Bilkent University

Department of Computer Engineering

CS-319 Project

Quadrillion

Final Report

Group No: 1H
Group Name: COGENE

Osman Orhan Uysal

Samet Özcan

Mehmet Alper Karadağ

Ziya Erkoç

Talha Murathan Göktaş

Introduction **3**

Design changes **3**

Lessons learnt **4**

User's Guide **4**

System requirements 4

Installation 4

How to use 4

How to play 5

How to compose level 5

1. Introduction

As soon as UML diagrams had finished, we started implementing the game. Firstly, Orhan Uysal created the skeleton code out of UML diagrams and implemented some basic functionalities. He is also responsible for implementing showing hint and coming up with an algorithm to find out if a board has a solution. Then, Ziya Erkoc prepared initial GUI elements, implemented some of the Piece and Ground functionalities such as moving, rotating and flipping (rotating the Ground is implemented by Alper Karadag). Ziya was also responsible for mainly Database interactions and partially Payment System. Alper Karadag; however, initialized the Payment System integration and Ziya maintained it. Samet and Murathan who put a lot of effort in preparing the reports are not much needed in the first-iteration implementation since they are responsible for furnishing the UI. Because we left improving UI to the second iteration, they will heavily take place in there. Below is the list of completed and uncompleted features:

Currently, we completed the following functionalities:

1. Game can now be both played in single player or ranked mode.
2. User can compose a new level and send it to the database.
3. Users can select levels (either user-created or predefined).
4. User can see leaderboard of the level they are playing.
5. User can buy hints. Currently, no real money transaction occurs but fake money.

Although we implemented above functionalities, our game suffers from lack of shining UI/UX. That is, although game functions well, user interface is not charming enough. Samet and Murathan will be responsible for it in the next iteration.

Uncompleted functionalities:

1. Users cannot use hints; namely, hint revelation functionality is not included yet
2. Pieces are not structured well within the game area (we are planning to add scrollbar).
3. Poor UI/UX
4. Back-board is not usable.

2. Design changes

We made the LevelManager and User class Singleton since only one instance of them will be needed.

We also added couple of new methods and new attributes to each class that won't be shown here for brevity.

Other than these, we have been faithful to our UML diagrams.

3. Lessons learnt

Having UML diagram before implementing relieved our burden a lot. Since we made sure that our logic works in UML, we just figuratively converted UML diagram into code.

We also realized the power of Design Patterns and harnessed it a lot in our projects. One of our confess is we gave so much responsibility to Core Game subsystem and especially to LevelManager class. Hence, our members who worked on that part did a lot of job. We could have divided the system more evenly.

We made LevelManager and User class singleton but this approach does not seem to be a good practice because every class will have access to both User and LevelManager functionalities. To address this issue, we are planning to have general manager class that may include instance of LevelManager and User class to manage the game. In that way, they do not need to be a Singleton anymore.

Also, we tried to position UI elements (i.e. buttons, text fields, labels) on our own instead of making use of JavaFX SceneBuilder. Hence, in the next iteration we plan to build our user interface using JavaFX's built-in UI builder.

4. User's Guide

a. System requirements

Quadrillon requires JVM (Java Virtual Machine) to run on a machine. Also, currently it requires internet connection to login and purchase hint operations. However, we are planning to add offline mode to the game.

b. Installation

- Go to the link below to see the project and reports
  https://github.com/Rgtemze/Quadrillion-CS-319
- Download and open the quadrillion.jar file
- Congratulations you have installed Quadrillion.

c. How to use

- First user has to log in to his account if he has no account he has to a create new account.
- In main menu user can select to play a casual game, ranked game or he can compose a new level.
- In casual game user can use hints select which level to play.
- In ranked game levels are given to user randomly and user cannot use hints. Also, his records after finishing the game are recorded into that level's leaderboard.
- In compose level user can create his own level by moving, rotating and flipping the 4 grounds.

d. How to play

- The purpose is to fill all empty spots with pieces.
- User should drag and drop the pieces in order to complete the game.
- Pieces cannot be dropped even one of the circles overlap with an unavailable spot.
- Pieces can be flipped with right mouse button and rotated with middle mouse button.
- User can use hints if he gets stuck.
- User can buy hints if he has no hint.
- If user can fill all the spots he can click submit button to complete the game.

e. How to compose level

- The purpose is to create new challanges for players by discovering new board combinations among quadrillion possibilities.
- User can drag and drop grounds.
- Grounds can be flipped with right mouse button and rotated with middle mouse button.
- User can click the check button after finishing composing level, if the level is valid, namely it is connected and has valid solution.

