CS 319 - Object-Oriented Software

Final Report

Katamino

Group 1-G

Mert Epsileli

Yusuf Samsum

Fırat Yıldız

Burak Korkmaz

Faruk Ege Hatırnaz

Table of Content

1. Implementation Process

In the implementation process, Intellij IDE was decided to use. Members of the group installed the Intellij IDE inside their computers. By the help of Intellij IDE, each computer has been connected to each other by the help of GitHub. Each computer connected to game repository in the GitHub and attain the power of commit changes on project. The changes made directly should not be sent another one by pulling the commits from GitHub, the changes made on project could easily be sent to members computers suddenly.

There was quite hard to distribute works at the beginning part of the project because the system has seen as inseparable. We tried to separate missions inside the group by looking at the functions decided in the design process. The team meeting were preferred instead of social media applications like Skype. We did not meet in each time with whole group, small separations inside group like 2 or 3 people create more efficient works. Also the usage of such social medias may cause disorder.

In the work distribution, Yusuf and Burak has supposed to work on UI design and frames changes inside the game. Mert was responsible from the Game Database and Firat was responsible from the Control Manager. Ege was responsible for Game Manager. Finally, all members work on the gathering all the components separately implemented. In the implementation part, because the Database and Control Manager parts are easier compared to UI and Game Manager, Firat and Meet help these teams too. If the main requirements of the game would implemented successfully, we think on adding some additional features as determined in the Design Report and Game Explanations in the GitHub repository.

What's more, we would change some features we decided in the design process because unpredictable problems such as the problems in the movement of the blocks forced us these changes. However, Object Oriented Programming which is the main aim of this project was tried to be sustained during the project.

At the end, a Katamino Game meets the main requirements were created for the demo.

1.1 System Requirements

Katamino Game works in JAVA Virtual Machine and requires Java SDK8. Game may work any type of operating system.

1.2 How to Use Katamino

- Enter the link in GitHub account: https://github.com/egehatirnaz/1g-katamino-CS319
- Pull the last version of the game into the computer
- Read the expressions in the Readme file to see the features of the game
- Push the Start" button
- Enter a valid nickname
- Select the mode
- After entering the game frame, you can play the game by using component in the Game Screen, enjoy it!

2. Design Changes and Improvements

In the design process, we design a Vertex Class which represents one of the blocks which will be used in the game. Additionally, there was a Square Class which represents each piece of this block. According to our design, these pieces would bind in a piece (which is vertex) and they would act by this master piece. There was some classes and methods designed in order to combine them. However, these became insufficient for combining the pieces and acting them with master piece(vertex). Also there is a two dimensional array which keeps the sides of the Squares with reference to Block(master pieces) add as parameter. Because of all these, we forced to change the design which works with images. We created images by the help of our Vertex Classes and used them in the implementation of actions such as rotations or dragging. We used the Jfxtras library to move image created. It has also some methods for rotation and dragging inside. These functions were used.

