/2022 14:05 AM]:  Placed successful |
| 3 | Place the last piece that make unbroken row of five stones vertically | | Click cross on the chessboard to place the piece | There are unbroken five stones vertically. System show “白旗获胜”.  And show a dialog ask whether to save  game. | There are unbroken five stones vertically. System show “白旗获胜” and show the dialog. | | Pass | | [Beijing 06/08/2022 14:30 AM]:  Winning judgment successful |

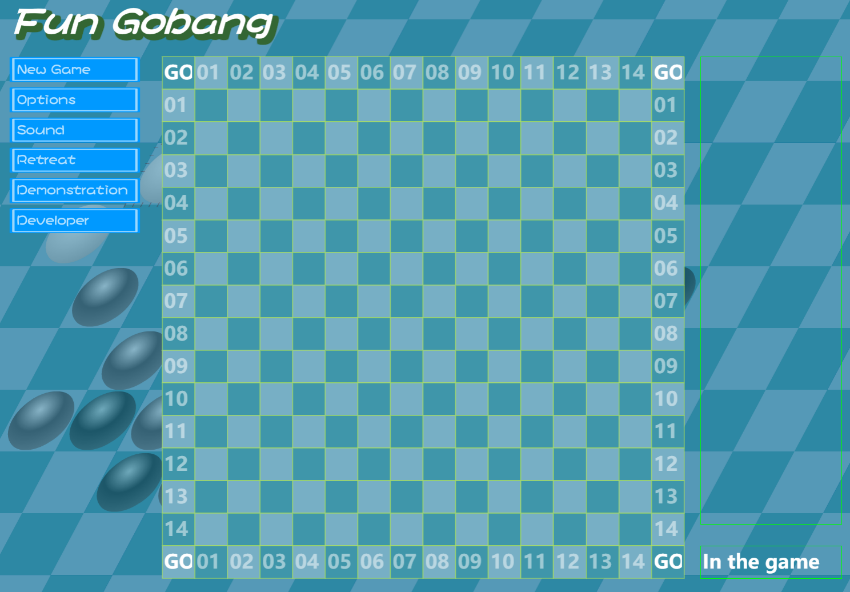
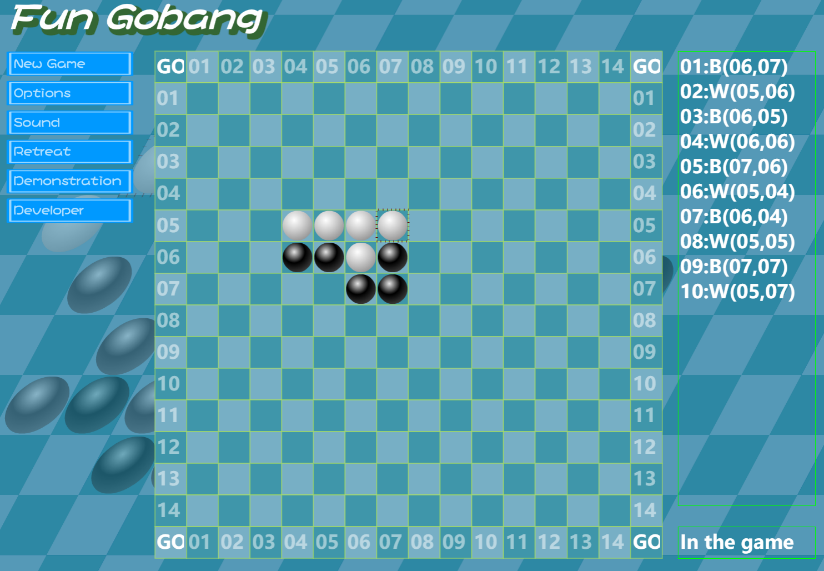
Start new game

Make unbroken row of four stones vertically

Unbroken five stones vertically. Show “白棋获胜”.

