

**Let's get rich in Chula Documentation**

**Created by**

Taeshin Srisomsak 6230198021

Patcharapong Khempan 6230362521

**2110215 Programming Methodology**

**Semester 2 Year 2020**

**Chulalongkorn University**

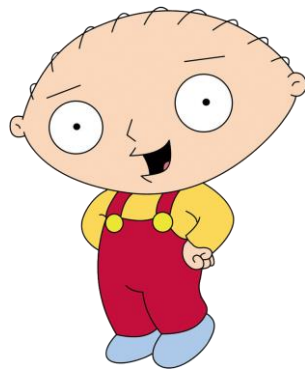
# Let's Get Rich in Chula

## Introduction

Let's Get Rich in Chula is a game based on Monopoly. Monopoly is a game where each player has to manage their money by investing in places. If a player loses all money, they will get bankrupt and lose the game.

## Gameplay and Rules

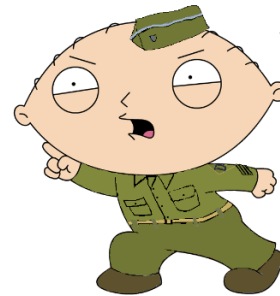
- There are two players.



Player 1 Character



Player 1 Color






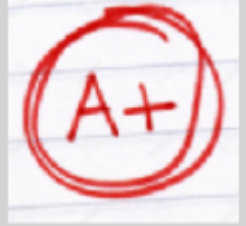



Player 2 Character



Player 2 Color

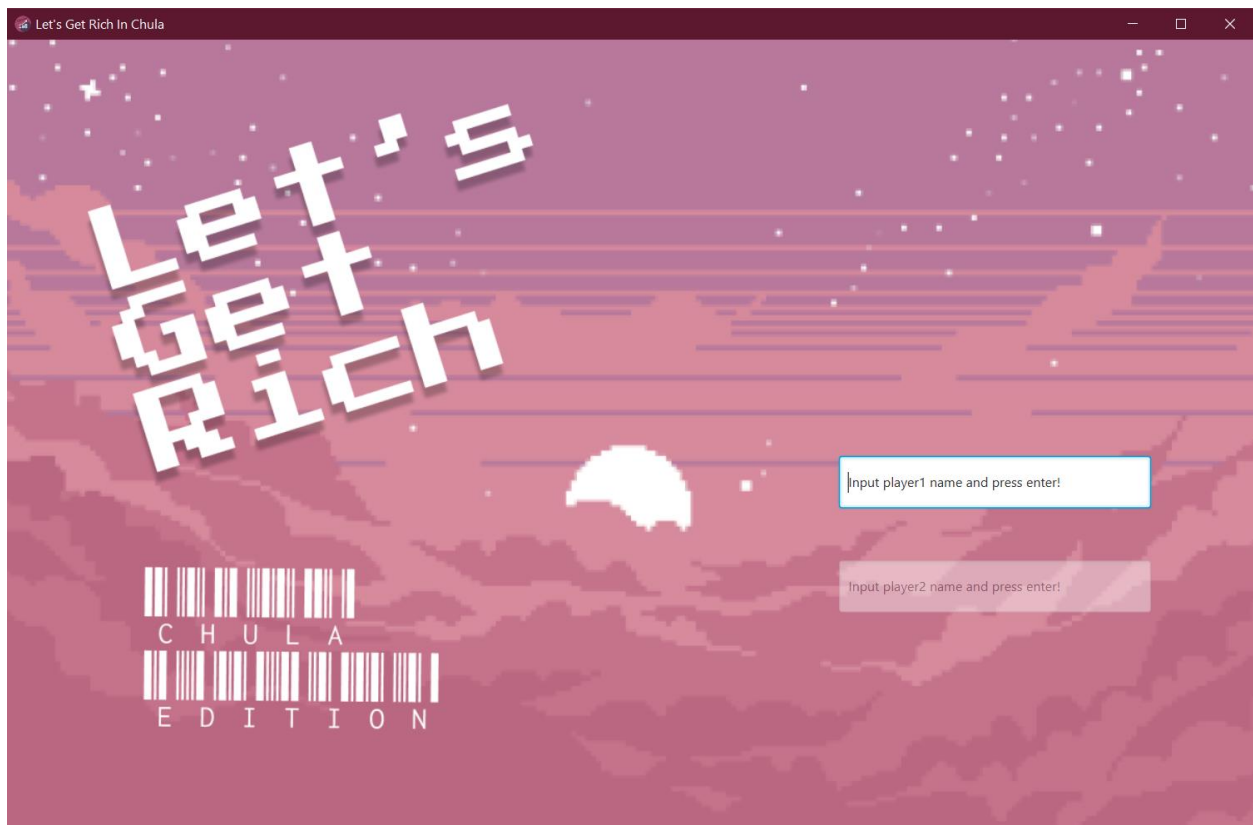
- Each player has given a starting money of 100,000
- Each turn a player gets to roll a dice and be moved to the corresponding position (tile). Each tile has its corresponding action listed by the following table :

	<p>Normal location tile – White Background</p> <p>A player can buy and the other can rebuy if they want.</p>
	<p>Island location tile – Yellow Background</p> <p>A player can only buy once, other player cannot rebuy.</p>
	<p>Jail Tile – LarnGear icon</p> <p>A player gets distracted in LarnGear, loses one turn for free.</p>
	<p>Item tile</p> <p>You will get one random card</p> <ol style="list-style-type: none"> <li>1. GetMoneyCard (Money +20000)</li> <li>2. LoseMoneyCard (Money -20000, if Player's Money less than 20000, gets reset to 0)</li> <li>3. ProtectionCard (Skip a turn where u don't have enough money)</li> </ol>
	<p>Drunk tile</p> <p>You are now drunk, 50% of your current money will be gone.</p>
	<p>"A" tile</p> <p>You get an A in ProgMeth. As a bonus, Aj.Toe gives you 50% more of your current money</p>

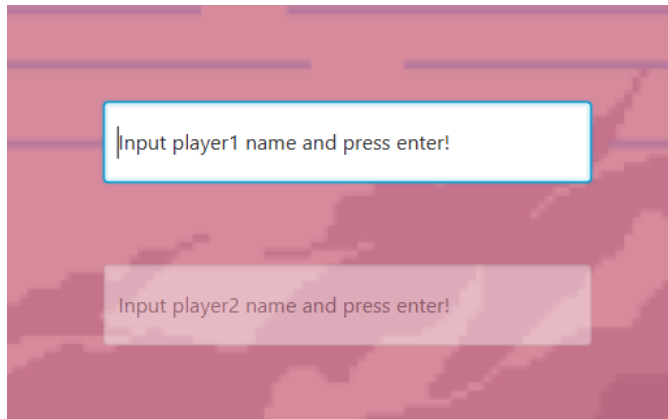
	<p>Start tile</p> <p>If you landed or passed here, you will get 20,000 money. But by landing here you can't do anything.</p>
-----------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------

- There are 30 turns to play, a player can win by either
  1. having overall value (their assets + their current money) more than the opponent.
  2. The opponent has no money and lands on your property.