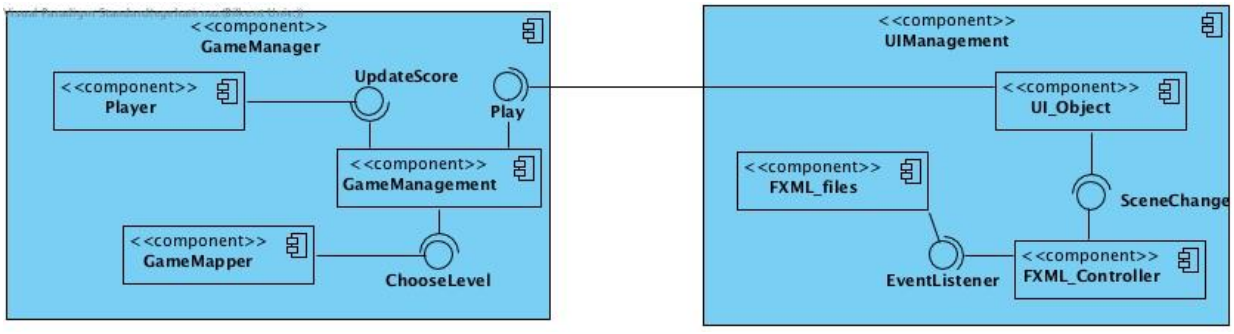

Figure 1) The System Diagram

2.1 Flexible Game Improvement

Normally, in the report we promised to implement the game which shapes according to user's playing style. However, we could not managed to do it. Instead we add an increasing size profile in the game. That offers to players a game which can be played in the changing sizes. The game frame is also flexible for the changing decisions of players. Players have chance to change the position of the block after it puts as it is seen in below. The brown line shown in *Figure 2* may move in increasing levels.

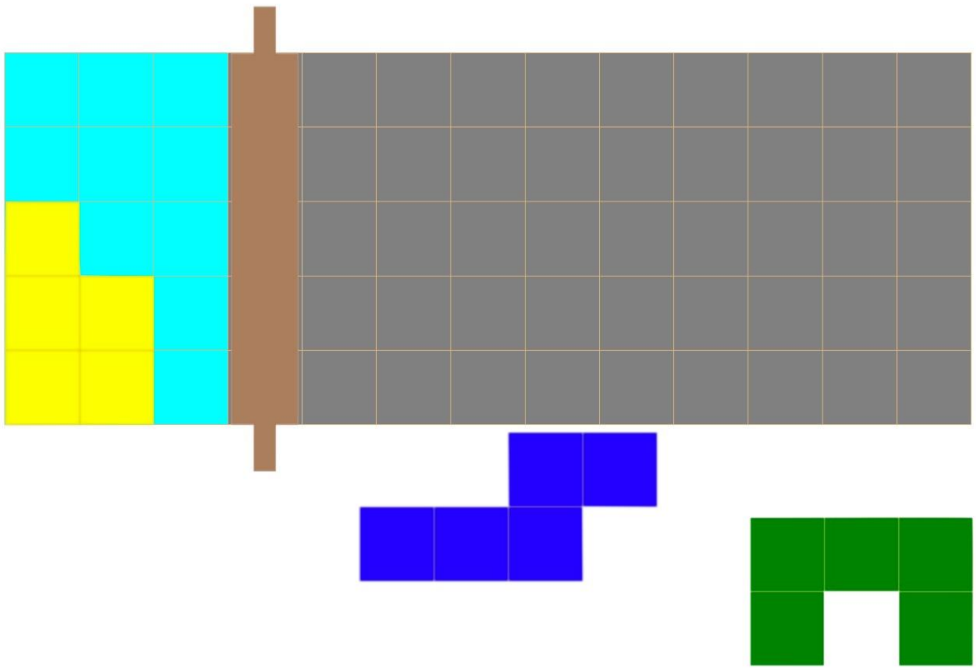

Figure 2) Flexibility in Main Screen

2.2 Dragging and Rotation Improvement

In the design stage, the drag and rotation with the blocks has not been promised but they implemented by the help of Jfxtras library in Java. In Java, it is possible to change the position of block or an image by the help of Jfxtras library. Normally, the refresh blocks feature have been promised but we did not managed to implement it because the blocks become passive after they add into the list (board). The blocks should be flexible to move but in our implementation model, it was quite hard to immobilize the blocks inside a panel if the blocks wanted to be moved. So that will be tried to implement in the final design. The actions are shown in Figure 3 and 4. Players have ability to rotate with double click and drag with mouse.