4.2.2.2 Judge\_Test\_002

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Winning\_Judge | | | | **Test Case ID** | | Judge\_002 | |
| **Test Case Description** | | Verify whether system can identify an unbroken row of five stones horizontally. And finish the game. | | | | **Test Priority** | | High | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. Don’t save game. | |
| Test Execution Steps | | | | | | | | | |
| **S.NO** | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 14:00 AM]:  Launch successful |
| 2 | Place pieces that make unbroken row of four stones horizontally | | Click crosses on the chessboard to place pieces | Pieces can be placed on the chessboard. | Pieces can be placed on the chessboard. | | Pass | | [Beijing 06/08/2022 14:05 AM]:  Placed successful |
| 3 | Place the last piece that make unbroken row of five stones horizontally | | Click cross on the chessboard to place the piece | There are unbroken five stones horizontally. System show “白旗获胜”.  And show a dialog ask whether to save  game. | There are unbroken five stones horizontally.  System show “白旗获胜” and show the dialog. | | Pass | | [Beijing 06/08/2022 14:10 AM]:  Winning judgment successful |

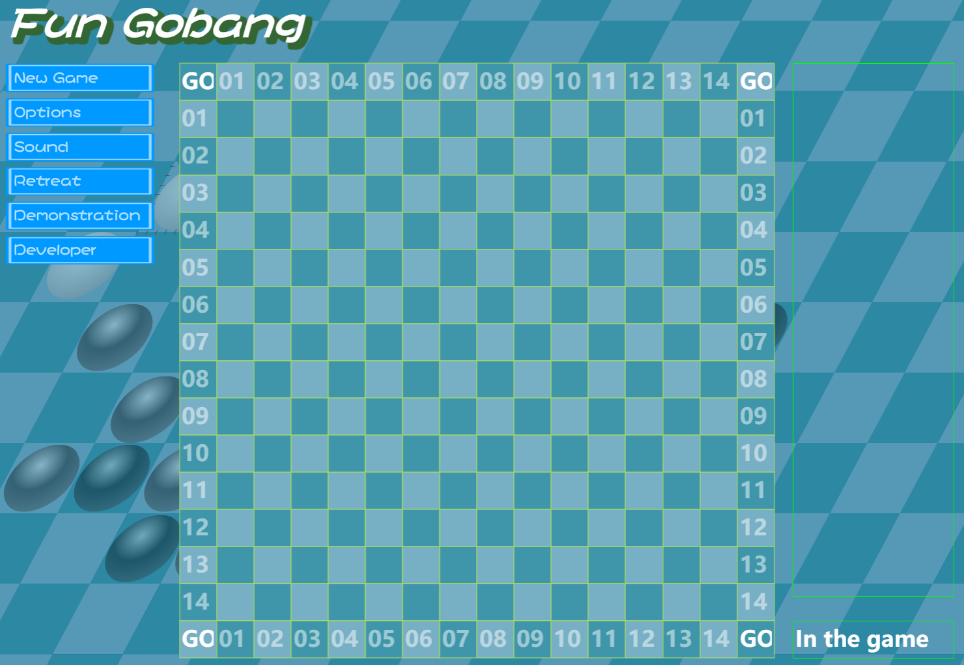
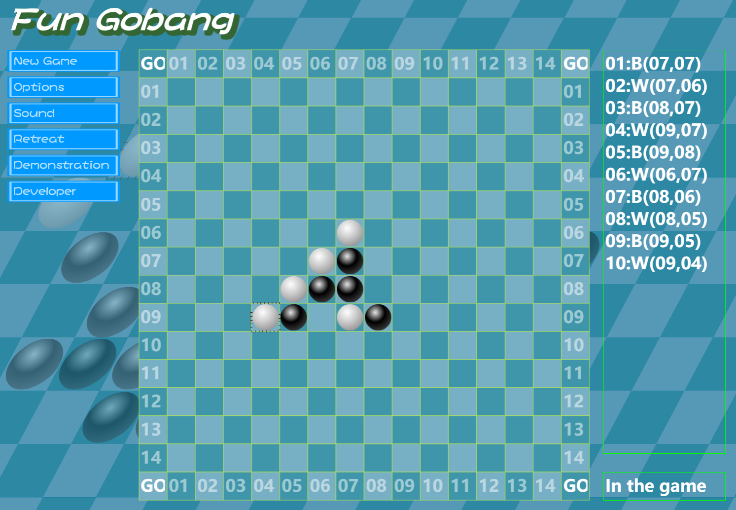
Start new game