## GUI of the Game



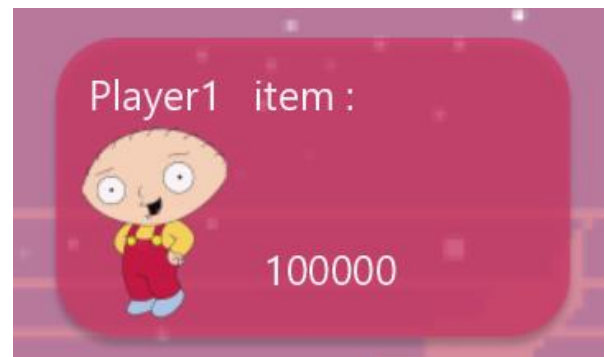
*Main Screen*

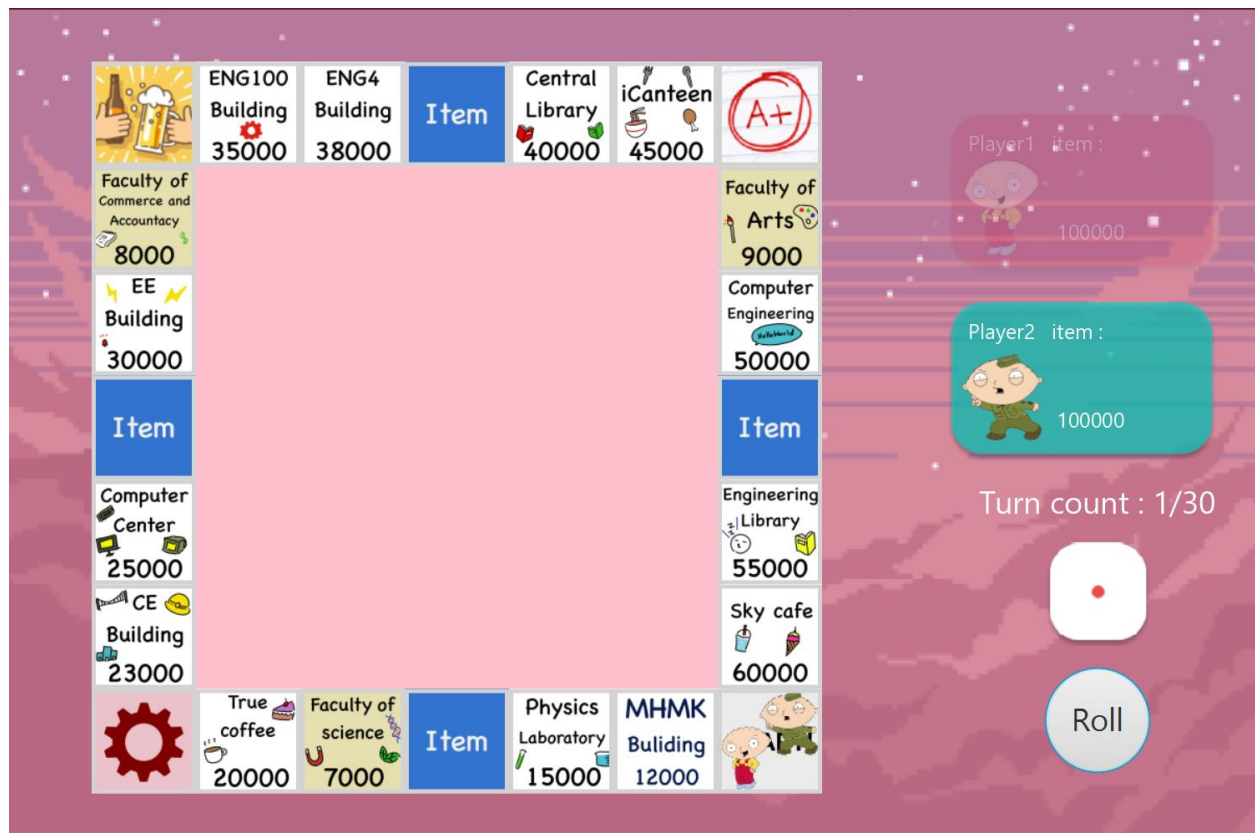


Main screen – You get to type Player1 and Player2's names in here.

You can use your mouse to click on the label to clear the text.

If a player hasn't typed their name in, or decided to leave it blank, the game will automatically assign the name "Player1" and "Player2" respectively.

