

**MARMARA UNIVERSITY**

**FACULTY OF ENGINEERING**

**CSE 1142 COMPUTER PROGRAMMING**

**SPRING 2021**

**TERM PROJECT**

**Project Name :** Blockie

**Instructor :** Sanem Arslan Yılmaz

**Due Date :** 28/05/2021

**Submission Date :** 28/05/2021

|  |  |  |
| --- | --- | --- |
|  | Student ID | Name Surname |
| 1 | 150119055 | Mustafa Tunahan BAŞ |
| 2 | 150120506 | Muhammed Cafer TUNCER |

# Problem Definition

We were asked to make a game in which the player has to destroy all the blocks in a level using his/her mouse. The less the player clicks,the more score he/she would get.The main purpose is to destroy as many blocks as possible for each click so that the score will be higher.

The rules are simple:

* There are 4 types of blocks,2 of them being destroyable whereas the other 2 are not.
* When the player clicks a block,the adjacent blocks are also affected except for diagonally adjacent blocks.
* Whenever a block is destroyed,it turns into the lower block type depending on its current type.
* When the player destroys one block from a click,he/she loses 3 points.
* When the player destroys two blocks from a click,he/she loses 1 point.
* When the player destroys three blocks from a click,he/she gains 1 point.
* When the player destroys four blocks from a click,he/she gains 2 points.
* When the player destroys five blocks from a click,he/she gains 4 points.
* Player can’t get to a level before completing all the previous ones.

# Implementation Details

The app class is the main class of the project,it is our presentation layer.All the UI-related things are inside it.

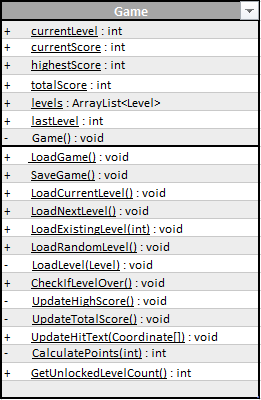
First,the game and levels are loaded by calling Game.LoadGame() in the start() method,

Secondly,the initialization methods are called in the start method(),these initialization methods create the UI’s written inside them,and make sure everything is ready to run.But they don’t show any of the stages, their task is to load stages to memory,so that we won’t have to create them over and over for each transition between stages

There are 4 different stages:main menu,levels menu,quit popup and game window.

Then,main menu is shown to the player calling ShowMainMenu() inside the start() method, and everything is ready to run

All the other methods handle different stuff related to UI.



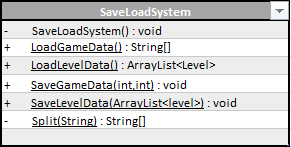
The Game class is where the functionality is added to the game,it is our business logic layer, and is responsible to process the data collected from user input(mouse clicks)

It communicates with both UI and data access layers and processes the data accordingly.

All the methods and fields inside Game class are static and also no-arg constructor is private so that one can create an instance of it which is what’s meant to be done.

LoadGame() and SaveGame() methods call other methods from SaveLoadSystem class so that they don’t do anything related to database.

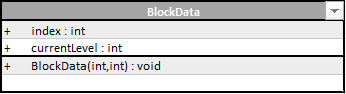
The other methods keep the game running processing data received from user inputs.

 The SaveLoadSystem class handles everything related to text files(our database),it is our data access layer.

The methods inside this class are all static because they’re only called from Game class also where everything is static.The no-arg constructor is private so that you can’t create instances of this type.

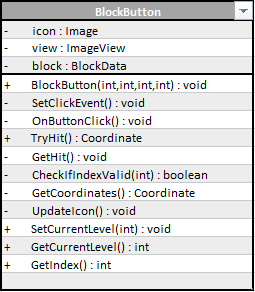
All the data game needs to run its functions is stored in text files,what this class does is read/save the data when necessary.

It takes/returns the data to/from Game class and communicates with no other class.

