DocNo: 001.I.1:1

Test Analysis Report

Version 1.0

**Group Member:**

施宇

鲁皓

万成城

曹翼丰

**Document Language:**

English

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 2015-11-20 | <1.0> | Finish the main structure of the document | 万成城 |
|  |  |  |  |
|  |  |  |  |

**Key Word**

Chinese chess, test analysis report

**Digest**

This document is to record the test phase. In this document, there are test analysis report and describe the test case and suggestions.

1. Introduction 4

1.1. Purpose 4

1.2. Background 4

1.3. Defination 4

1.4. Reference 5

2. Test Overview 5

3. Test Result and Findings 5

3.1. Function Test 5

3.2. Testcase Result 5

3.2.1. Basic logic 5

3.2.2. Artificial Intelligence 6

3.2.3. Network Connection 6

3.2.4. Game Mode 6

3.2.5. 3D Effect 6

4. Analysis Abstracts 7

4.1. Capability 7

4.2. Flaws and Limitations 7

4.2.1. Some existed problems 7

4.2.2. Some unrealized functions 7

4.3. Suggestions 7

4.4. Evaluation 7

5. Test Cost 7

# Introduction

## Purpose

This document is our test analysis report for the Chinese Chess game, which illustrates the details of test result according to test plan document. This document is the main reference for our testing. Therefore, the readers for this document are mainly the testers and the manager of the project.

## Background

The whole project began at October 2015. The project has finished requirement analysis, system designing, and coding. After coding out the whole system, the testers have the general idea of the whole project. They master the testing techniques and start the test.

## Defination

1. Black box testing: A test method, which testers only pay attention to input and output.
2. White box testing: A test method, which testers know the inside instruction of test object.
3. Test script: A small teat program for testing to call unit or be called by unit.
4. Equivalence partition: A test method in black box testing. It uses a set of values selected, instead of many input value, which are dealt with in the same way.
5. Boundary designing: It is the extension of the equivalence partition; usually it is the boundary of equivalent class.
6. Unit testing: Test on the smallest unit, such as class, in the software.
7. Integration testing: Test on the combination of several units to check if they can work together.
8. Regression testing: In integration testing, some integration must be test again to check if they can work with other integrations.
9. System testing: Compared with requirement definition, look for some parts which are not coincident with the requirement.
10. Stress testing: Test if the system can afford heavy using stress.

## Reference

*“Object-oriented Software Engineering, Using UML, Patterns, and Java, Third Edition”*

byBernd Bruegge and Allen H. Dutoit

Test Plan Document

# Test Overview

|  |  |
| --- | --- |
| Module | Function |
| Basic Logic | Manage the whole chess board and decide whether the game ends |
| Artificial Intelligence | Support a man-machine chess game |
| Network Connection | Support a man-man chess game |
| Game Mode | Support multiple game modes |
| 3D Effect | Support 3D movements of hoodles |

# Test Result and Findings

## Function Test

All the function runs well.

## Testcase Result

### Basic logic

|  |  |  |  |
| --- | --- | --- | --- |
| Test Name | Action | Expect Result | Actual Result |
| Basic Logic 1 | Input a legal movement | Program goes well. | Fit expectation |
| Basic Logic 2 | Input an illegal movement | No movement happens. | Fit expectation |

### Artificial Intelligence

|  |  |  |  |
| --- | --- | --- | --- |
| Test Name | Action | Expect Result | Actual Result |
| Artificial Intelligence 1 | Start a man-machine chess game | Program goes well. | Fit expectation |
| Artificial Intelligence 2 | Start a machine-machine chess game | Program goes well. | Fit expectation |

### Network Connection

|  |  |  |  |
| --- | --- | --- | --- |
| Test Name | Action | Expect Result | Actual Result |
| Network Connection 1 | Start a man-man chess game with network | Program goes well. | Fit expectation |
| Network Connection 1 | Start a man-man chess game without network | Turn to a man- machine chess game. | Fit expectation |

### Game Mode

|  |  |  |  |
| --- | --- | --- | --- |
| Test Name | Action | Expect Result | Actual Result |
| Game Mode 1 | Start a man-machine chess game with different modes. | Program goes well. | Fit expectation |
| Game Mode 2 | Start a machine-machine chess game with different modes. | Program goes well. | Fit expectation |

### 3D Effect

|  |  |  |  |
| --- | --- | --- | --- |
| Test Name | Action | Expect Result | Actual Result |
| 3D Effect 1 | Start a man-machine chess game with different 3D-effects. | Program goes well. | Fit expectation |
| 3D Effect 2 | Start a machine-machine chess game with different 3D-effects. | Program goes well. | Fit expectation |

# Analysis Abstracts

## Capability

The Tested Chinese Chess game have the follow functions:

1. Multi-players, including both AI and online players.
2. Multiple game modes.
3. 3D effect

## Flaws and Limitations

### Some existed problems

1. Artificial intelligence

The AI is not smart enough for lack of chess book of 6 players

1. Sound effect

The sound effect is plain and simple.

### Some unrealized functions

All functions are realized.

## Suggestions

Repeal functions can be added to man-machine game.

## Evaluation

The Chinese chess game can run in a correct and efficiency way, and has high immersion characteristic.

# Test Cost

No special cost is spend but the personnel work time.