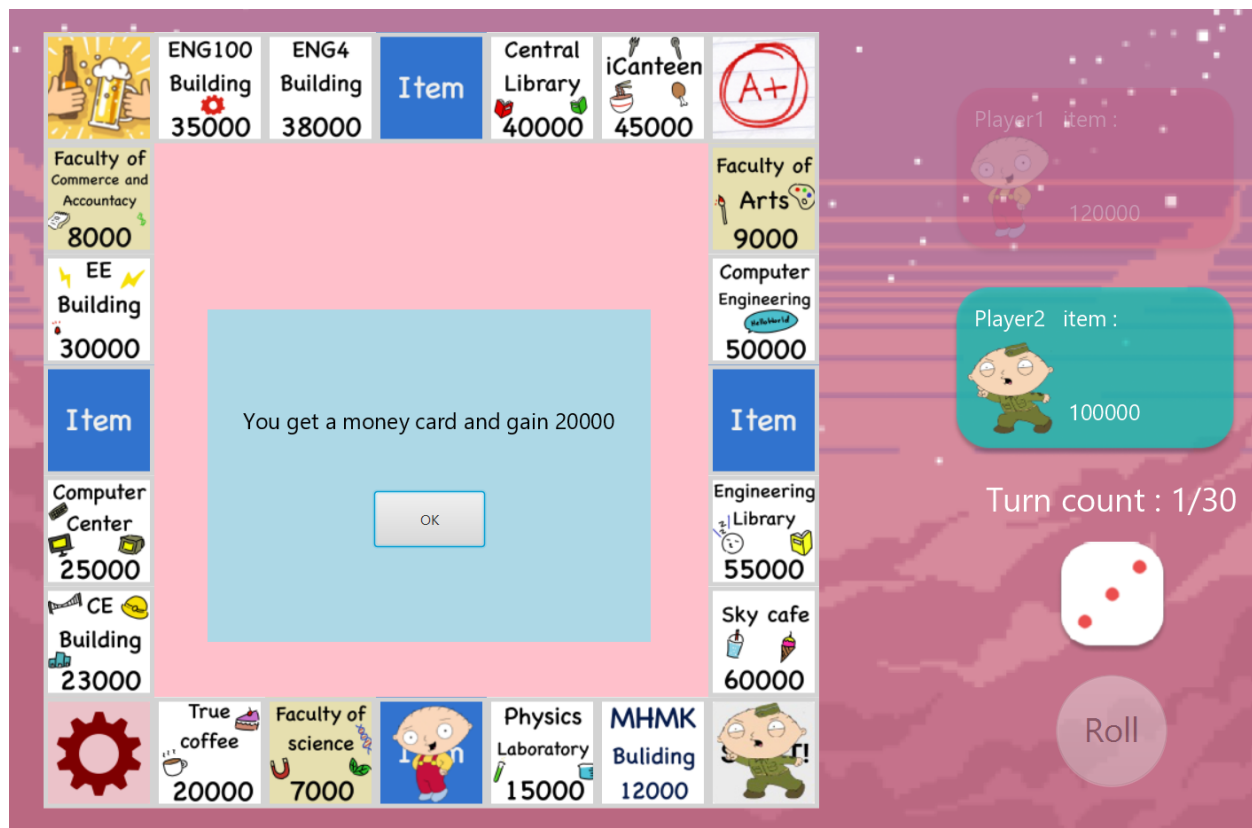




*Gameplay Scene*

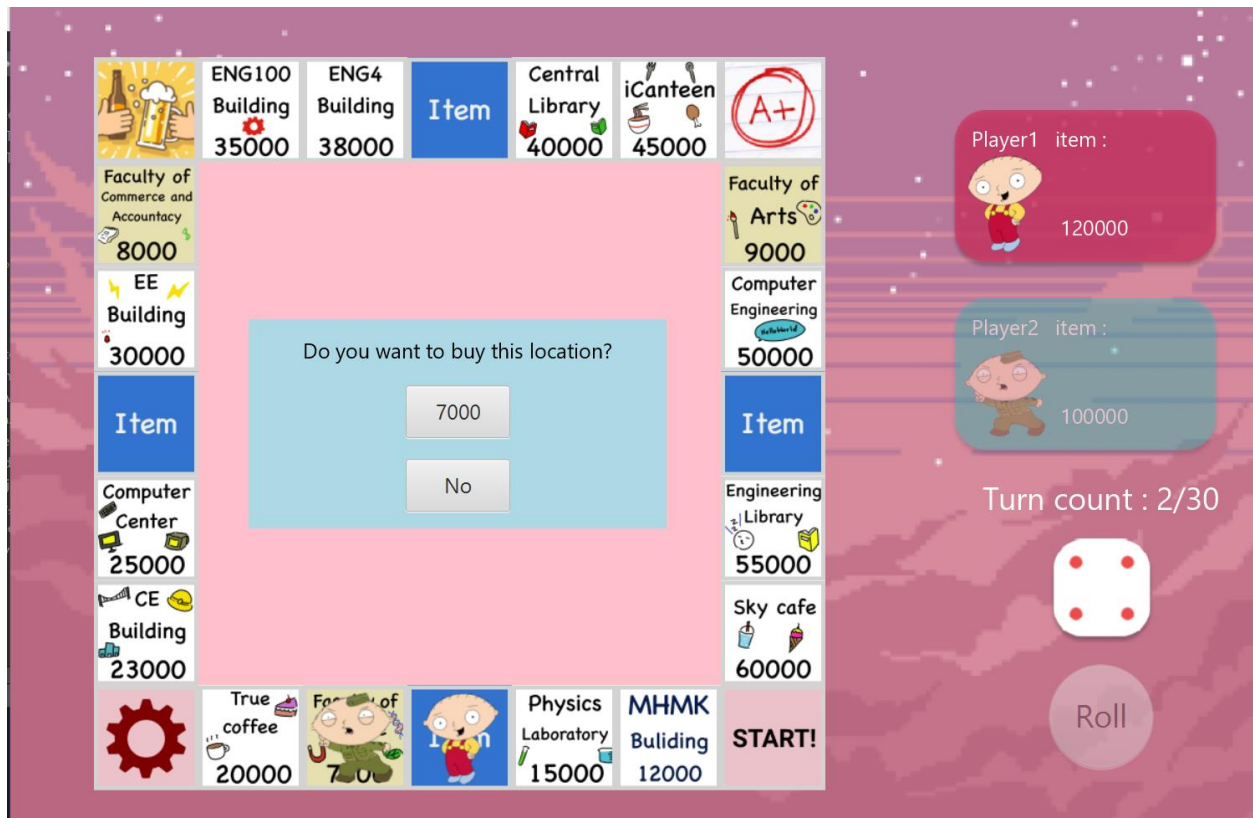
## Gameplay Scene

The blinking player menu indicates who will be playing in this current turn. A player must click “Roll” to proceed.



Since Player1 has rolled a 3, he moves 3 position from start, making him landed on ItemTile. On ItemTile, he is lucky to get a money card item.

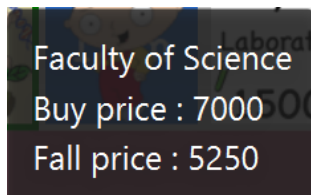
After he clicks OK, Player2 gets to play next and the turn count increases.



Now Player2 has rolled and got a 4, he moved 4 positions to an IslandLocation (No one else can retake this place once taken). He now has two options to choose, either buy it (spends 7000 money) or just decline it.



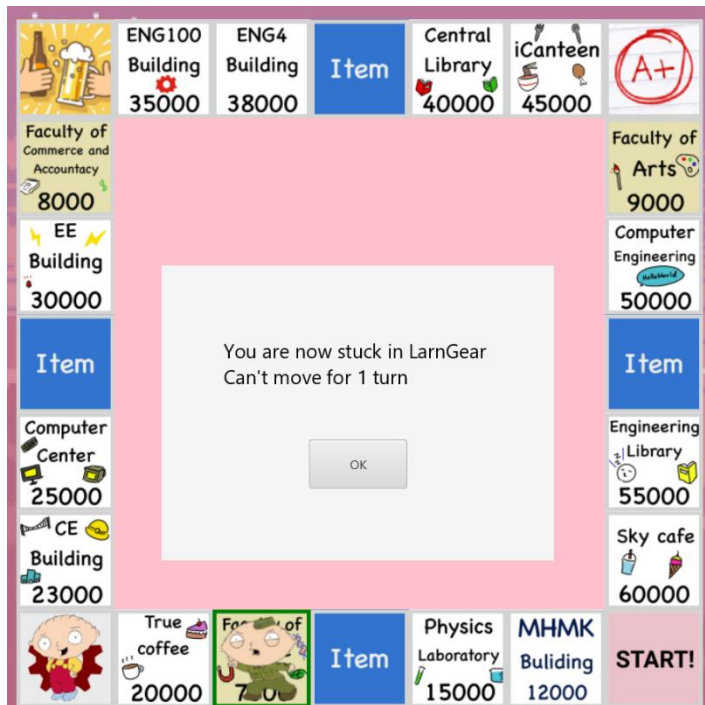
This is what happens when he buys it, the location has the green border around indicating that this location belongs to Player2.