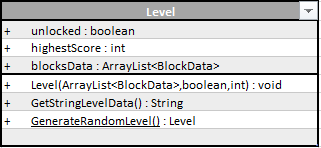


The BlockData class is used to hold index and type data of blocks in the game,the reason why these data are kept in a different class as an inner type is because Level class also uses these data to create levels.

If these data were to be kept directly inside BlockButton class,then all the properties of a button would be loaded to memory for every single block in every single level which is unnecessary in the logic layer.



The BlockButton class extends from Button class in javafx.scene.control,it represents a single block in the game.The click event,image and block data(index and type) are assigned in the 4 int parameter taking constructor.

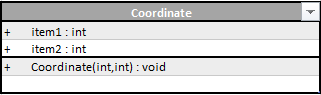
 This class handles mouse click events int the game and returns the data to Game class.

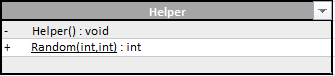
The Level class represents a level in the game.

Levels are created by SaveLoadSystem.LoadLevelData() and returned to Game.LoadGame() and then they’re assigned to Game.levels variable so that the game can function.

SaveLoadSystem class uses Level.GetStringLevelData() method which turns the level data into text in a format which level data is held in.

Level class can also generate a random level using GenerateRandomLevel() method.

 The coordinate class represents a coordinate in the game



Helper class is not an entity of the game,it just possesses some useful static methods that are not directly related to any classes.

It’s no-arg constructor is also private

The only method it has(Random(int,int) )generates a random int between the limits specified in the parameters,it does not exclude either of the lower and upper limits.

# Additional Functions

Our team added a main menu,a levels menu, and a quit popup menu to the game.

Also we added a restart button to the game window,

A generate random level option was also added.So that even if all the levels are finished players can see new levels.

Save and load systems are added so that player’s progress remains when the game is closed

There is a total score label in the main menu besides from each level’s own high score.

There is a current level label in the main menu telling which level will be loaded when player clicks Play button.

# Project Review

We believe our project does not have any incompleteness,we implemented everything we were told to,and we added several additional functionalities which might be modified or improved but we don’t consider them as incomplete parts.