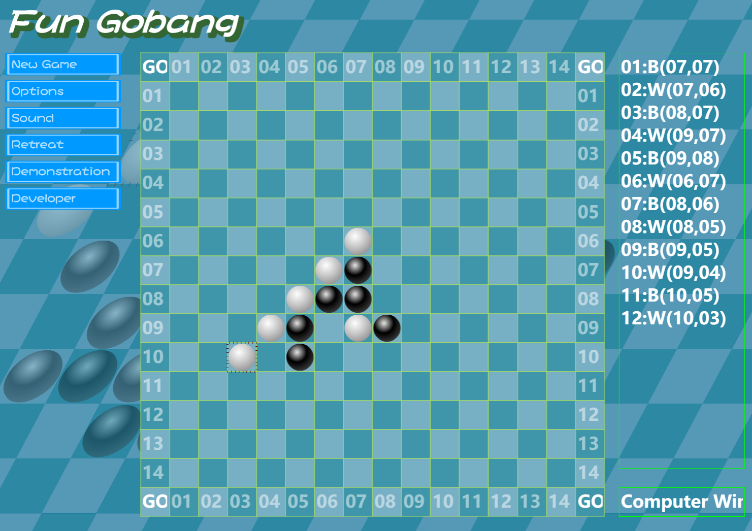
****Make unbroken row of four stones horizontally

Unbroken five stones horizontally. Show “白棋获胜”.

4.2.2.3 Judge\_Test\_003

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Winning\_Judge | | | | **Test Case ID** | | Judge\_003 | | |
| **Test Case Description** | | Verify whether system can identify an unbroken row of five stones bottom-left to top-right diagonally(左下到右上). And finish the game. | | | | **Test Priority** | | High | | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. Don’t save game. | | |
| Test Execution Steps | | | | | | | | | | |
| **S.NO** | | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 14:40 AM]:  Launch successful |
| 2 | | Place pieces that make unbroken row of four stones diagonally | | Click crosses on the chessboard to place pieces | Pieces can be placed on the chessboard. | Pieces can be placed on the chessboard. | | Pass | | [Beijing 06/08/2022 14:45 AM]:  Placed successful |
| 3 | | Place the last piece that make unbroken row of five stones bottom-left to top-right diagonally | | Click cross on the chessboard to place the piece | There are unbroken five stones diagonally. System show “白棋 获 胜 ”. And show a dialog ask whether to save game. | There are unbroken five stones diagonally. System show “白棋 获 胜 ” and show the dialog. | | Pass | | [Beijing 06/08/2022 14:50 AM]:  Winning judgment successful |