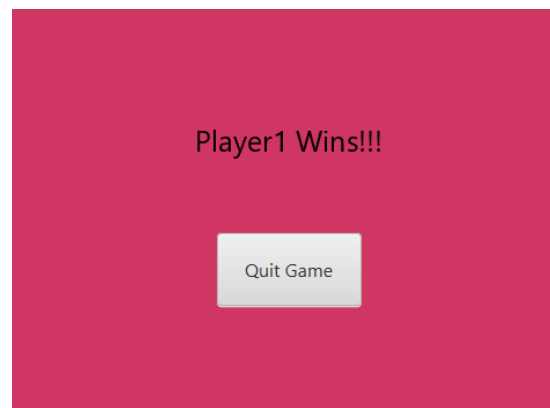
Now if Player1 lands on this, he will lose money. You can see the money stats by hovering over each location.

Now the turn gets increased, Player1 gets to play again.





Oh no, Player1 rolled a 3 and go straight into LarnGear, he now stuck in there for one turn.



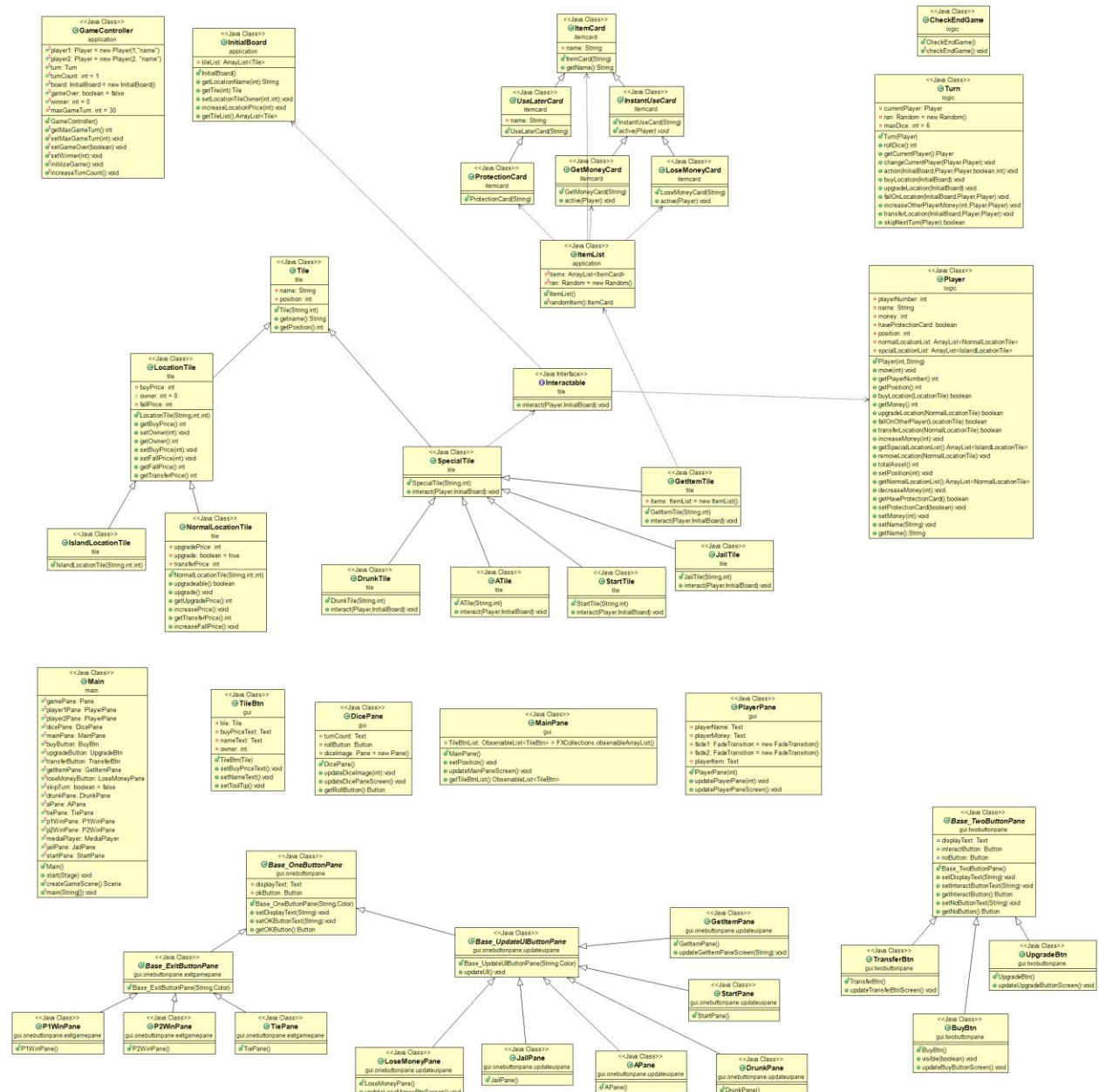
The game continues for 30 turns if no one gets bankrupt. Each player will get their money and their property value calculated. A player with more value wins.

Or if someone bankrupts, the other one automatically wins as well.

## 1.Package tile

## Fields

- String name	Name of each tile
- position	Position of the tile



### Constructor

+ Tile(String name,int Position)	- Initialize tile name - Initialize tile position
----------------------------------	------------------------------------------------------

### Methods

+ String getname()	Return name
+ int getPosition()	Return position

## 1.2 Class LocationTile extends Tile

### Fields

- int buyPrice	Price to buy this tile
- int owner	Owner of this tile 0 = no owner 1 = owner is player1 2 = owner is player 2
- int fallPrice	Price if other player fall on this tile

### Constructor

+ LocationTile(String name,int buyprice,int position)	- super(name,position) form Tile class - Initilize buyPrice - Initialize fallPrice = $0.75 * \text{buyPrice}$
-------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------

### Methods

+ int getBuyPrice()	return buyPrice
+ void setOwner(int playerNumber)	Set owner of this tile by number 1 = player1 2= player2
+ int getOwner()	Return number of owner
+ void setBuyPrice(int buyPrice)	Set buyPrice
+ void setFallPrice(int newfallprice)	Set fallPrice
+ int getFallPrice()	Get fallPrice
+ int getTransferPrice()	Get transferPrice

### 1.3 Class NormalLocationTile extends LocationTile

fields

- int upgradePrice	Price to upgrade this location
- Boolean upgrade	Show that this location can be upgrade or not
- int transferPrice	Price to transterfer this location to other player

Constructor

+ NormalLocationTile(Sting name,int buyprice,int position)	-super(name,buyprice,position) From LocationTile -Initialize transferPrice to 1.5*buyprice -Initialize upgradePrice to 0.5*buyprice
---------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------

### Methods

+ Boolean upgradeable	Return upgrade
+ void upgrade()	Set upgrade to false
+ int getUpgradePrice()	Get upgradePrice
+ void increasePrice()	-Increase upgradePrice = $0.5 * \text{buyPrice}$ -Increase transferPrice = $0.5 * \text{buyPrice}$ -Increase fallPrice = $0.5 * \text{buyPrice}$
+ int getTransferPrice()	Get transferPrice
+ void increaseFallPrice()	Double current fallPrice

