Individual Report

# Components / objects on form

In the “StartGameFrom” we made use of:

* TextBoxes ; For receiving user input
* Labels : For organizing the form and instructions for players..
* Panels : For organizing and improving the design of the form.
* Buttons : For saving the players information and to swap to the next form.
* ButtonRest : For players to select what dragon type they want
* PictureBoxes : For giving a beter design to the form and so that the players can see what the dragons look like.

In the “TurnPlayerFrom” we used:

* Labels : For organization and making instructions clear, also displays the battle log and shows statistics.
* Buttons : Player gets to choose what their move is(Attack, Special Attack and Block).
* Panels : For organizing and design improvements
* PictureBoxes : To display what dragons the players chose and what dragons are in battle

# Functionality of Game

The StartGameForm is where the players will input the information that will be used when the game begins in the TurnPlayerForm. The TurnPlayerForm will then displayed the information gathered from the players in The StartGameForm. The players can then Fight each other and each turn will be displayed in the Battle Log where most of the information will be displayed during the duration of the game and will also display who the winner of the game is.

### What each form does:

StartGameForm : This is where the player input will be entered and saved

TurnPlayerForm : This is where the game starts, and is played till one of the players are defeated

### What users will provide as input:

The players will enter their name and a name for their dragon. Then the players get to select the dragon type they want (Fire dragon, Ice dragon, Wind dragon or Earth dragon)

### Output generated on form:

The input gathered from the users will be used in the TurnPlayerForm where all input will be displayed. Both players names, dragon names and dragon types but HP will be added which is added to the dragons in the StartGameForm.

### Why output would be generated:

Output would be generated so that players can see whether its their turn or not. It will also help the players know how much HP their dragons have left.

## Descriptive Response

The game starts off with the “StartPlayerForm” which asks the player for input that will be used in the TurnPlayer form. There is a panel where Player One can fill in their information and right next to that panel there is another one that asks the Second player for their information.

The form asks the player to enter their name and a name for their dragon, they can then then proceed by choosing which dragon type they want to use (Four different types of dragons: The Fire dragon, Ice dragon, Wind dragon and the Earth dragon) with each dragon having its own unique abilities where certain dragons have more HP, more attack power, better defence and a good special attack than some of the other dragons so you have to choose carefully.

The dragon’s abilities and HP is added to the dragons in this form (StartGameForm) and gets transferred over to the TurnPlayerForm once the player presses the “start game” button. All the dragons are displayed on the right side of the form with all of their statistics included (HP, Attack power, Special Attack power and then their Block power) so that the players can see what the dragons statistics are and how the dragons look if that helps to finalize their decision.

Once the player is happy with the dragon they chose they can press on the save button and the Second player may then proceed to do the same. Entering their name and a name for their dragon, selecting a dragon type and then they can save when they’re happy.

Once both players are ready and have saved the player can press on the “Start Game” button which will take the player to the next form (TurnPlayerForm) where the game(combat) will begin.

The TurnPlayer form then takes the players input and displays what the players have entered in the StartGameForm so the first player’s dragon name, dragon type and dragon HP will be displayed in a panel to the left.

The first player will have options to make the first move: Attack, Special Attack or Block. The second player will be displayed on a panel to the right of player one (in the middle of the form) where their name, dragon name, dragon type and dragon HP will be displayed. To the right side of the form there will be pictures showing the dragons which the players chose with the first player’s dragon on top and second player’s dragon on the bottom.

With each turn the players will swap places where player one will move to player two’s panel and player two will get the option to Attack, Special Attack or Block and after each turn the player move will be displayed at the bottom of the form in the “Battle log” with the attack taking a certain amount of HP off of the other players dragon, which then changes the amount displayed on the panels where the player details are displayed.