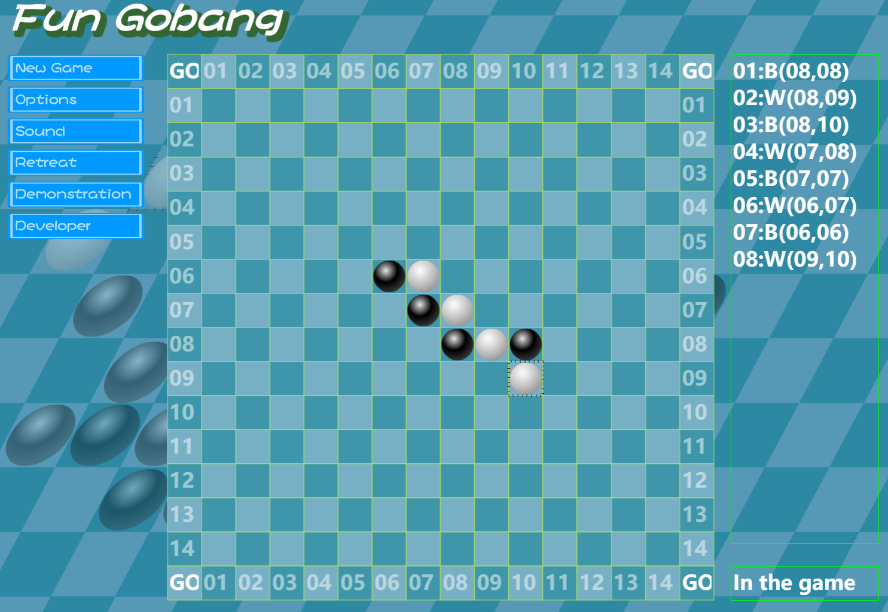
****Start new game

****Make unbroken row of four stones bottom-left to top-right diagonally

Unbroken five stones diagonally. Show “白旗获胜”.

4.2.2.4 Judge\_Test\_004

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Winning\_Judge | | | | **Test Case ID** | | Judge\_004 | | |
| **Test Case Description** | | Verify whether system can identify an unbroken row of five stones top-left to bottom-right diagonally(左上到右下). And finish the game. | | | | **Test Priority** | | High | | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. Don’t save game. | | |
| Test Execution Steps | | | | | | | | | | |
| **S.NO** | | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 15:00 AM]:  Launch successful |
| 2 | | Place pieces that make unbroken row of four stones diagonally | | Click crosses on the chessboard to place pieces | Pieces can be placed on the chessboard. | Pieces can be placed on the chessboard. | | Pass | | [Beijing 06/08/2022 15:05 AM]:  Placed successful |
| 3 | | Place the last piece that make unbroken row of five stones top- left to bottom-right diagonally | | Click cross on the chessboard to place the piece | There are unbroken five stones diagonally. System show “白棋 获 胜 ”. And show a dialog ask whether to save game. | There are unbroken five stones diagonally. System show “白棋 获 胜 ” and show the dialog. | | Pass | | [Beijing 06/08/2022 15:10 AM]:  Winning judgment successful |

