



CS319 - Object Oriented Software Engineering

Final Report

Katamino PC Game

Group 3C

Murat Tüver , 21602388

Sera Fırıncioğlu , 21401803

Berk Yıldız , 21502040

Pegah Soltani , 21500559

Can Savcı , 21300803

1. Implementation	2
1.1 Subsystem Requirements	3
1.2 User's Install Guide	3
2. Changes & Improvements	3
2.1 Design Changes & Improvements	3
2.1.1. Highscore and Leaderboard Improvement	4
2.1.2. Coin and Bonus Improvement	4
2.1.3. Multiplayer Improvement	4
3. User's Manual	5
3.1 Main Menu	5
3.2 How to Play?	6
3.3 How to Choose a Mode ?	7
3.4 How to Proceed ?	8
3.5 Settings	9
3.6 Credits	10
4. What is Missing ?	11
5. Conclusion	11

1. Implementation

We decided to implement our game in NetBeans for more compatibility. It is used FireBase Realtime Database to keep the data of leaderboard. We used Facade and Singleton Design Pattern for GameManager, GameEngine and SoundManager classes. For the GUI of our project it is used Java AWT API.

The work done by each group member is as follows :

- We all contributed to the reports as much as we could by dividing the reports as parts to finish more efficiently and in a short time. For the UML diagrams in reports, we mostly work together while drawing the most of the diagrams since we also discussed and share our ideas about them to make correctly.
- Murat continued to work on the controller class, Database and Multiplayer process. He also worked on the game logic like drag&drop, movements of the pentominoes. He contributed to the Class Diagrams with Pegah.
- Pegah contributed the slides, worked on the pentominoes and board design with Can. She also contributed Class Diagram, State Diagram, GUI. She contributed Theme and Sound Classes.
- Berk worked on Level classes and MainMenu classes. He also contributed to reports.
- Sera worked on MockUp and GUI (UI/UX). She contributed to reports, slides and worked on Theme and Sound class. She worked on Use Case Diagram with Murat.
- Can mostly worked on design of the board and the pentominoes. He contributed to State and Sequence Diagrams with Pegah. He also contributed to the reports, slides, Theme and Sound classes.

1.1 Subsystem Requirements

Katamino can be downloaded and played in any computer which has Java Virtual Machine and Java SDK.

1.2 User's Install Guide

Players can see the related reports and description of the game from the link below :

<https://github.com/murattuver/Cerca-Trova>

Katamino can be clone from GitHub via bash or download as zip to play.

- For players who are not familiar with the Katamino, it is provided Tutorial option in the menu.
- For the credits, it is provided Credits button in the menu.
- For any other information about the game, it can be seen our reports from the GitHub link above.

2. Changes and Improvements

2.1. Design Changes and Improvements

In the first iteration, our design served the purpose of implementing single player; however, for the multiplayer, we had to change our design. We stucked MVC design pattern while implementing the game. In addition to this, we used a subsystem decomposition model while finishing the implementation of the game. We added new classes and we re-arranged the existing packages because of adding the multiplayer option.

As we mentioned before, we used Google Firebase Realtime Database to keep the data of the Leaderboard. Also, we used Facade and "Singleton Design Pattern" for GameManager, GameEngine and SoundManager classes, because we wanted only one GameManager object to keep track of the game, only one GameEngine object to run the

game without an intervention and only one SoundManager object to keep track of which sound is going to be played.

2.1.1. Highscore and Leaderboard Improvement

We considered implementing the feature of highscore and the leaderboard only for the Classic mode. However, we implemented the features mentioned on the above on the Arcade mode as well. Players can see the leaderboard on the menu.