The battle log will describe everything that happens like when a player attacks it will display a message saying that “(Player One’s dragon type + name) attacked (Player Two’s dragon type + name) for x amount of damage”. The same message will be displayed for the special attack but the damage will just be increased and then for the block message it will display “(Player One’s dragon type + name) is blocking x amount of damage for the next turn” so when player two attacks then a certain amount of damage will be blocked.

The battle log will also display who the winner is when the game is over by displaying “(Player One’s dragon type + name) has defeated (player Two’s dragon type + name)”

// StartGameForm

start

Declarations

TextBox txtPlayerName1 = new TextBox()

TextBox txtDragonName1 = new TextBox()

RadioButton rbtnFireDragon1 = new RadioButton()

RadioButton rbtnIceDragon1 = new RadioButton()

RadioButton rbtnWindDragon1 = new RadioButton()

RadioButton rbtnEarthDragon1 = new RadioButton()

Button btnSave1 = new Button()

TextBox txtPlayerName2 = new TextBox()

TextBox txtDragonName2 = new TextBox()

RadioButton rbtnFireDragon2 = new RadioButton()

RadioButton rbtnIceDragon2 = new RadioButton()

RadioButton rbtnWindDragon2 = new RadioButton()

RadioButton rbtnEarthDragon2 = new RadioButton()

Button btnSave2 = new Button()

Button btnStartGame = new Button()

TurnPlayerForm turnPlayer

string p1Data[3]

string p2Data[3]

num p1Values[4]

num p2Values[4]

string FIRE\_DRAG\_NAME = "Fire Dragon"

num FIRE\_DRAG\_HP = 20

num FIRE\_DRAG\_ATK = 5

num FIRE\_DRAG\_SPATK = 12

num FIRE\_DRAG\_BLOCK = 4

string ICE\_DRAG\_NAME = "Ice Dragon"

num ICE\_DRAG\_HP = 30

num ICE\_DRAG\_ATK = 4

num ICE\_DRAG\_SPATK = 9

num ICE\_DRAG\_BLOCK = 5

string WIND\_DRAG\_NAME = "Wind Dragon"

num WIND\_DRAG\_HP = 40

num WIND\_DRAG\_ATK = 3

num WIND\_DRAG\_SPATK = 7

num WIND\_DRAG\_BLOCK = 5

string EARTH\_DRAG\_NAME = "Earth Dragon"

num EARTH\_DRAG\_HP = 50

num EARTH\_DRAG\_ATK = 2

num EARTH\_DRAG\_SPATK = 5

num EARTH\_DRAG\_BLOCK = 6

boolean player1Saved = false

boolean player2Saved = false

btnSave1.registerListener(btnSave1Click())

btnSave2.registerListener(btnSave2Click())

btnStartGame.registerListener(btnStartGameClick())

stop

//Module for “save” button 1

private void btnSave1Click()

Declarations

num PLAYER\_NUMBER = 1

string playerName1

string dragonName1

string type

num hp

num atk

num spatk

num block

playerName1 = txtPlayerName1.getText()

dragonName1 = txtDragonName1.getText()

if playerName1 <> "" AND dragonName1 <> "" then

if rbtnFireDragon1.getCheckedStatus() then

type = FIRE\_DRAG\_NAME

hp = FIRE\_DRAG\_HP

atk = FIRE\_DRAG\_ATK

spatk = FIRE\_DRAG\_SPATK

block = FIRE\_DRAG\_BLOCK

endif

if rbtnIceDragon1.getCheckedStatus() then

type = ICE\_DRAG\_NAME

hp = ICE\_DRAG\_HP

atk = ICE\_DRAG\_ATK

spatk = ICE\_DRAG\_SPATK

block = ICE\_DRAG\_BLOCK

endif

if rbtnWindDragon1.getCheckedStatus() then

type = WIND\_DRAG\_NAME

hp = WIND\_DRAG\_HP

atk = WIND\_DRAG\_ATK

spatk = WIND\_DRAG\_SPATK

block = WIND\_DRAG\_BLOCK

endif

if rbtnEarthDragon1.getCheckedStatus() then

type = EARTH\_DRAG\_NAME

hp = EARTH\_DRAG\_HP

atk = EARTH\_DRAG\_ATK

spatk = EARTH\_DRAG\_SPATK

block = EARTH\_DRAG\_BLOCK

endif

saveValues(PLAYER\_NUMBER, playerName1, dragonName1, type, hp, atk, spatk, block)

player1Saved = true

endif

return

//Module for “save” button 2

private void btnSave2Click()

