Group 10

Author: zhuwangshu

Software VALIDATION

Pocker Game

Table of Contents

[System Architecture 2](#_Toc11030927)

[T1: Unit Test 2](#_Toc11030928)

[T1.1: start Test 2](#_Toc11030929)

[T1.2: Lord test 3](#_Toc11030930)

[T1.3 3](#_Toc11030931)

[T1.4 3](#_Toc11030932)

[T1.5 4](#_Toc11030933)

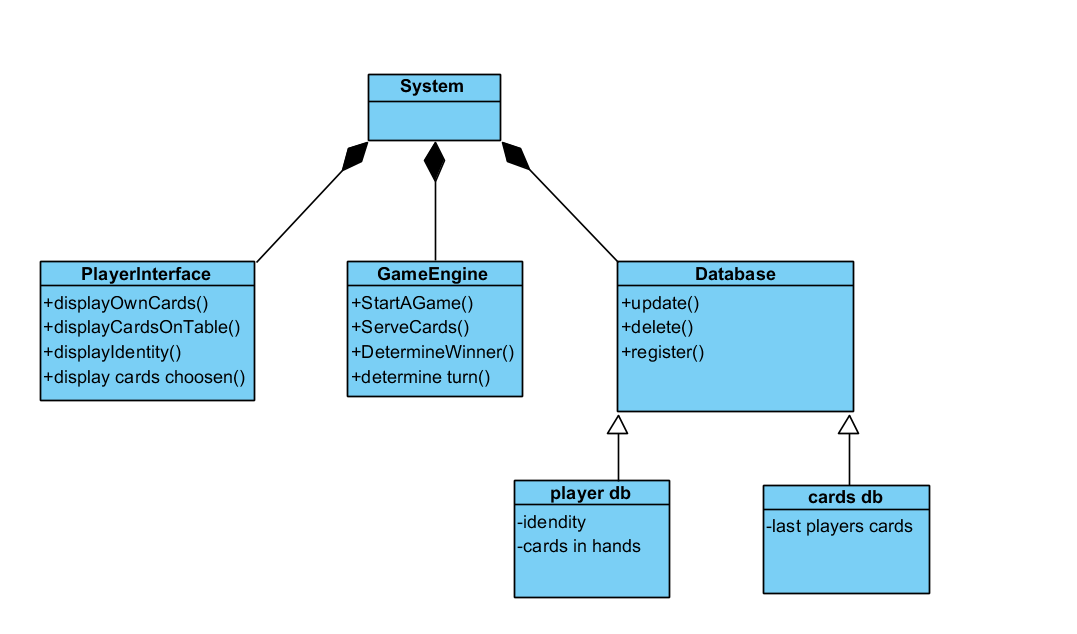
[T1.6 4](#_Toc11030934)

[T1.7 ilegal cards 5](#_Toc11030935)

[T2: Functional Test 6](#_Toc11030936)

## System Architecture

The system architecture is shown below:



## T1: Unit Test

### T1.1: start Test

testCase.verifyEqual(testCase.UI1.messageLabel.Text,{'welcome to pocker game !'} );

testCase.verifyEqual(testCase.UI2.messageLabel.Text,{'welcome to pocker game !'} );

testCase.verifyEqual(testCase.UI3.messageLabel.Text,{'welcome to pocker game !'} );

|  |  |
| --- | --- |
|  | Test Case T1.1 |
| Coverage Item | Tcover1.1 |
| Input | testCase.press(testCase.UI1.startButton);  testCase.press(testCase.UI2.startButton);  testCase.press(testCase.UI3.startButton); |
| State | Master = MasterServer2;  testCase.UI1 = playerUItest;  testCase.UI2 = playerUItest;  testCase.UI3 = playerUItest;    testCase.UI1.MasterProcessor=Master;  testCase.UI2.MasterProcessor=Master;  testCase.UI3.MasterProcessor=Master;    Master.Player1UI = testCase.UI1;  Master.Player2UI = testCase.UI2;  Master.Player3UI = testCase.UI3; |
| Expected Output | 'welcome to pocker game !' |