2.1.2. Coin and Bonus Improvement

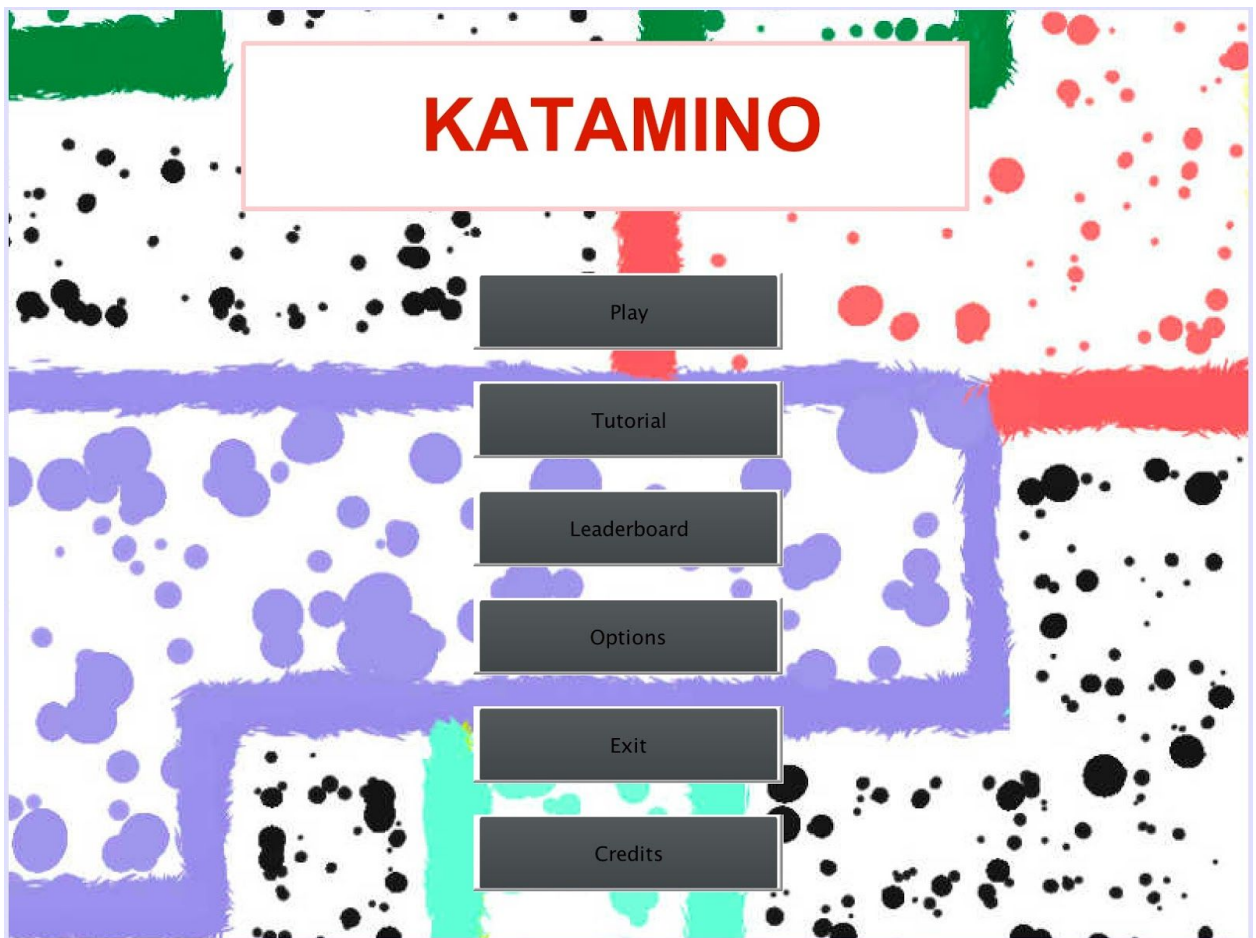
First, we considered adding the coin feature on the Classic mode, as we mentioned for the case of adding leaderboard. However, we added the feature on Arcade mode as well. The amount of earned coin will be proportional to the remaining time when the puzzle is finished. When enough coins are earned, the player can buy the objects whose area can be one or two square(s). We called one and two square(s) as “Bonus Pentominoes”. However, it will be hard to buy them, because they will be the game changers. We believed that the feature can make the game more attractive and more competitive, otherwise the game would be so ordinary.

