**TESTING RAPOR FOR CHESS PROJECT**

**Prepared by:**

Group 8

AHMET -

ALİ EMRE BÜYÜKERSOY - 111044020

EGEMEN ÇAĞRIŞLI -

HALİL KÖSE -

MEHMET ÖZTÜRK -

SELMAN AHATLI - 111044038

YASİN TULUMEN - 121044020

RECEP SİVRİ - 121044038

GÖZDE DOĞAN - 131044019

YASİN AÇIKGÖZ - 131044070

CONTENTS

[1. SELECTED METHOD 3](#_Toc468924022)

[2. SELECTED METHOD PROPERTIES 3](#_Toc468924023)

[3. CONTROL FLOW GRAPH 4](#_Toc468924024)

[4. FULL STATEMENT COVERAGE PATHS 10](#_Toc468924025)

[5. INPUTS THAT PROVIDES PATHS 10](#_Toc468924026)

[6. TABLO 10](#_Toc468924027)

# SELECTED METHOD

start method selected in main.java file.

->public void start (Stage primaryStage) throws Exception

The selected method call in main method.

# SELECTED METHOD PROPERTIES

In start metod contains all event handlers for graphical user interface which are all event handlers of chess table which are 64 buttons.These buttons changes their css properties for example if we clicked to a button which contains pawn then we clicked to another button which doesn’t contains any piece than pawn making move from first place to second place with changing own css properties.we added menu to graphical user interface so we added this property to start method. In start method we added grid properties which is a property of JavaFx , this property provides us setting layout in every place that want by us. In start method contains the event handler that provides playing chess that changing css properties 64 buttons and another event handler provides restart game which is event handler of restart button and previous game event handler provides that show previous status of the game that is event handler of previous button.

|  |
| --- |
| window=primaryStage  Space  1  Grid  Open  Restart  i = 0  j = 0 |

# CONTROL FLOW GRAPH

2

i<8

4

i = 7

3

i >= 0

Button[i]

j=0

7

False True False

5

j<8

True

8

j= 7

--i

button[i][j] = new ExtendButton()

a=i

b=j

9

6

j >= 0

True False

12

10

Button[i][j]

True False

--j

GridPane.setConstraints(button[i][j], 7-i, 7-j)

11

refreshTable()

CurrentButton = e.getSource()

i=7

13

startingStatusHandler || game.getIsComputerOn() ==0

i>=0

True False

14

j>=0

i--

j=7

24

15

True False

currentStatus=game.playGame(..)

25

True False

16

26

17

game.getIsComputerOn()!=0 && (currentStatus==2 || !startingStatusHandler)

j--

GridPane.setConstraints(button[i][j], 7-i, 7-j)

18

setButtoons(button)

other operations

True False

19

**game.getIsComputerOn()==1**

i=7

True

27

21

i>=0

True False False

20

game.playComputerEasy()

i--

j=7

28

23

j>=0

j--

grid.getChilderen().addAll(button[i][j])

22

29

**game.getIsComputerOn()==2**

Other operations

Previous.setDisable(true)

31

True False

30

**game.getIsComputerOn()==3**

game.playComputerMedium()

32

i=0

33

i<8

True False

38

i++

j=0

34

j<8

True False False

35

currentStatus==1

startingStatusHandler=true

tempMoveList.clear()

tempMovesList.addAll(game.playComputerHard())

True

39

button[i][j].setDisable(true)

40

36

41

Menu Operations

True False

43

42

37

a++

markButton(button[][])

a<tempMoveList.size()

a=markButton

44

True False

45

currentStatus==2

markButton(currentButton)

46

True

47

button[…][…].setStyle(…)

markButton=tempMoveList.size()

….

ListGame.add(temp)

Platform.exit()

currentStatus==3

49

48

51

50

True

52

fileChooser=new FileChooser()

…

extFilter=new FileChooser.ExtensionFilter(‘\*’, ‘\*’)

fileChooser.getExtensionFilters().add(extFilter)

file=fileChooser.showOpenDialog(null)

load

others operations

fileChooser=new FileChooser()

… extFilter=new ….()

file=fileChooser.showSaveDialog(primaryStage)

save

53

False

54

56

55

True

restart

load operations

file!=null

57

58

59

True False

60

restart

Open Menu

Menu Edit

Menu View

61

62

Open.Color==1

True False

66

64

63

startingStatusHandler=False

Open.Color==0

True False

currentStatus=0

game.restartGame()

True

67

startingStatusHandler=True

65

74

69

68

previous

a++

b=0

a<8

a=0

True False True

71

b<8

False

70

temp=ListGame.peek()

m

n

setButtons()

75

72

False

76

temp!=Null

True

b++

button

True

78

77

73

m=0

n=0

m<8

m=0

n=0

80

79

m<8

True False True

n<8

m++

n=0

85

81

m++

n=0

86

n<8

True False False

m<8

m=0

n++

button[m][n].setStyle(…)

87

82

True

n++

button[m][n].setStyle(…)

88

83

84

setButtoons(button)

True False

n<8

m++

n=0

89

90

False

91

True

92

n++

GridPane.setConstraits(…)

93

m=0

True False

m<8

94

96

95

n<8

m++

n=0

False

True

operations

project(about)

n++

grid.getChildren.addAll(..)

97

98

True False

start

99

100

True False

Open.Color==1

101

True False

startingStatusHandler=true

Open.Color==0

startingStatusHandler=false

102

103

True

a++

b=0

a<8

a=0

104

105

107

106

True

b<8

110

108

Restart.setDisable(false)

game.restartGame()

Previous.setDisable(false)

False

b++

button[a][b].setDisable(false)

109

Exit

111

True False

113

112

return;

Platform.exit()

# FULL STATEMENT COVERAGE PATHS

# INPUTS THAT PROVIDES PATHS

Inputs For Path1;

Inputs For Path2;

Inputs For Path3;

Inputs For Path4;

# TABLO

|  |  |  |  |
| --- | --- | --- | --- |
| Path | Excepted Output | Observed Output | Error |
| Path1 |  |  |  |
| Path2 |  |  |  |
| Path3 |  |  |  |
| Path4 |  |  |  |
| Path5 |  |  |  |
| Path6 |  |  |  |
| Path7 |  |  |  |
| Path8 |  |  |  |

Table : Path-Excepted Input-Observed Output-Error