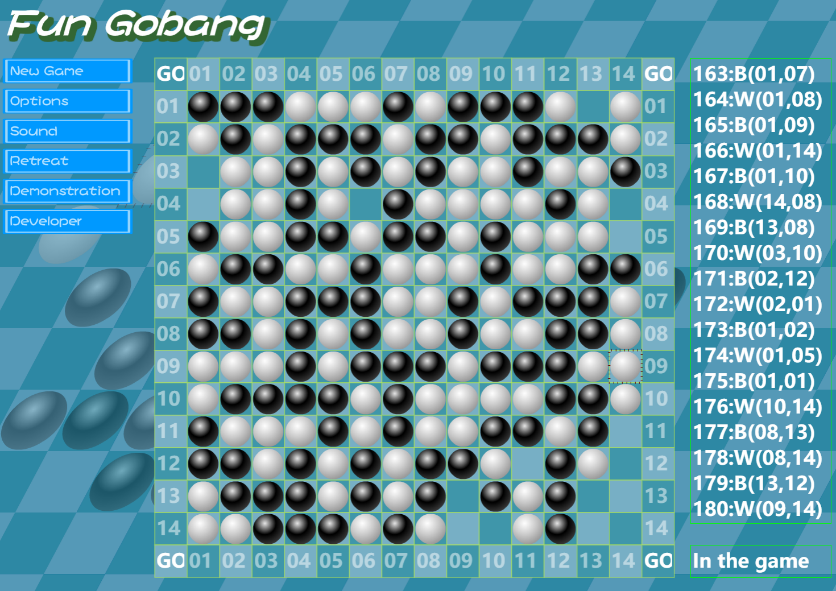
****Start new game

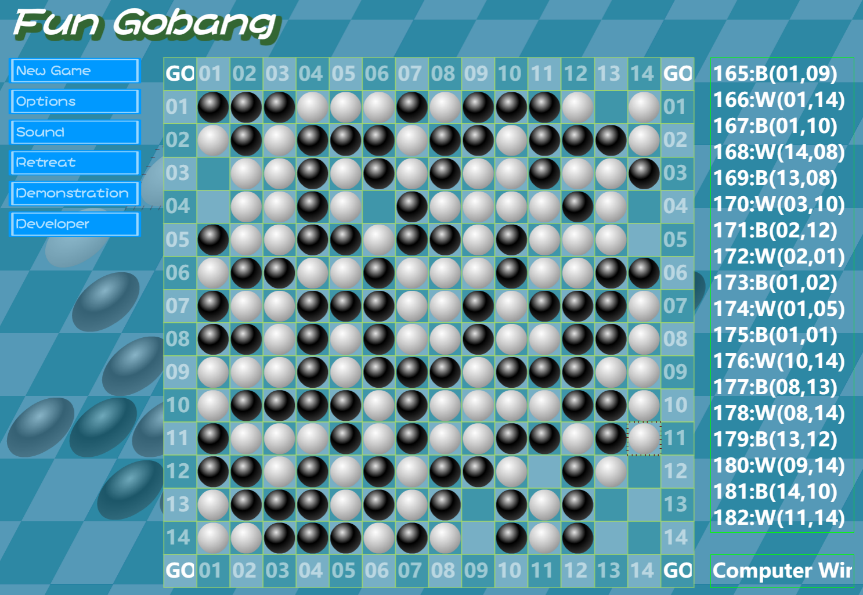
****Make unbroken row of four stones top-left to bottom-right diagonally

Unbroken five stones diagonally. Show “白棋获胜”.

4.2.2.5 Judge\_Test\_005

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Winning\_Judge | | | | **Test Case ID** | | Judge\_005 | | |
| **Test Case Description** | | Verify whether system can identify an unbroken row of five stones near the boundary. | | | | **Test Priority** | | High | | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. Don’t save game. | | |
| Test Execution Steps | | | | | | | | | | |
| **S.NO** | | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 15:20 AM]:  Launch successful |
| 2 | | Place pieces that make unbroken row of four stones near the corner of  boundary. | | Click crosses on the chessboard to place pieces | Pieces can be placed on the chessboard. | Pieces can be placed on the chessboard. | | Pass | | [Beijing 06/08/2022 15:25 AM]:  Placed successful |
| 3 | | Place the last piece that make unbroken five stones in the boundary. | | Click cross on the chessboard to place the piece | There are unbroken five stones in a line. System show “白棋获胜”. And show a dialog ask whether to save  game. | There are unbroken five stones in a line. System show “白棋 获 胜 ” and show the dialog. | | Pass | | [Beijing 06/08/2022 15:30 AM]:  Winning judgment in boundary successful |