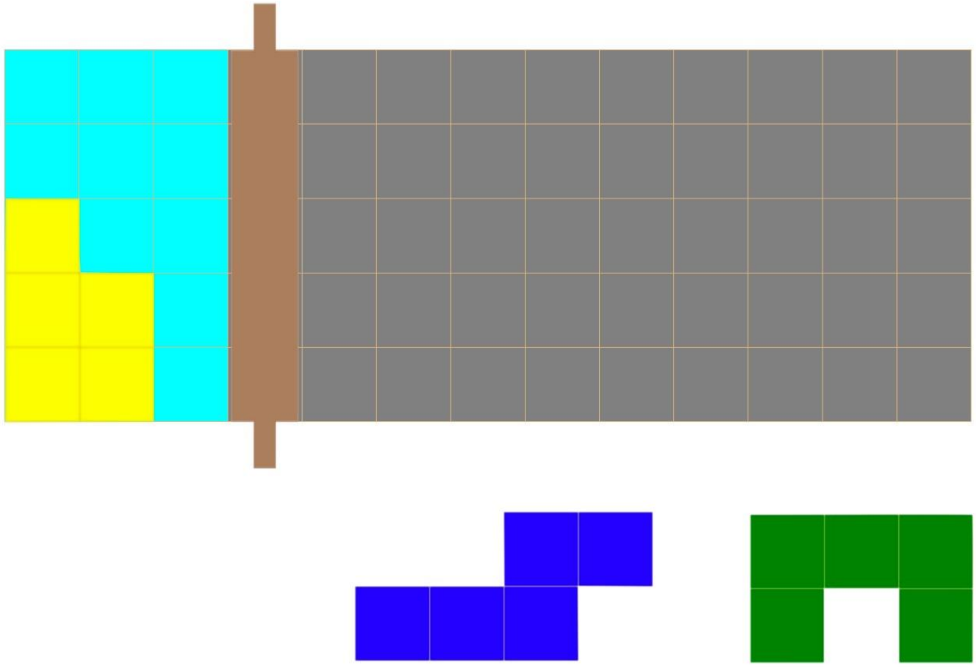

Figure 3) Rotation in Main Screen (1)

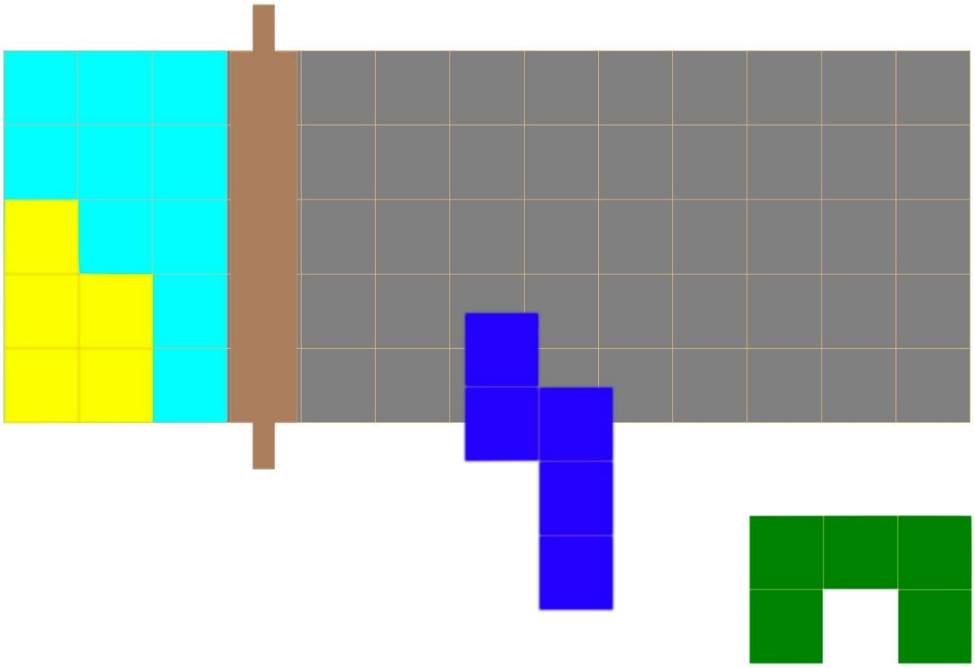

Figure 4) Rotation in Main Screen (2)

2.3 Immobilizing the Block Improvement

Working with the coordinates was not possible to move the blocks. So, if player wants to immobilize it, he/she should click the mouse. By the help of image combining methods, the board and the image combined to each other. If it is combined, that is automatically immobilizing and that block is become inactive. After a piece become inactive, it inactivates the related part of the board too. By the help of the data which says all the pieces of the board is inactivated, the game may be finished. As it is shown in Figure 5, pieces may be strict their places.