2.1.3. Multiplayer Improvement

In the first iteration, we did not consider the multiplayer game option and we did not implement it. However, we reviewed critics and we decided to implement the multiplayer option. We have managed to finish in the second iteration. The multiplayer option requires only two players and the players can play together by selecting the multiplayer option on the main menu.

3. User's Manual

To install and execute the game, the players should follow the instructions on the section 1.2 which is called "The User's Install Guide". When the players successfully execute the game, they should see the main menu.



3.1 Main Menu

The main menu is the main screen of the game and the player can make the transitions using the menu. If the player does not know anything about the game, he/she can use the Tutorial option. If the player knows the core logic of the game, he/she can play the game by choosing whether Single Player or Multiplayer button. The player can start a single player game if the user wants to play alone. The players can start a

multiplayer game if they want to play together. Multiplayer game starts a new game with only two players.

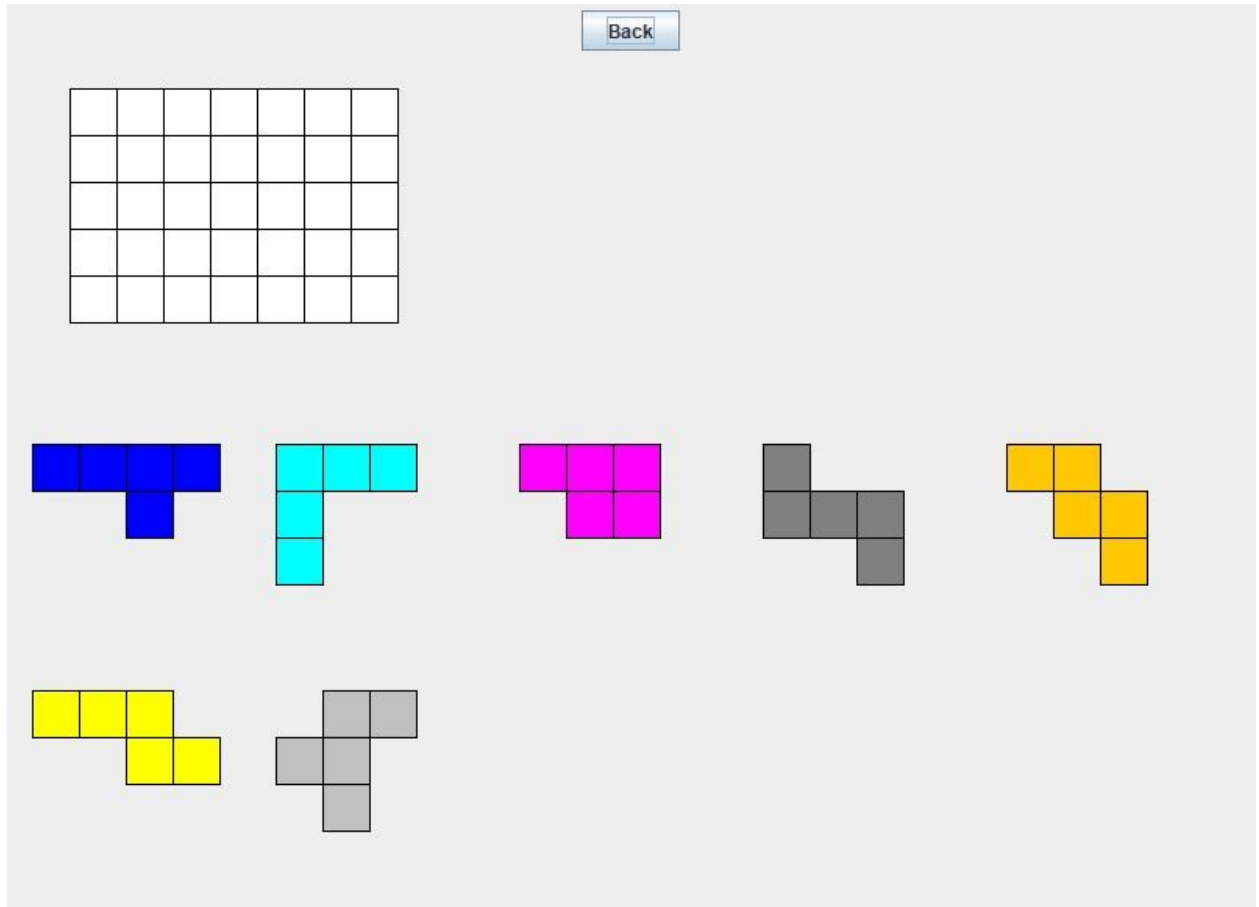
3.2 How to Play

The player can simply click on the Tutorial button on the main menu. After that, the player should watch a video which explains the core logic of the game. The video will be short and precise, after watching the video, the game will ask the player whether he/she wants to play the sample level. If the player wants to play the sample level, he/she should click the “Yes” option, then the sample level will start. In the sample level, the player should simply execute the instructions in order to proceed. Instructions will be like as follows: “Choose a pentomino”, “ Drag the pentomino”, “Rotate the pentomino”, “Put the pentomino on the board”, “Complete the puzzle”... etc. When the user finishes the sample level, he/she will be brought in the main menu.