* Test coverage: 1/1=100%
* Test result: 1 passed

## T1.2: Lord test

Hat is visible on the lord’s head

## T1.3

Cards be played is shown correctly on the table

## T1.4

testCase.verifyEqual(testCase.UI1.messageLabel.Text,"they all pass " );

|  |  |
| --- | --- |
|  | Test Case T1.1 |
| Coverage Item | Tcover1.1 |
| Input | testCase.press(testCase.UI2.passButton);    testCase.press(testCase.UI3.passButton); |
| State | Master = MasterServer2;  testCase.UI1 = playerUItest;  testCase.UI2 = playerUItest;  testCase.UI3 = playerUItest;    testCase.UI1.MasterProcessor=Master;  testCase.UI2.MasterProcessor=Master;  testCase.UI3.MasterProcessor=Master;    Master.Player1UI = testCase.UI1;  Master.Player2UI = testCase.UI2;  Master.Player3UI = testCase.UI3; |
| Expected Output | "they all pass " |

* Test coverage: 1/1=100%

## T1.5

See boom , airplane ,wangzha picture on table

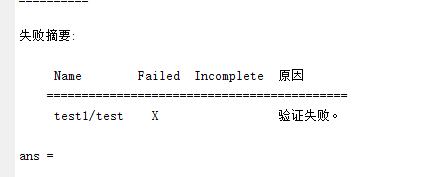
## T1.6

Gameover

testCase.verifyEqual(testCase.UI2.messageLabel.Text,'game over lord is the winner' );

|  |  |
| --- | --- |
|  | Test Case T1.1 |
| Coverage Item | Tcover1.1 |
| Input | testCase.press(testCase.UI2.passButton);    testCase.press(testCase.UI3.passButton); |
| State | Master = MasterServer2;  testCase.UI1 = playerUItest;  testCase.UI2 = playerUItest;  testCase.UI3 = playerUItest;    testCase.UI1.MasterProcessor=Master;  testCase.UI2.MasterProcessor=Master;  testCase.UI3.MasterProcessor=Master;    Master.Player1UI = testCase.UI1;  Master.Player2UI = testCase.UI2;  Master.Player3UI = testCase.UI3; |
| Expected Output | "they all pass " |

* There is a little bug ,the windows has shown “game over lord is the winer” but it always failed the test.

失败摘要也没有错误原因记录 

## T1.7 ilegal cards

testCase.verifyEqual(testCase.UI1.messageLabel.Text,"illegal cards!!!!! choose again " );

pause(1);

|  |  |
| --- | --- |
|  | Test Case T1.1 |
| Coverage Item | Tcover1.1 |
| Input | testCase.press(testCase.UI1.card12Button);  pause(1);  testCase.press(testCase.UI1.card13Button);  pause(1);  testCase.press(testCase.UI1.card14Button);  pause(1);  testCase.press(testCase.UI1.card15Button);  pause(1);  testCase.press(testCase.UI1.card16Button);  pause(1);  testCase.press(testCase.UI1.card17Button);  pause(1);  testCase.press(testCase.UI1.card10Button);  pause(1);  testCase.press(testCase.UI1.card11Button);  pause(0.3);  testCase.press(testCase.UI1.card9Button);  pause(0.3);  testCase.press(testCase.UI1.playButton); |
| State | Master = MasterServer2;  testCase.UI1 = playerUItest;  testCase.UI2 = playerUItest;  testCase.UI3 = playerUItest;    testCase.UI1.MasterProcessor=Master;  testCase.UI2.MasterProcessor=Master;  testCase.UI3.MasterProcessor=Master;    Master.Player1UI = testCase.UI1;  Master.Player2UI = testCase.UI2;  Master.Player3UI = testCase.UI3; |
| Expected Output | "illegal cards!!!!! choose again " |

* Test coverage: 1/1=100%

## T2: Functional Test

第二局 本局包含测试规则：单张 对子 连对 四带二 炸弹 王炸

开始发牌

玩家 1 3 3 4 4 7 8 10 J Q Q K K K A A A 大王

玩家 2 3 3 5 5 5 5 7 7 8 8 9 9 10 J Q A 2

玩家 3 4 4 6 6 6 6 7 8 9 9 10 10 J J Q 2 2

底牌 K 2 小王

Oder UI table to see the cards played more clearly and without the UI effect test speed can be faster