## 1.4 Class IslandLocationTile extends LocationTile

### Constructor

+ IslandLocationTile(String name,int buyprice,int position)	Super(name,buyprice,position) From LocationTile
-------------------------------------------------------------	----------------------------------------------------

## 1.5 Interface Interactable

### Methods

+ void interact(Player player,InitialBoard board)	Interact when player step on tile that is interactable
---------------------------------------------------	--------------------------------------------------------

## 1.6 Class SpecialTile extends Tile implements Interactable

### Constructor

+ SpecialTile(String name,int position)	Super(name,position) from Tile
-----------------------------------------	--------------------------------

#### Methods

+ void interact(Player player,InitialBoard board)	Interact when player step on tile that is interactable
---------------------------------------------------	--------------------------------------------------------

### 1.7 Class StartTile extends SpecialTile

#### Constructor

+ StartTile(String name,int position)	Super(name,position) from SpatialTile
---------------------------------------	---------------------------------------

#### Methods

+ void interact(Player player,InitialBoard board)	-Set StartPane to be visible
---------------------------------------------------	------------------------------

### 1.8 Class JailTile extends SpecialTile

#### Constructor

+ JailTile(String name,int position)	Super(name,position) from SpecialTile
--------------------------------------	---------------------------------------

#### Methods

+ void interact(Player player,InitialBoard board)	-Set jailPane to be visible
---------------------------------------------------	-----------------------------

## 1.9 DrunkTile extends SpecialTile

### Constructor

+ DrunkTile(String name,int position)	Super(name,position) from SpecialTile
---------------------------------------	---------------------------------------

### Methods

+ void interact(Player player,InitialBoard board)	-Set drunkPane to be visible -Decrease current player money by 50%
---------------------------------------------------	-----------------------------------------------------------------------

## 1.10 ATile extends SpecialTile

### Constructor

+ ATile(String name,int position)	Super(name,position) from SpecialTile
-----------------------------------	---------------------------------------

### Methods

+ void interact(Player player,InitialBoard board)	-Set aPane to be visible
---------------------------------------------------	--------------------------

## 1.11 GetItemTile extend SpecialTile

### Fields

- ItemList items	List of item cards
------------------	--------------------

### Constructor

+ GetitemTile(String name,int position)	Super(name,position) from SpecialTile
-----------------------------------------	---------------------------------------

### Methods

+ void interact(Player player,InitialBoard board)	- randomly give player one of three item card If card is getmoney and losemoney card active card ability But if it is protection set Player Protectioncard to true
---------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2. Package itemcard

### 2.1 Class ItemCard

#### Fields

- String name	Name of item card
---------------	-------------------

#### Constructor

+ ItemCard(String name)	Initialize name of item card
-------------------------	------------------------------

#### Methods

+ String getName()	Get name of item card
--------------------	-----------------------

### 2.2 Abstract Class InstantUseCard extends ItemCard



### Constructor

+ InstantUseCard(String name)	Super(name) from ItemCard
-------------------------------	---------------------------

### Methods

+ void active( <i>Player player</i> )	
---------------------------------------	--

## 2.3 Abstract Class UseLaterCard extends ItemCard

### Constructor

+ UseLaterCard(String name)	Super(name) from ItemCard
-----------------------------	---------------------------

## 2.4 Class GetMoneyCard extends InstantUseCard

### Constructor

+ GetMoneyCard(String name)	Super(name) from ItemCard
-----------------------------	---------------------------

### Methods

+ void active( <i>Player player</i> )	-Increase player money by 20000 -Update getItemPane Screen -Update player 1 and Player 2 Pane -Visible getItemPane
---------------------------------------	--------------------------------------------------------------------------------------------------------------------------------

## 2.5 Class LoseMoneyCard extends InstantUseCard

### Constructor

+ LoseMoneyCard(String name)	Super(name) from ItemCard
------------------------------	---------------------------

### Methods

+ void Active(Player player)	-Decrease player money by 20000 If player have less than 2000 set money to 0 - Update getItemPane Screen -Update player 1 and Player 2 Pane -Visible getItemPane
------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2.6 Class ProtectionCard extends UseLaterCard

### Constructor

+ ProtectionCard(String name)	Super(name) from ItemCard
-------------------------------	---------------------------

## 3.Package application

### 3.1 Class GameController

#### Fields

+ <u>Player player1</u>	Player 1 Initialize player number to 1 and name to "name"
+ <u>Player player2</u>	Player2

	Initialize player number to 2 and name to "name"
<u>+ Turn turn</u>	Turn to control things to do in 1 turn
<u>+ int turnCount</u>	Number of turn ,Initialize to 1
<u>+ int maxGameTurn</u>	Number of maximum turn
<u>+ InitialBoard board</u>	Board of the game that contain list of location,Initial new board
<u>+ Boolean gameOver</u>	Game over status,Initialize to false
<u>+ int winner</u>	Number of player that win the game,Initialize to 0

### Methods

<u>+ void setGameOver(Boolean status)</u>	Set gameOver
<u>+ void setWinner(int player)</u>	Set winner
<u>+ void initializeGame()</u>	Initialize turn to start with player1
<u>+ void increaseTurnCount()</u>	Increase turnCount by 1
<u>+ int getMaxGameTurn()</u>	Get number of maximum turn
<u>+ void setMaxGameTurn(int maxGameTurn)</u>	Set number of maximum turn

## 3.2 Class InitialBoard

### Fields

- ArrayList<Tile>tileList	ArrayList that contain tile in the board
---------------------------	------------------------------------------

### Constructor

+ initialBoard()	-Initialize tileList -add every tile on the board to tileList
------------------	------------------------------------------------------------------

### Methods

+ String getLocationName(int index)	Get Tile name in tileList from input index
+ Tile getTile(int index)	Get Tile in tileList from input index
+ void setLocationTileOwner(int index,int playerNumber)	Set owner of locationTille in Tile list to match input number
+ void increaseLocationPrice(int index)	Call method increasePrice on selected NormalLocationTile
+ ArrayList<Tile> getTileList()	Get tileList

## 3.3 Class ItemList

### Fields

- <u>ArrayList&lt;ItemCard&gt; items</u>	ArryList that contains ItemCard
- <u>Random ran</u>	Constant Random

### Constructor

+ ItemList()	- Initialize items -add GetMoneyCard,LoseMoneyCard,ProtectionCard to items
--------------	----------------------------------------------------------------------------------

### Methods

+ <u>ItemCard randomItem()</u>	Random item in items and return ItemCard
--------------------------------	------------------------------------------