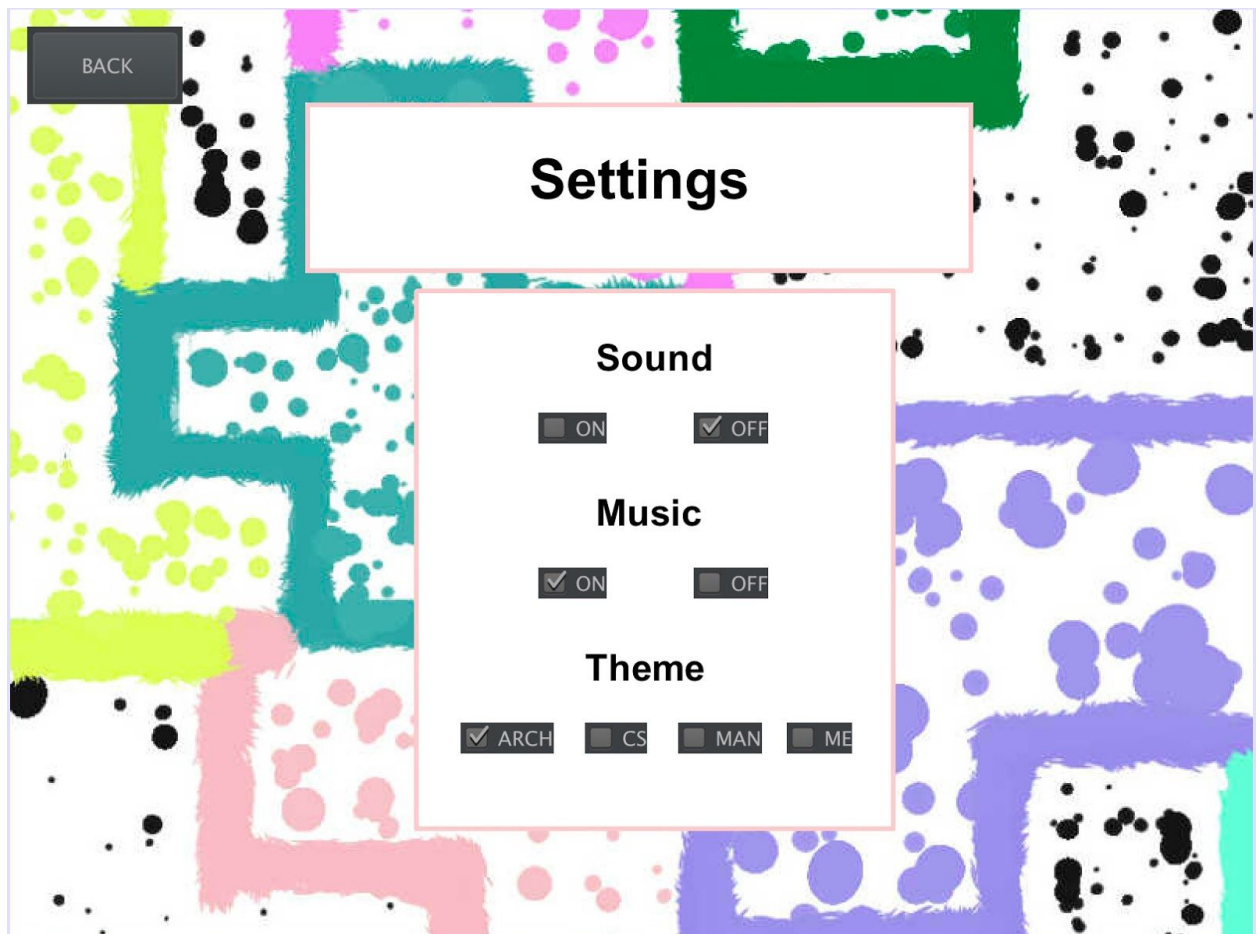
3.3 How to Choose a Mode

Whether the player chose Single Player or Multiplayer option, he/she can choose the modes which are Classic and Arcade. Classic mode is designed to give the player a chance to practice and improve themselves on the game. Classic mode has no timer and the player can finish it regardless of the remaining time. However, the Arcade mode is more competitive because of the timer, the remaining time matters. Regardless of the selected mode, the player will choose a level, it could be from the first level to the last unlocked level. When the player chooses the level, he/she can finally play the game.