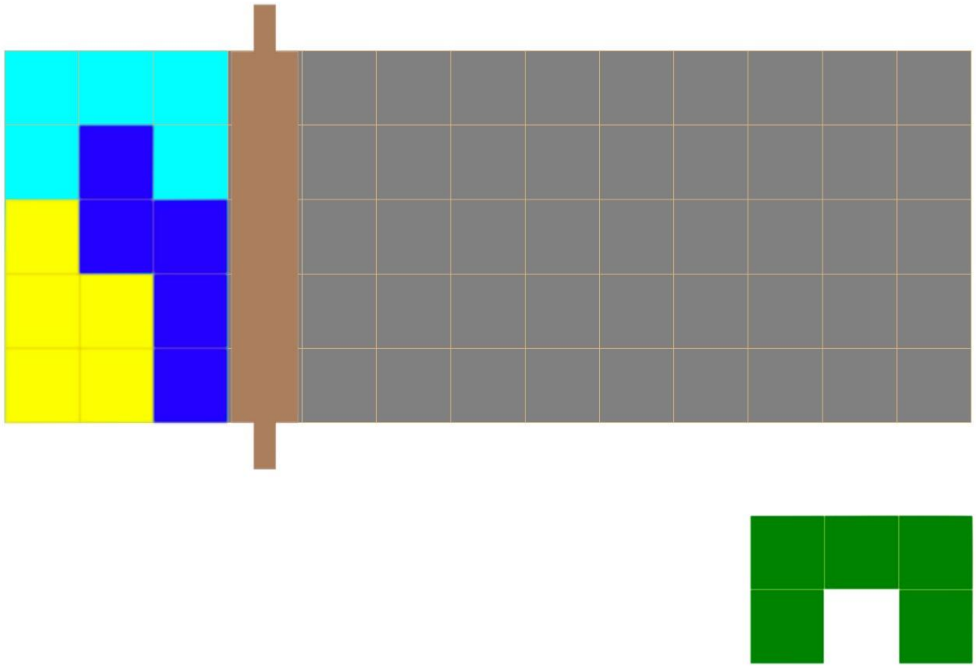

Figure 5) Immobilizing in Main Screen

2.4 Music Usage Improvement

If the player wants to adjust the music level or change the type of the music inside game, he/ she does that inside the settings part of the project. These buttons were activated and open to use it however they work starting from the first demo of the project. User have chance to adjust the music level but there is small bugs on changing the music type in the first demo which will be solved in the last demo.

2.5 Leaderboard Improvement

As it is promised, we add the leaderboard screen after the game finishes for the last demo. It will not be visible in the first demo because the name control system will be finalized after the first demo. A Leaderboard Database created by the information came from Nickname Control system. The scores and the valid names kept in the system will create a leader board for players at the end of the game. The Leaderboard Screen will disappear after the game finishes below. The last part of the Game Frame is shown in the Figure 6 in below. After that frame the leader board become visible as shown in Figure 11.