Declarations

num PLAYER\_NUMBER = 2

string playerName2

string dragonName2

string type

num hp

num atk

num spatk

num block

playerName2 = txtPlayerName2.getText()

dragonName2 = txtDragonName2.getText()

if playerName2 <> "" AND dragonName2 <> "" then

if rbtnFireDragon2.getCheckedStatus() then

type = FIRE\_DRAG\_NAME

hp = FIRE\_DRAG\_HP

atk = FIRE\_DRAG\_ATK

spatk = FIRE\_DRAG\_SPATK

block = FIRE\_DRAG\_BLOCK

endif

if rbtnIceDragon2.getCheckedStatus() then

type = ICE\_DRAG\_NAME

hp = ICE\_DRAG\_HP

atk = ICE\_DRAG\_ATK

spatk = ICE\_DRAG\_SPATK

block = ICE\_DRAG\_BLOCK

endif

if rbtnWindDragon2.getCheckedStatus() then

type = WIND\_DRAG\_NAME

hp = WIND\_DRAG\_HP

atk = WIND\_DRAG\_ATK

spatk = WIND\_DRAG\_SPATK

block = WIND\_DRAG\_BLOCK

endif

if rbtnEarthDragon2.getCheckedStatus() then

type = EARTH\_DRAG\_NAME

hp = EARTH\_DRAG\_HP

atk = EARTH\_DRAG\_ATK

spatk = EARTH\_DRAG\_SPATK

block = EARTH\_DRAG\_BLOCK

endif

saveValues(PLAYER\_NUMBER, playerName2, dragonName2, type, hp, atk, spatk, block)

player2Saved = true

endif

return

//Module for “Start Game” button

private void btnStartGameClick()

if player1Saved AND player2Saved then

turnPlayer = new TurnPlayerForm()

turnPlayer.saveValues(p1Data, p2Data, p1Values, p2Values)

turnPlayer.show()

this.hide()

endif

return

private void saveValues(num playerNum, string playerName, string dragonName, string dragType, num dragHp, num dragAtk, num dragSpatk, num dragBlock)

if playerNum == 1 then

p1Data[0] = playerName

p1Data[1] = dragonName

p1Data[2] = dragType

p1Values[0] = dragHp

p1Values[1] = dragAtk

p1Values[2] = dragSpatk

p1Values[3] = dragBlock

else

p2Data[0] = playerName

p2Data[1] = dragonName

p2Data[2] = dragType

p2Values[0] = dragHp

p2Values[1] = dragAtk

p2Values[2] = dragSpatk

p2Values[3] = dragBlock

endif

return

// TurnPlayerForm

start

Declarations

Button btnAttack = new Button()

Button btnSpAttack = new Button()

Button btnBlock = new Button()

Button btnRest = new Button()

Label lblBattle = new Label()

Label lblDragonNameType1 = new Label()

Label lblHp1 = new Label()

Label lblOpponent = new Label()

Label lblDragonNameType2 = new Label()

Label lblHp2 = new Label()

string p1Data[]

string p2Data[]

num p1Values[]

num p2Values[]

num playerTurn

boolean p1IsResting = false

boolean p2IsResting = false

boolean p1IsBlocking = false

boolean p2IsBlocking = false

boolean p1HasPlayed = false

boolean p2HasPlayed = false

btnAttack.registerListener(btnAttackClick())

btnSpAttack.registerListener(btnSpAttackClick())

btnBlock.registerListener(btnBlockClick())

btnRest.registerListener(btnRestClick())

btnRest.setVisibile(false)

playerTurn = takeInitiative()

switchPlayer()

stop

//Module for “Attack” button

private void btnAttackClick()