## 4.Package Logic

### 4.1 Class CheckEndGame

#### Methods

+ <u>void checkEndGame()</u>	<p>Check if the game is end of not</p> <p>1.if turnCount == maxiumTurn (GameController.getMaxGameTurn())</p> <p>    Check totalAsset() of both player if equals set visible tiePane</p> <p>    ,If player1 &gt; player2 set visible p1WinPane ,if player2 &gt; player1 set visible p2WinPane</p> <p>2.if winner != 0</p> <p>    If winner == 1 set visible p1WinPane, if winner == 2 set visible p2WinPane</p>
------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 4.2 Class Player

#### Fields

- int playerNumber	<p>Number of player</p> <p>1 == player1</p> <p>2 == player 2</p>
- String name	Name of player

- int money	Money of player
- Boolean haveProtectionCard	Show that player have ProtectionCard or not
- int position	Position of player
- ArrayList<NormalLocationTile> normalLocationList	List that contains NormalLocationTile that player own
- ArrayList<IslandLocationTile> spacialLocationList	List that contains IslandLocationTile that player own

### Constructor

Player(int playerNumber,String name)	<ul style="list-style-type: none"> <li>- Initialize player name</li> <li>- Initialize playerNumber</li> <li>- Initialize money to 100000</li> <li>- Initialize haveProtectionCard to false</li> <li>- Initialize position to 0</li> <li>- Initialize normalLocationList and spacialLocationList</li> </ul>
--------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Methods

+ void move(int step)	Create int newPosition = (position + step)%24 and set position to newPosition
+ int getPlayerNumber()	Return playerNumber
+ int getPosition()	Return position

+ boolean buyLocation(LocationTile location)	If player money >= location buyPrice decrease player money and add location to normalLocationList if location is NormalLocation and add to spacialLocationList if location is IslandLocation If buy success return true else return false
+ int getMoney	Return money
+ boolean upgradeLocation(NormalLocationTile normalLocation)	If playermoney >= normalLocation upgradePrice decrease player money and return true else return false
+ boolean fallOnOtherplayer(LocationTile location)	If player money >= location fallPrice decrease player money and return true else return false
+ boolean transferLocation(NormalLocationTile normalLocation)	If player money >= normalLocation transferPrice decrease player money,add normalLocation to normalLocationList and return true else return false
+ void increaseMoney(int amount)	Increase player money by amount
+ ArrayList<IsLandLocationTile> getSpacialLocationList()	Return spacialLocationList
+ void removeLocation(NormalLocationTile normalLocationTile)	Remove normalLocationTile from normalLocationList

+ int totalAsset()	Return total asset of player that equal player current money + fallPrice for each location that player own
+ void setPosition(int position)	Set position of player
+ ArrayList<NormalLOcationTile> getNormalLocationList()	Return normalLocationList
+ void decreaseMoney(int amount)	Decrease player money by amount
+ Boolean getHaveProtectionCard	Return haveProtectionCard
+ void setProtectionCard(Boolean b)	Set haveProtectionCard to b
+ void setMoney(int money)	Set player money
+ void setName(String text)	Set player name
+ String getName()	Return name

#### 4.3 Class Turn

##### Fields

- Player currentPlayer	Current player of this turn
- Random ran	Use to random
- int maxDice	Max number on dice

##### Constructor

+ Turn(Player player)	Set currentPlayer to player
-----------------------	-----------------------------

##### Methods

+ int rollDice()	Return Random number that not exceed maxDice and +1
+ Player getCurrentPlayer()	Return currentPlayer

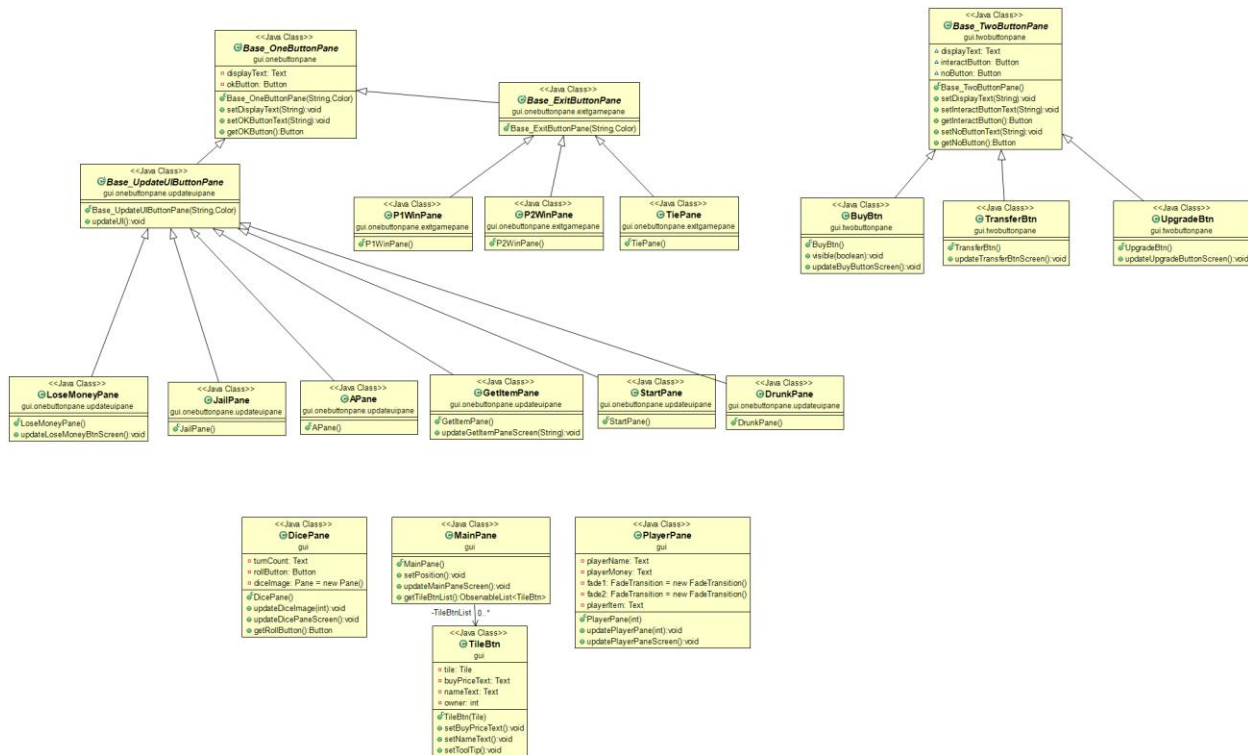


+ void changeCurrentPlayer(Player player1,Player player2)	If currentPlayer number is 1 change currentPlayer to player2 else change currentPlayer to player1
+ void action(InitialBoard board,Player player1,Player player2,Boolean gameOver,int winner)	If position of player on board is LocationTile and this tile owner == 0 call method buyLocation(board) but if location owner == currentPlayerNumber and Location is NormalLocation call method upgradeLocation(board) If Tile owner != currentPlayerNumber call method fallOnLocation(board,player1,player2) And if Tile is NormalLocationTile call method transferLocation(board,player1,player2) If player don't have enough monry call method getHaveProtectionCard to check that player have protectionCrd or not if yes setProtection to false,update playerPane,increase turn and change turn but if player don't have protectionCard setGameOver to true and set winner to opposite player If tile is SpacialTile call method interact(currentPlayer,board) of that Tile Increase turnCount UpdateDicePane and playerPane changeTurn of player