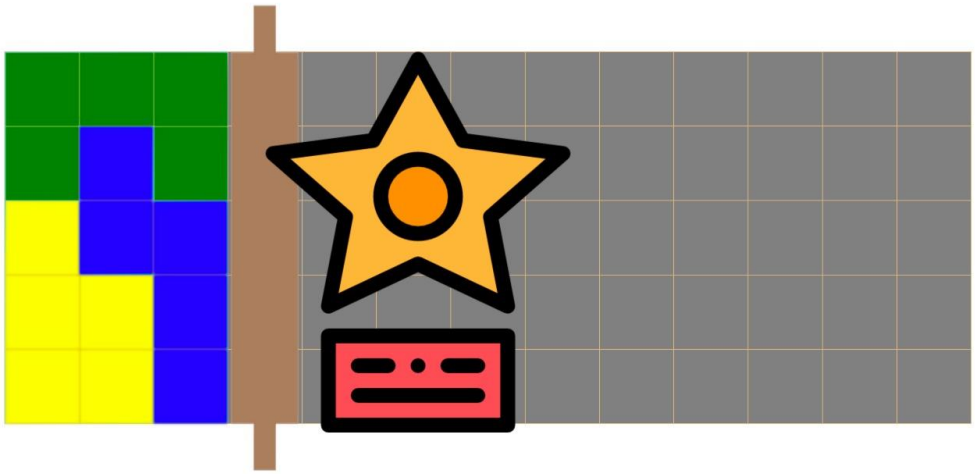

Figure 6) Leaderboard Screen

3. User's Manual

3.1 Start Frame

This the first screen when the user see firstly is the start frame. The start frame includes the options "Start", "Settings", "Quit". User may choose either of them. Game starts with the "Start" button.

Figure 7) Start Screen

3.2 Settings Frame

The options of the game will be included to the “Settings" which are Music and Font. User have chance to change the initial settings according to their preferences.

3.3 Nickname Screen

The Nickname Screen will get a name from the user and will check whether it is true. Unless the nickname is true, the invalid nickname message disappears in the screen and user should try another one to enter the game.