Declarations

num hp

num atk

num block

num damage

string attacker

string defender

string blockMessage

string battleText = lblBattle.getText()

if playerTurn == 1 then

hp = p2Values[0]

atk = p1Values[1]

attacker = p1Data[1]

defender = p2Data[1]

if p2IsBlocking then

block = p2Values[3]

damage = atk - block

if damage < 0 then

damage = 0

endif

blockingMessage = " blocks it and"

else

damage = atk

blockingMessage = ""

endif

hp = hp - damage

if hp < 0 then

hp = 0

endif

p2Values[0] = hp

p1HasPlayed = true

else

hp = p1Values[0]

atk = p2Values[1]

attacker = p2Data[1]

defender = p1Data[1]

if p1IsBlocking then

block = p1Values[3]

damage = atk - block

if damage < 0 then

damage = 0

endif

blockingMessage = " blocks it and"

else

damage = atk

blockingMessage = ""

endif

hp = hp - damage

if hp < 0 then

hp = 0

endif

p1Values[0] = hp

p2HasPlayed = true

endif

battleText = battleText, attacker, " attacks ", defender, "! ", defender, blockMessage, " takes ", damage, " damage. ", defender " is now on ", hp, " HP\n"

battleText = battleText, "--------------------------------------------------------------------------\n"

lblBattle.setText(battleText)

switchPlayer()

return

//Module for “Special Attack” button

private void btnSpAttackClick()

Declarations

num hp

num spatk

num block

num damage

string attacker

string defender

string blockMessage

string battleText = lblBattle.getText()

if playerTurn == 1 then

hp = p2Values[0]

spatk = p1Values[2]

attacker = p1Data[1]

defender = p2Data[1]

if p2IsBlocking then

block = p2Values[3]

damage = spatk - block

if damage < 0 then

damage = 0

endif

blockingMessage = " blocks it and"

else

damage = spatk

blockingMessage = ""

endif

hp = hp - damage

if hp < 0 then

hp = 0

endif

p2Values[0] = hp

p1IsResting = true

p1HasPlayed = true

else

hp = p1Values[0]

spatk = p2Values[2]

attacker = p2Data[1]

defender = p1Data[1]

if p1IsBlocking then

block = p1Values[3]

damage = spatk - block

if damage < 0 then

damage = 0

endif

blockingMessage = " blocks it and"

else

damage = spatk

blockingMessage = ""

endif

hp = hp - damage

if hp < 0 then

hp = 0

endif

p1Values[0] = hp

p2IsResting = true

p2HasPlayed = true

endif

battleText = battleText, attacker, " special attacks ", defender, "! ", defender, blockMessage, " takes ", damage, " damage. ", defender " is now on ", hp, " HP\n"

battleText = battleText, "--------------------------------------------------------------------------\n"

lblBattle.setText(battleText)

switchPlayer()

return

//Module for “Block” button

private void btnBlockClick()

Declarations

string currentDragon

string nextDragon

string battleText = lblBattle.getText()

if playerTurn == 1 then

currentDragon = p1Data[1]

nextDragon = p2Data[1]

p1IsBlocking = true

p1HasPlayed = true

else

currentDragon = p2Data[1]

nextDragon = p1Data[1]

p2IsBlocking = true

p2HasPlayed = true

endif

battleText = battleText, currentDragon, " prepares to block ", nextDragon, "'s next attack\n"

battleText = battleText, "--------------------------------------------------------------------------\n"

lblBattle.setText(battleText)

switchPlayer()

return

private void btnRestClick()

rest(playerTurn)

return

private num takeInitiative()