Start new game ****

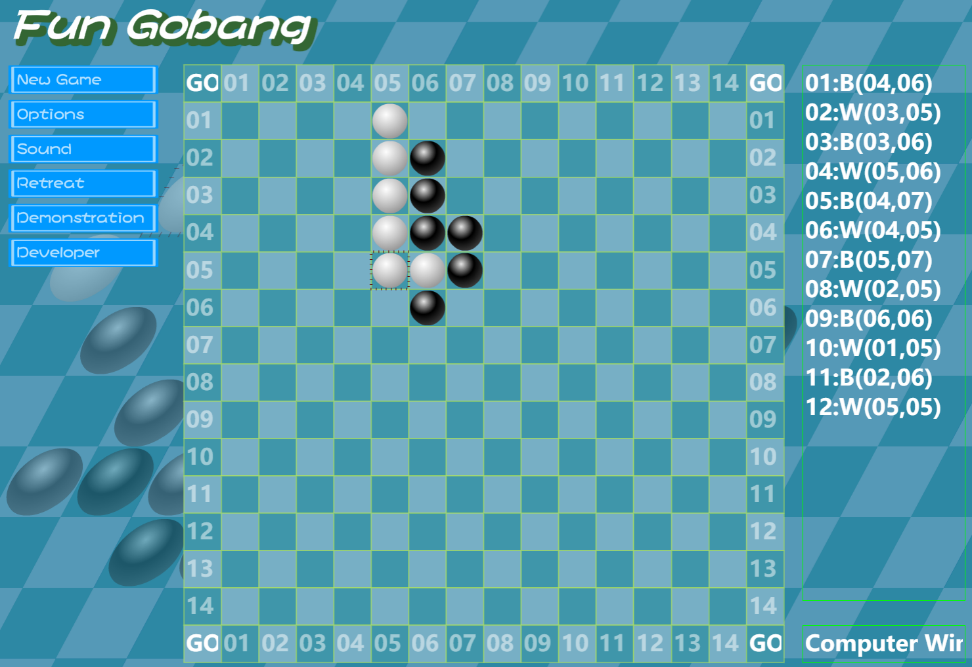
Make unbroken row of four stones near the boundary

Unbroken five stones near the boundary. Show “白棋获胜”.

* + 1. **Finished game demonstration**
       1. ****Demonstration\_Test\_001

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Finished game demonstration | | | | **Test Case ID** | | Demonstration\_001 | | |
| **Test Case Description** | | Verify whether system can shore the unfinished game, and reload the game after. In the correct process. | | | | **Test Priority** | | High | | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. | | |
| Test Execution Steps | | | | | | | | | | |
| **S.NO** | | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 15:40 AM]:  Launch successful |
| 2 | | Place the last piece that make unbroken five stones in the boundary. | | Click cross on the chessboard to place the piece | There are unbroken five stones in a line. System show “白棋获胜”. And show a dialog ask whether to save  game. | There are unbroken five stones in a line. System show “白棋 获 胜 ” and show the dialog. | | Pass | | [Beijing 06/08/2022 15:50 AM]:  Winning judgment in boundary successful |
| 3 | | Click “demonstration ” button | | Null | System show the last game that ended | Show the last game that ended. | | Pass | | [Beijing 06/08/2022 15:55 AM]:  Demonstration successful |

Start new game

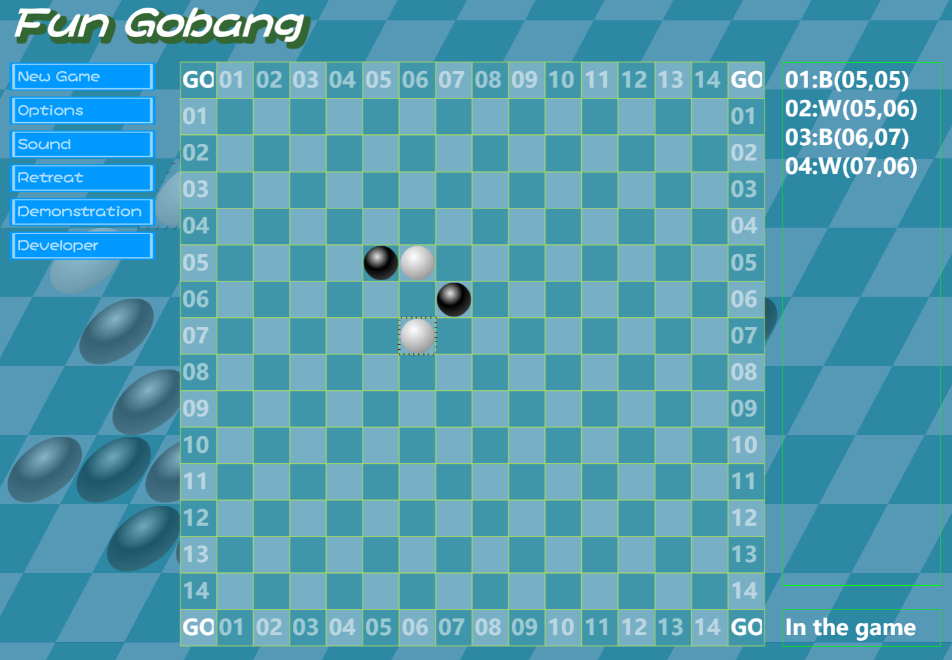
Unbroken five stones vertically. Show “白棋获胜”.