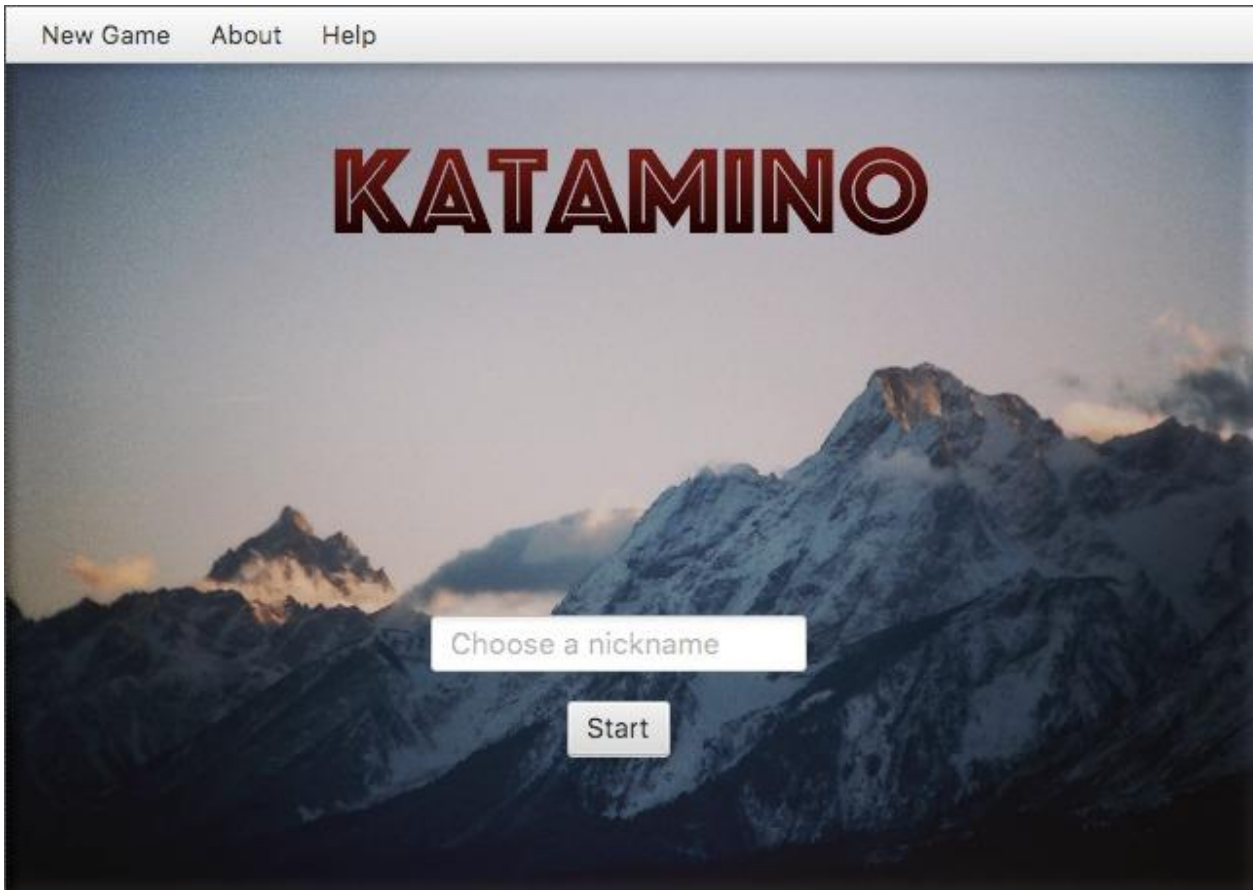

Figure 8) Nickname Screen

3.4 Choose Game Mode Screen

This screen is for user to choose the game mode he wants to play and a link between the game frame and nickname choose class.

Figure 9) Game Mode Screen

3.5 Game Screen

In the Game Screen, a small game frame, the area where blocks stay, a hint button exist. The hint button is used when the user can not find the solution. Computer will put the right block into its right name by this button in each push. Blocks stay in the frame are available for dragging and game will be played by the help of them. Game frame will be updated if the block is put in it's right place. The user also have chance to quit game by closing screen. As shown in the Figure 10, the Game Screen is quite simple and easy to play. Player will just drag these components into the blue area. Shaded area is forbidden for players. It is open in increasing levels.

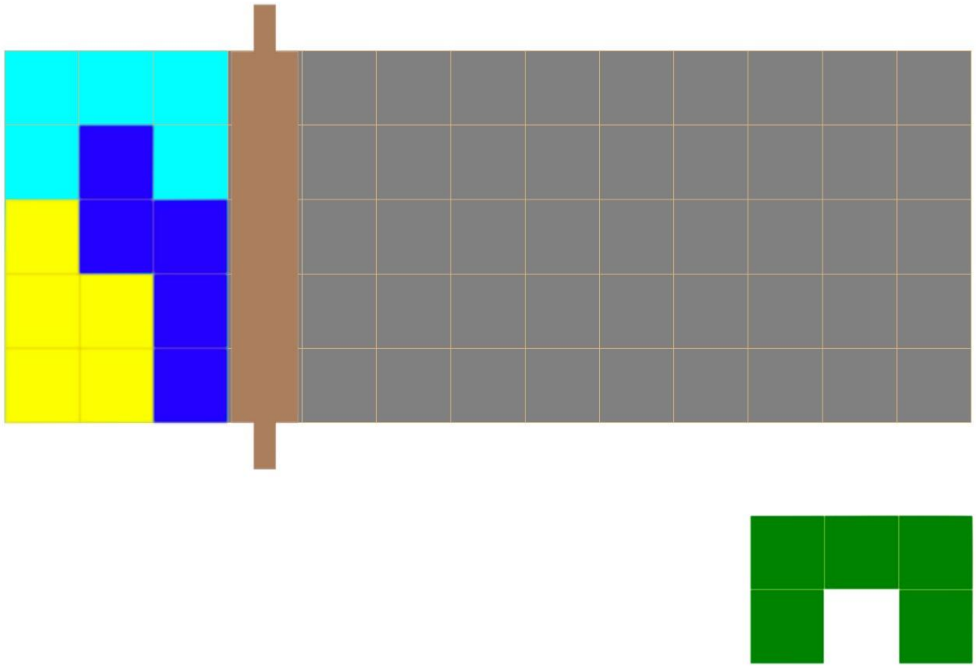

Figure 10) Game Screen

3.6 Leaderboard Screen

The leaderboard screen is visible when the game finished. It opens automatically at the end of the game. It gets datas from the Leaderboard Database and shows the leaderboard according to the entered nickname.

Figure 11) Leaderboard Screen

4.0 Glossary & References

1. Write Once, Run Anywhere

   (https://en.wikipedia.org/wiki/Write_once,_run_anywhere)

2. Java Virtual Machine (Bill Venners, *Inside the Java Virtual Machine* Chapter 5)

3. Java Runtime Environment (https://techterms.com/definition/jre)

4. Java Development Kit (https://en.wikipedia.org/wiki/Java_Development_Kit)

5. JavaFxtra Library (http://jfxtras.org/)