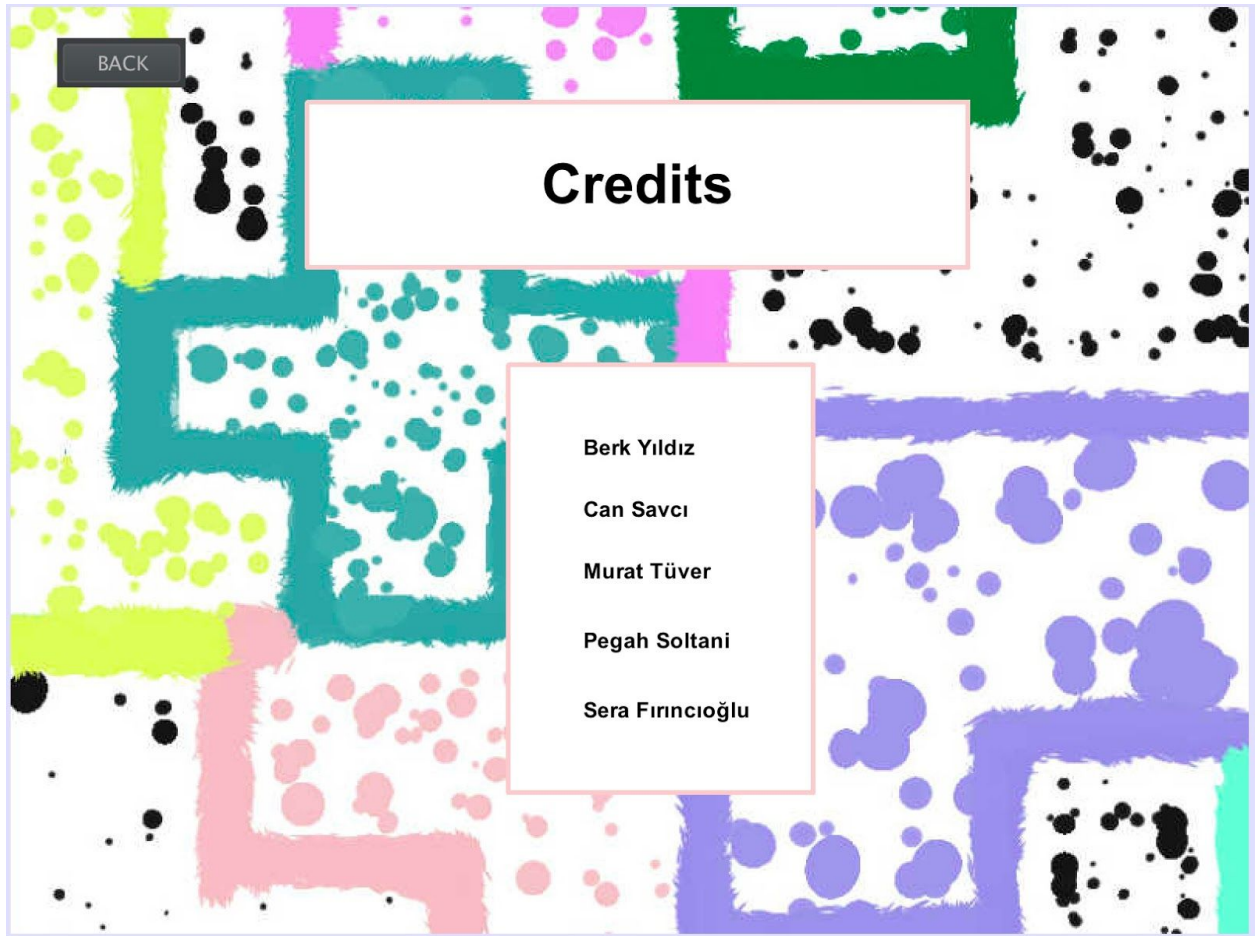
3.4 How to Proceed

The player has to complete the puzzle to win the game. To complete the puzzle, the player has to put the given pentominoes to the corresponding places. The player can drag a pentomino to drop at the board or rotate the pentomino to place better. If the player plays in the Arcade mode, he/she has to complete the puzzle before the timer stops. Otherwise, there is no restriction on finishing the game. In the Arcade mode, if the timer stops before the player completes the puzzle, the level restarts. Otherwise, if the puzzle is completed, the player will unlock the next level until all of the levels are finished.



3.5 Settings

The player can view/change his/her settings by pressing the Settings button on the Main Menu. He/she can turn on/off the music and sound, he/she can change the theme.



3.6 Credits

The player can see the designers/developers who contributed more or less by clicking the Credits button on the Main Menu.

4. What is Missing ?

We reached all of our goals mentioned in the previous reports. However, if we could make the self-criticism, we could add more features to our existing game. On the other hand, the game has the strict rules, we could not make it flexible. Other than that, we are happy because we reached our goals.

5. Conclusion

We finished implemented the project and we saw that the key of following “The Principles of Object Oriented Design” is the internal / external interactions. The external interactions can be described as the information exchange within the stakeholders. From the developers’ point of view, they can learn what the exceptions and the desires are. When the analysts and the developers deduct the expectations and the desires, the internal interactions should be made. The interaction should mainly focus on how to add more features, how to implement the application and how to make the application more interesting, portable and sustainable. We did great at brainstorming and lots of ideas were put forward.

In order to make a successful internal interaction, the interactions should be consistent and we set the meetings according to the principles of working together. However, our meetings were set instantaneously, because we had the different schedules. Moreover, we needed to stay approximately 8 - 12 hours in the meetings. Therefore, the productivity of meetings were decreased as the time passes. We all had other responsibilities; therefore, we could not follow the schedule entirely. We could set the short meetings and we could have distributed the jobs or goals and we could set the deadline for it. However, we did not do the way which is mentioned previously because it needs lots of time and effort, we had other responsibilities to do.

We implemented the project according to the internal interaction, we saw each member’s strengths and weaknesses and we distributed the responsibilities among ourselves, some of us helped developing the project and some of us helped designing the UI/UX of

the application. We have covered the weaknesses of each other. Sometimes we had a hard talk but we managed to make the healthy interactions.

In conclusion, Object Oriented Design is no other than establishing good relations and trusting each other. Coding is the 10% of the Object Oriented Design. Making the analysis of a problem and creating a design to model the problem or the solution is important as well but the analysis and the design stage comes from internal and external interactions; therefore, if we make the interactions healthier, the probability of making a desirable application will be increased. We tried our best to reach that conclusion and implementing the conclusion in our term project.