Declarations

num p1Roll

num p2Roll

num startPlayer

p1Roll = randomRoll()

p2Roll = randomRoll()

while p1Roll == p2Roll

p1Roll = randomRoll()

p2Roll = randomRoll()

endwhile

if p1Roll > p2Roll then

// PlayerTurns are inverted so that when switchPlayer() Module is called after takeInitiative() Module then it will switch to the correct player’s turn

playerTurn = 2

else

playerTurn = 1

endif

return startPlayer

private num randomRoll()

Declarations

num roll

num MIN = 1

num MAX = 6

Random random = new Random()

roll = random.next(MIN, MAX)

return roll

public void saveValues(string p1Data[], string p2Data[], num p1Values[], num p2Values[])

this.p1Data = p1Data

this.p2Data = p2Data

this.p1Values = p1Values

this.p2Values = p2Values

return

//Module for switching players after each turn

private void switchPlayer()

Declarations

string player

string dragon

string type

num hp

string opponent

string opponentDragon

string opponentType

num opponentHp

string battleText = lblBattle.getText()

if p1HasPlayed AND p2HasPlayed then // Begin new round

p1HasPlayed = false

p2HasPlayed = false

takeInitiative()

switchPlayer()

// Return is used early since the battle log was updated through the previous switchPlayer() call

return

endif

if playerTurn == 1 then

playerTurn = 2

player = p2Data[0]

dragon = p2Data[1]

type = p2Data[2]

hp = p2Values[0]

opponent = p1Data[0]

opponentDragon = p1Data[1]

opponentType = p1Data[2]

opponentHp = p1Values[0]

if p2IsResting then

btnAttack.setVisibile(false)

btnSpAttack.setVisibile(false)

btnBlock.setVisibile(false)

btnRest.setVisibile(true)

endif

if p2IsBlocking then

p2IsBlocking = false

endif

else

playerTurn = 1

player = p1Data[0]

dragon = p1Data[1]

type = p1Data[2]

hp = p1Values[0]

opponent = p2Data[0]

opponentDragon = p2Data[1]

opponentType = p2Data[2]

opponentHp = p2Values[0]

if p1IsResting then

btnAttack.setVisibile(false)

btnSpAttack.setVisibile(false)

btnBlock.setVisibile(false)

btnRest.setVisibile(true)

endif

if p1IsBlocking then

p1IsBlocking = false

endif

endif

if NOT(p1HasPlayed OR p2HasPlayed) then // Start of new round

battleText = battleText, \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

battleText = battleText, player, "'s dragon ", dragon, " takes inititative!\n"

battleText = battleText, "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

endif

battleText = battleText, player, "'s Turn:\n"

if hp == 0 then

btnAttack.setEnabled(false)

btnSpAttack.setEnabled(false)

btnBlock.setEnabled(false)

btnRest.setEnabled(false)

battleText = battleText, dragon, " is unable to continue. ", opponentDragon, " is the winner!"

endif

lblBattle.setText(battleText)

lblDragonNameType1.setText(dragon, ", the ", type, "'s Turn")

lblHp1.setText("HP: ", hp)

lblOpponent.setText("Opponent: ", opponent)

lblDragonNameType2.setText(opponentDragon, ", the ", opponentType)

lblHp2.setText("HP: ", opponentHp)

return

//Module for setting dragon’s tiredness level

private void rest(num dragonNum)

Declarations

string dragon

string battleText = lblBattle.getText()

if dragonNum = 1 then

dragon = p1Data[1]

p1IsResting = false

p1HasPlayed = true

else

dragon = p2Data[1]

p2IsResting = false

p2HasPlayed = true

endif

battleText = battleText, dragon, " is to tired tired to fight, and rests a while\n"

battleText = battleText, "--------------------------------------------------------------------------\n"

lblBattle.setText(battleText)

btnAttack.setVisibile(true)

btnSpAttack.setVisibile(true)

btnBlock.setVisibile(true)

btnRest.setVisibile(false)

switchPlayer()

return