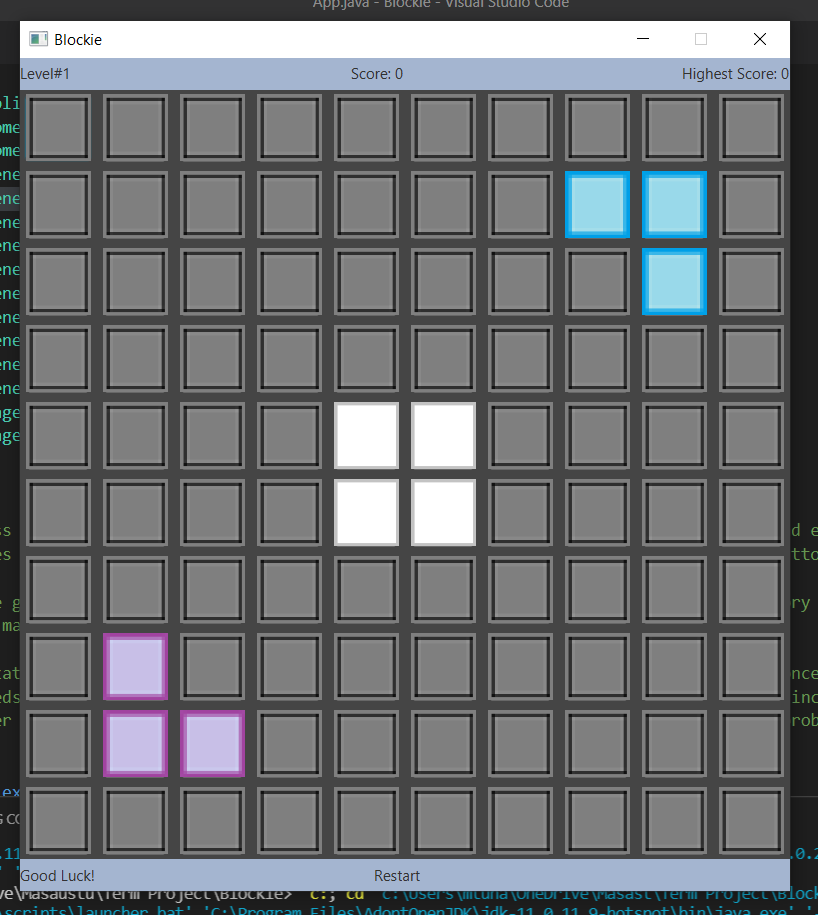
# Test Cases

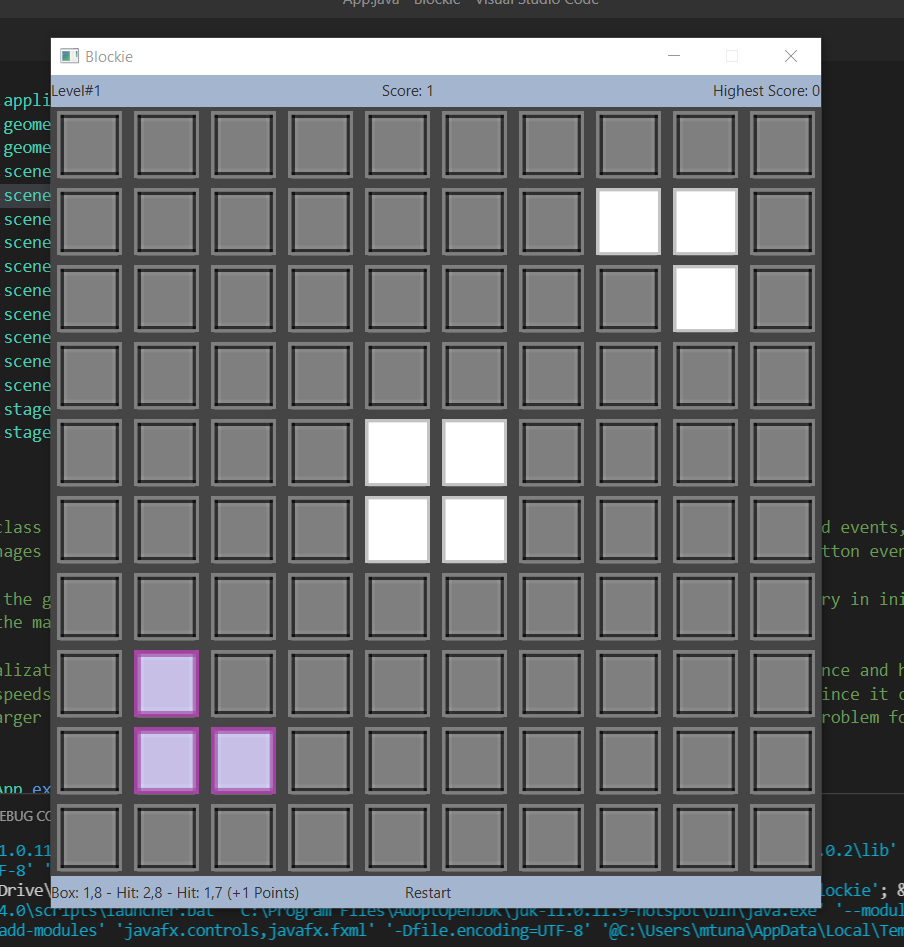


First,we opened the game