## Methods

+ void buyLocation(InitialBoard board)	updateBuyButtonScreen and set visible of buyButton to true
+ void upgradeLocation(InitialBoard board)	updateUpgradeButtonScreen and set visible of upgradeButton to true
+ void fallOnLocation(InitialBoard,Player player1,Player player2)	updateLoseMoneyButton and setVisible to true Increase other player money
+ void increaaseOtherPlayerMoney(int amount,Player player1,Player player2)	If currentPlayer is player1 increase player2 money else increase player 1 money
+ void transferLocation(InitialBoard board,Player player1,Player player2)	updateTransferbtnScreen and setVisible of transferButton to true
+ Boolean skipNextTurn(Player player)	If player position is 6 return true else return false

## 5. Package gui



UML For Package GUI Only

### 5.1 Package gui.onebuttonpane

#### 5.1.1 Abstract Class Base\_OneButtonPane extends VBox

Base\_OneButtonPane is a pane that has only one button (OK). Every pane that has only one button will inherit properties.

#### Fields

- Text displayText	JavaFX Text
- Button okButton	JavaFX Button

#### Constructor

+ Base_OneButtonPane(String display, Color bgColor)	Initialize a pane that contains displayText and okButton with a background color of bgColor
-----------------------------------------------------	---------------------------------------------------------------------------------------------

## Methods

+ void setDisplayText(String newText)	Set displayText Text
+ void SetOkButtonText(String newText)	Set OKButton Text
+ Button getOKButton()	Get OKButton

### 5.1.2 Package gui.onebuttonpane.exitgamepane

5.1.2.1 Abstract Class Base\_ExitButtonPane extends Base\_OneButtonPane

#### Constructor

+ Base_ExitButtonPane(String display, Color bgColor)	Create a OneButtonPane with a button with text("Quit Game") and set the button onAction to quit the application.
------------------------------------------------------	------------------------------------------------------------------------------------------------------------------

5.1.2.2 Class P1WinPane extends Base\_ExitButtonPane

#### Constructor

+ P1WinPane()	Initialize ExitButtonPane with buttonText = "Player1 Wins!!!" and bgColor = rgb(207,54,100)
---------------	---------------------------------------------------------------------------------------------

5.1.2.3 Class P2WinPane extends Base\_ExitButtonPane

#### Constructor

+ P2WinPane()	Initialize ExitButtonPane with buttonText = "Player2 Wins!!!" and bgColor = rgb(10,205,193)
---------------	---------------------------------------------------------------------------------------------

#### 5.1.2.4 Class TiePane extends Base\_ExitButtonPane

##### Constructor

+ TiePane()	Initialize ExitButtonPane with displayText = "Both player have equal total asset → Tie!!" and bgColor = Color.LIGHTBLUE
-------------	----------------------------------------------------------------------------------------------------------------------------------

#### 5.1.3 Package gui.onebuttonpane.updateuipane

##### 5.1.3.1 Abstract Class Base\_UpdateButtonPane extends Base\_OneButtonPane

##### Constructor

+ Base_UpdateButtonPane (String display, Color bgColor)	Create a OneButtonPane from super class.
------------------------------------------------------------	---------------------------------------------

##### Method

+ void updateUI()	<ul style="list-style-type: none"><li>- Disable roll button</li><li>- Update player1&amp;2 pane</li><li>- Update dicePane</li><li>- Increase turn count</li></ul>
-------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------

##### 5.1.3.2 Class APane extends Base\_UpdateUIButtonPane

##### Constructor

+ APane()	Create an UpdateUIButtonPane with displayText = "You get grade A in Progmeth so Aj.Toe"+"\\n" + "give you 50% of your current money!!!" and bgColor = Color.LIGHTYELLOW
-----------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	Set OKButton onAction to set Main.aPane's visibility to false
--	---------------------------------------------------------------

#### 5.1.3.3 Class DrunkPane extends Base\_UpdateUIButtonPane

##### Constructor

+ DrunkPane ()	<p>Create an UpdateUIButtonPane with displayText = "You are drunk and lose 50% of your money!!!" and bgColor = Color. MEDIUMPURPLE</p> <p>Set OKButton onAction to set Main.drunkPane's visibility to false</p>
----------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### 5.1.3.4 Class GetItemPane extends Base\_UpdateUIButtonPane

##### Constructor

+ GetItemPane()	<p>Create an UpdateUIButtonPane with displayText = "" and bgColor = Color. LIGHTBLUE</p> <p>Set OKButton onAction to set Main.getItemPane's visibility to false and then call updateUI()</p>
-----------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

##### Method

+ void updateGetItem PaneScreen(String text1)	Method for setting displayText to text1
--------------------------------------------------	-----------------------------------------

### 5.1.3.5 Class JailPane extends Base\_UpdateUIButtonPane

#### Constructor

+ JailPane()	Create an UpdateUIButtonPane with displayText = "You are now stuck in LarnGear \n"+ "Can't move for 1 turn " and bgColor = Color. WHITESMOKE  Set OKButton onAction to set Main.jailPane's visibility to false
--------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 5.1.3.6 Class LoseMoneyPane extends Base\_UpdateUIButtonPane

#### Constructor

+ LoseMoneyPane()	Create an UpdateUIButtonPane with displayText = "You lose : " and bgColor = Color. LIGHTBLUE Set OKButton onAction to set Main.loseMoneyButton's visibility to false and updateUI
-------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### Methods

+ updateLoseMoneyBtnScreen ( )	Set displayText to "You lose : " + getFallPrice()
--------------------------------------	------------------------------------------------------

### 5.1.3.7 Class StartPane extends Base\_UpdateUIButtonPane

#### Constructor

+ StartPane()	Create an UpdateUIButtonPane with
---------------	-----------------------------------

	displayText = "You got 20,000 for passing the start!" and bgColor = Color.WHITESMOKE Set OKButton onAction to set Main.startPane's visibility to false and updateUI
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 5.2 Package gui.twobuttonpane

### 5.2.1 Abstract Class Base\_TwoButtonPane extends VBox

Base\_TwoButtonPane is a pane that has two buttons (Interact/NO). Every pane that has two buttons will inherit properties.