System show the last game that ended

* + - 1. ****Demonstration\_Test\_002

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | Finished game demonstration | | | | **Test Case ID** | | Demonstration\_002 | | |
| **Test Case Description** | | Verify whether system can shore the unfinished game, and reload the game after. In the correct process. | | | | **Test Priority** | | High | | |
| **Pre-Requisite** | | Start a new game. | | | | **Post Requisite** | | Come back to main menu page. | | |
| Test Execution Steps | | | | | | | | | | |
| **S.NO** | | **Action** | | **Inputs** | **Expected Output** | **Actual Output** | | **Test result** | | **Test Comments** |
| 1 | | Start new game. | | Jack John | Start new game successful. | Start new game successful. | | Pass | | [Beijing 06/08/2022 15:40 AM]:  Launch successful |
| 2 | | Place some pieces on the chessboard. | | Click crosses on the chessboard to place pieces | Pieces can be placed on the chessboard. And game is not over. | Pieces can be placed on the chessboard. And game is not over. | | Pass | | [Beijing 06/08/2022 15:45 AM]:  Placed successful |
| 3 | | Click “demonstration ” button | | Null | System show”you can use this button when one round finshed” | System show”you can use this button when one round finshed” | | Pass | | [Beijing 06/08/2022 15:55 AM]:  Demonstration unsuccessful |

Start new game

Place some pieces

Game paused

# Defect Reporting

## Defect Tracking

<Tool Name> will be used for defect/Issue tracking.

## Defect Reporting and Reports

Defects will be reported until ------------------------- daily. Defect reports will be generated by ----------------------------------------------to be reviewed and analyzed during the daily ------------------------------------defects meeting. The reports will be saved on the network and can be found by accessing the below link:

## Defect Management Process

<Define defect management process>

## Defect Severity Definitions

|  |  |
| --- | --- |
| **Critical** | The defect causes a catastrophic or severe error that results in major problems and the functionality rendered is unavailable to the user. A manual procedure cannot be either implemented or a high effort is required to remedy the defect. Examples of a critical defect are as follows:   * System abends * Data cannot flow through a business function/lifecycle * Data is corrupted or cannot post to the database |
| **Medium** | The defect does not seriously impair system function can be categorized as a medium Defect. A manual procedure requiring medium effort can be implemented to remedy the defect. Examples of a medium defect are as follows:   * Form navigation is incorrect * Field labels are not consistent with global terminology |
| **Low** | The defect is cosmetic or has little to no impact on system functionality. A manual procedure requiring low effort can be implemented to remedy the defect. Examples of a low defect are as follows:   * Repositioning of fields on screens * Text font on reports is incorrect |

# Environment

## Environment

* The System Testing Environment will be used for System Testing.

# Test Schedule

System testing is scheduled for a period of x weeks starting mm/dd/yyyy to mm/dd/yyyy. The test team will complete the execution of all the tests during the first x weeks. The defects retesting and regression testing will occur in the last week of System Testing. The run dates for defect retesting period may be changed according to the need to retest and close the defects. The defects retesting will reduce the number of open defects that need to be carried to UAT.

# Assumptions

* Define test plan assumptions..

# Risks and Contingencies

Define risks and contingencies.

# Appendices

# 