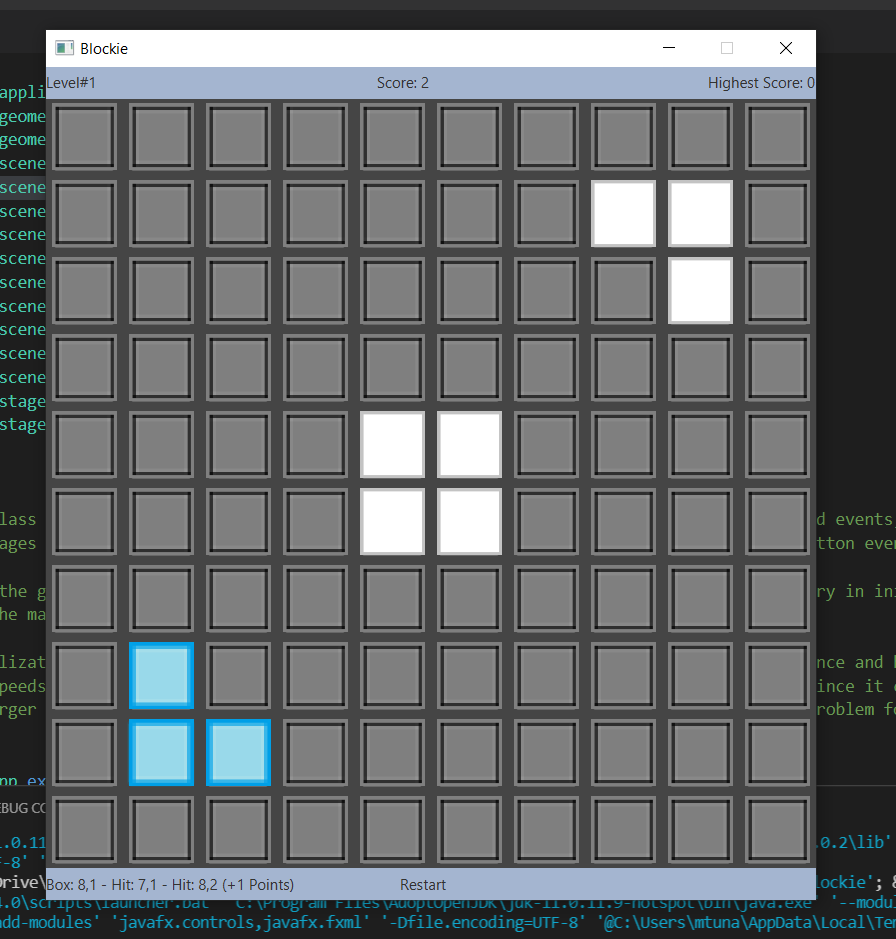
This is how our game opens with default data(nothing is updated/played yet)