#### Fields

- Text displayText	JavaFX Text
- Button interactButton	JavaFX Button
- Button noButton	JavaFX Button

#### Constructor

+ Base_OneButtonPane(String display, Color bgColor)	Initialize a pane that contains displayText and okButton with a background color of Color.LIGHTBLUE
-----------------------------------------------------	-----------------------------------------------------------------------------------------------------

#### Methods

+ void setDisplayText(String newText)	Set displayText Text
+ void setInteractButtonText(String newText)	Set interactButton Text
+ Button getInteractButton()	Get interactButton



+ void setNoButtonText(String newText)	Set noButton Text
+ Button getNoButton()	Get noButton

### 5.2.2 Class BuyBtn extends Base\_TwoButtonPane

#### Constructor

+ BuyBtn()	-Create an UpdateUIButtonPane with displayText = "Do you want to buy this location?" - set onAction for buy button to call method buyLocation if method return true change owner of location to current player and set style of this tile to match to color of player -update pane -check for skipTurn and end turn
------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### Methods

+ void visible(Boolean b)	Set visibility for BuyBtn
+ void updateBuyButtonScreen()	Set text for interactButton

### 5.2.3 Class TransferBtn extends Base\_TwoButtonPane

#### Constructor

+ TransferBtn()	-Create an UpdateUIButtonPane with displayText = "Do you want to transfer this location to you?"
-----------------	--------------------------------------------------------------------------------------------------

	<ul style="list-style-type: none"> <li>- set onAction for transfer button to call method transferLocation if method return true change owner of location to current player , set style of this tile to match to color of player,remove location form other player and add to current player</li> <li>-update pane</li> <li>-check for skipTurn and end turn</li> </ul>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Methods

+ void visible(Boolean b)	Set visibility for TransferBtn
+ void updateBuyButtonScreen()	Set text for interactButton

## 5.2.4 Class UpgradeBtn extends Base\_TwoButtonPane

### Constructor

+ UpgradeBtn()	<ul style="list-style-type: none"> <li>-Create an UpdateUIButtonPane with displayText = "Do you want to upgrade this location?"</li> <li>- set onAction for upgrade button to call method upgradeLocation</li> <li>-update pane</li> <li>-check for skipTurn and end turn</li> </ul>
----------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

--	--

## Methods

+ void visible(Boolean b)	Set visibility for UpgradeBtn
+ void updateBuyButtonScreen()	Set text for interactButton

## 5.3 Class DicePane extends VBox

### Fields

- Text turnCount	Turn count of the game
- Button rollButton	Button to roll the dice
- Pane diceImage	Image of the dice

### Constructor

+ DicePane()	<ul style="list-style-type: none"> <li>- Initialize roll button,set text to "roll" and set font to 30</li> <li>- update dice image to start with 1</li> <li>- set onAction of roll button to call method rollDice,update dice image with step,update each player pane and change turn</li> <li>- Initialize turnCount,set font to 30 and set fill color to white</li> </ul>
--------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Methods

+ void updateDiceImage(int step)	Set image of dice to match step
+ void updateDicePaneScreen()	Set text of turnCount
+ Button getRollButton()	Return rollButton

## 5.4 Class MainPane extends Pane

### Fields

- ObservableList<TileBtn> TileBtnList	List that contains all TileBtn in board
------------------------------------------	-----------------------------------------

### Constructor

+ MainPane()	- setPrefSize to 700,700 - setBackground with color PINK - initialize TileBtnList -setPosition()
--------------	-----------------------------------------------------------------------------------------------------------

### Methods

+ void setPosition()	Set position of all TileBtn on the board and set Background image of each TileBtn
+ void updateMainPaneScreen()	Update Background image of each TileBtn to match the situation
+ ObservableList<TileBtn> getTileBtnList()	Return TileBtnList

## 5.5 PlayerPane extends Pane

### Fields

- Text playerName	Player name
- Text playerMoney	Player money
- FadeTransition fade1	fadeTransition for player1
- FadeTransition fade2	fadeTransition for player2
- Text playerItem	Player Item

### Constructor

PlayerPane(int player)	<ul style="list-style-type: none"><li>- Initialize playerMoney</li><li>- Initialize playerName</li><li>- Initialize playerItem</li><li>- set each fields by int player that input</li></ul>
+ void updatePlayerpane(int player)	<ul style="list-style-type: none"><li>- update each fields of player</li></ul>
+ void updatePlayerPaneScreen()	Update fade effect for PlayerPane

## 5.6 TileBtn extends Pane

### Fields

- Tile tile	Tile for each TileBtn
- Text buyPriceText	Text that show buyPrice
- Text nameText	Text of tile name
- int owner	Owner of the tile

### Constructor

+ TileBtn(Tile tile)	<ul style="list-style-type: none"><li>- Initilize tile</li><li>- setPrefSize(100, 100)</li><li>- setBackground color to WHITE</li></ul>
----------------------	-----------------------------------------------------------------------------------------------------------------------------------------

	- Initilize nameText and buyPriceText
--	---------------------------------------

## Methods

+ void setBuyPriceText()	If tile is Locationtile set buyPriceText
+ void setNameText()	Set nameText to match Tile name
+ void setToolTip()	Set up tooltip to show information for each tile

## 6 Package main

### 6.1 Class Main

#### Fields

+ <u>Pane gamePane</u>	Pane of the game
+ <u>PlayerPane player1Pane</u>	Pane for player1
+ <u>PlayerPane player2Pane</u>	Pane for player2
+ <u>DicePane dicePane</u>	Pane for dice and roll button
+ <u>MainPane mainPane</u>	Pane for board
+ <u>BuyBtn buyButton</u>	Buy button
+ <u>UpgradeBtn upgradeButton</u>	Upgrade button
+ <u>TransferBtn transferButton</u>	Transfer button
+ <u>GetItemPane getItemPane</u>	Pane when player get item
+ <u>LoseMoneyPane loseMoneyButton</u>	Pane when player lose money
+ <u>boolean skipTurn</u>	Boolean skipTurn
+ <u>DrunkPane drunkPane</u>	Pane when player land on DrunkTile
+ <u>APane aPane</u>	Pane when player land on ATile

+ <u>TiePane tiePane</u>	Pane for game that tie
+ <u>P1WinPane p1WinPane</u>	Player1 win pane
+ <u>P2WinPane p2WinPane</u>	Player2 win pane
+ <u>MediaPlayer mediaPlayer</u>	Media player
+ <u>JailPane jailPane</u>	Pane when player land on JailTile
+ <u>StartPane startPane</u>	Pane when player land on StartTile

### Methods

+ void start(Stage primaryStage)	<ul style="list-style-type: none"> <li>- create new pane and create scene from this pane to be 1200,800 sized ,set title to “Let’s Get Rich In Chula”,set image background,add TextField for player1 and player2 name</li> <li>-set onAction when player1 press enter to disable Textfield for player1 and when player2 press enter call method initializeGame and set primaryStage scene to method createGameScene()</li> </ul>
+ <u>Scene createGameScene()</u>	<ul style="list-style-type: none"> <li>- Initialize bgm for the game</li> <li>- create new gamePane</li> <li>- add all gamePane in fields to new gamePane</li> <li>- set layout for each pane</li> <li>- create game scene form new game pane</li> <li>- return new game scene</li> </ul>

<u>+void main(String[] args)</u>	Launch the program