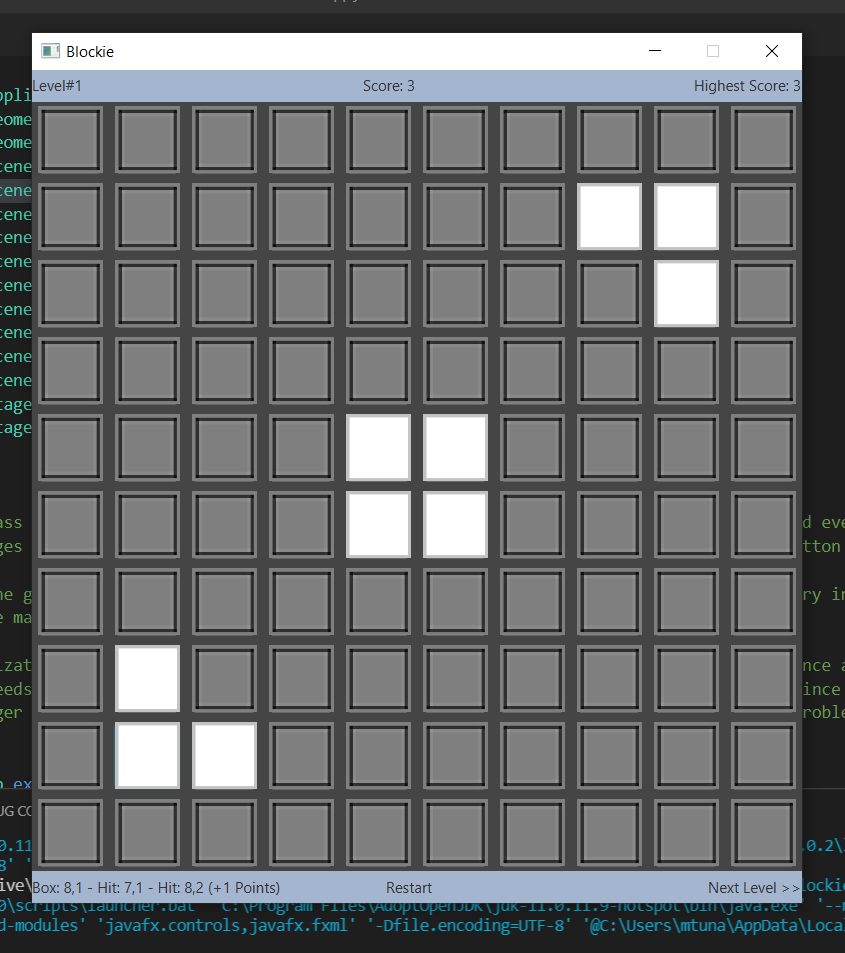
Then we opened the first level





Then we clicked on the block at position (1,8)

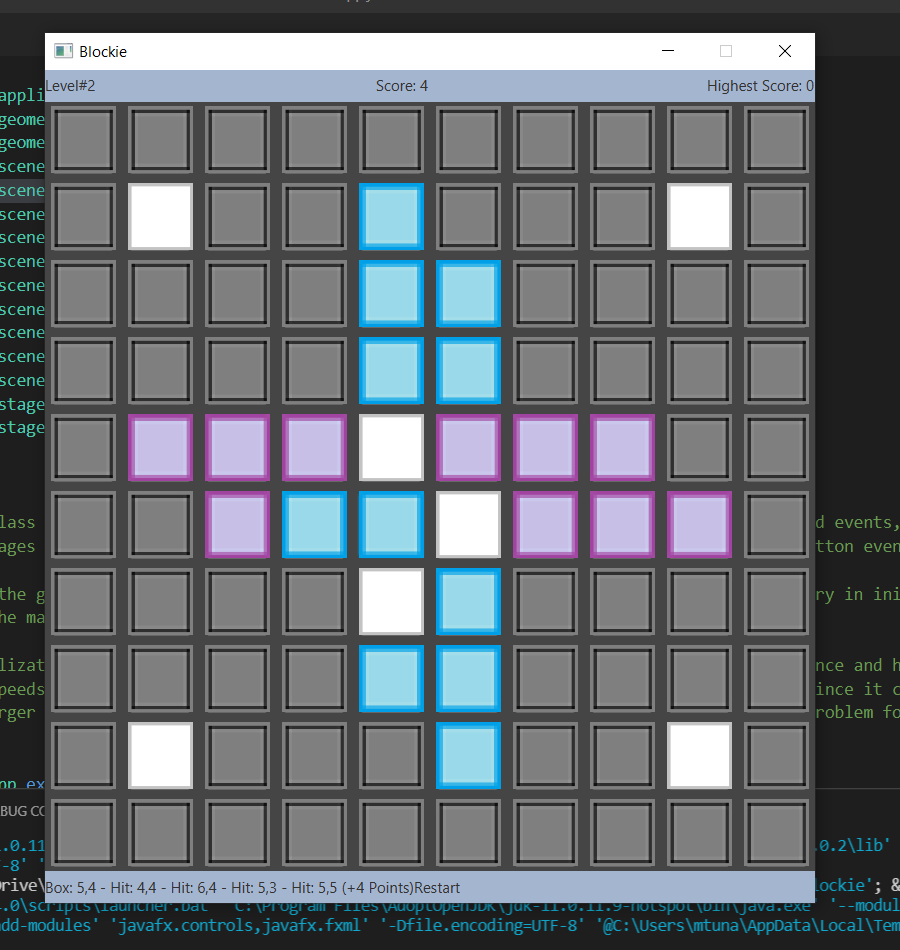
 Then we clicked on the block at position(8,1)

 Then we clicked on the same block again,high score was updated and next level label showed up,we continued with the next level.

Everything worked fine so far.



Level2 was loaded.

 Then we clicked on the block at the position (5,4).

 We kept playing and passed all the levels till level6.

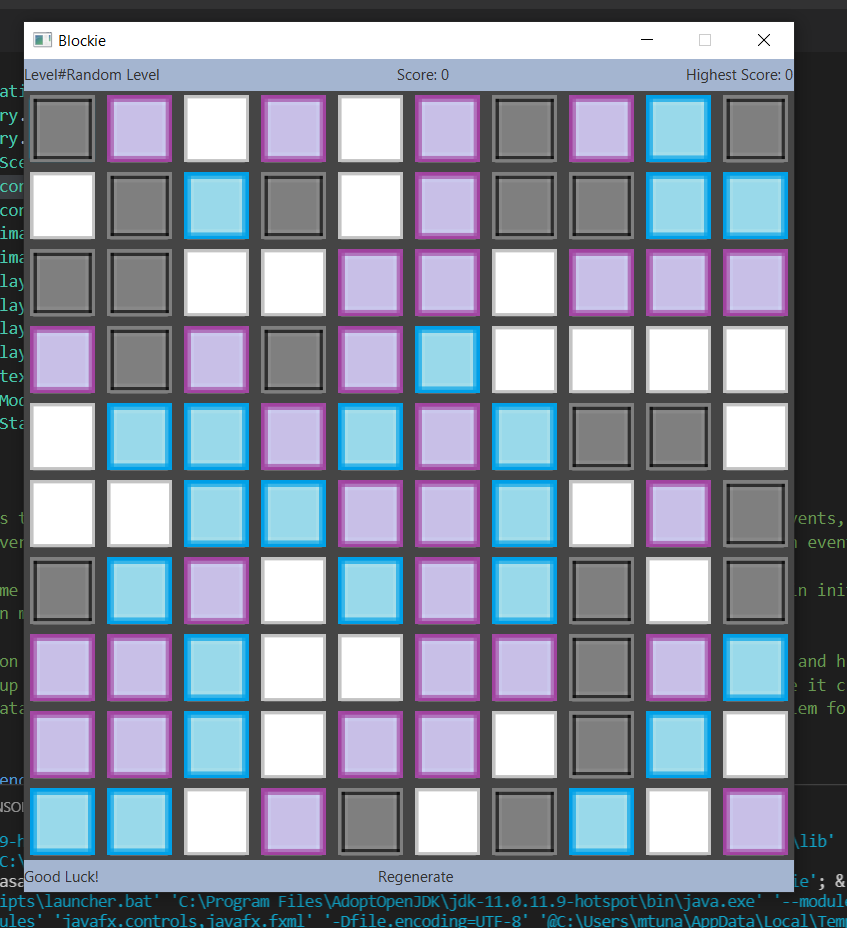
 Then we returned back to the main menu by closing the game window(that loads the main menu)

Everything seems updated and fine.

 Then we entered levels menu.

Levels’ high score and unlocked properties were updated.

We tried clicking on locked levels,but nothing happened which is what we were expecting

 Then we went back to main menu and clicked on Generate Random button which generates a random level and loads it.

It did its work and gave us a nice looking level.

 Then we clicked on the label Regenerate,and it loaded us another random level.

 Then we closed the game window again and clicked on Quit button and it showed this popup.

We tested if No button worked or not,it worked as expected.

 Then,we tested whether the close button on top right of the screen also worked,and it also did work.

So we closed the game clickking on Yes button,and reopened it,and saw everything was still fine which indicates our save/load system also works well.

We tested every possible thing that can be done on our game,and our game passed all the tests successfully,everything worked just fine.