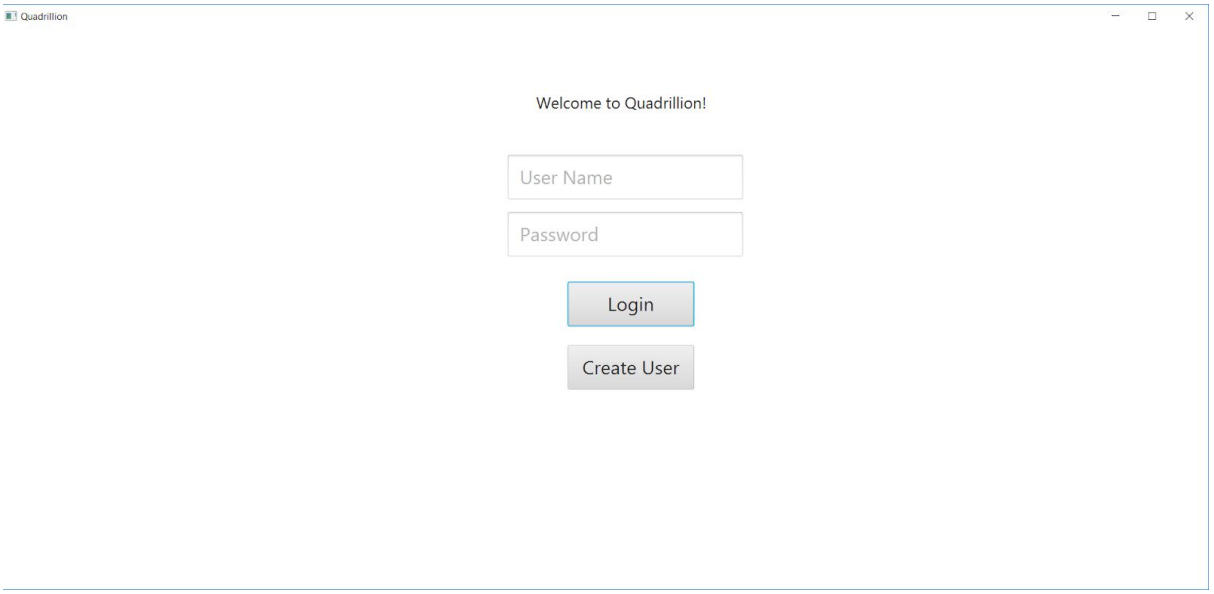

Figure 1: Login Screen

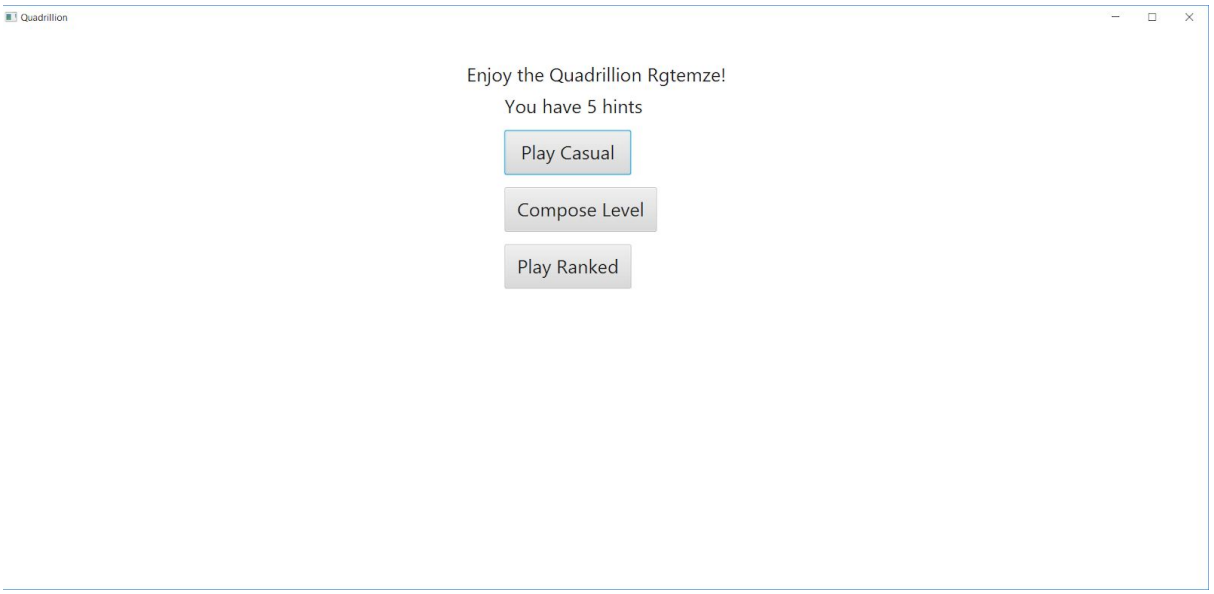

Figure 2: MainMenu Screen

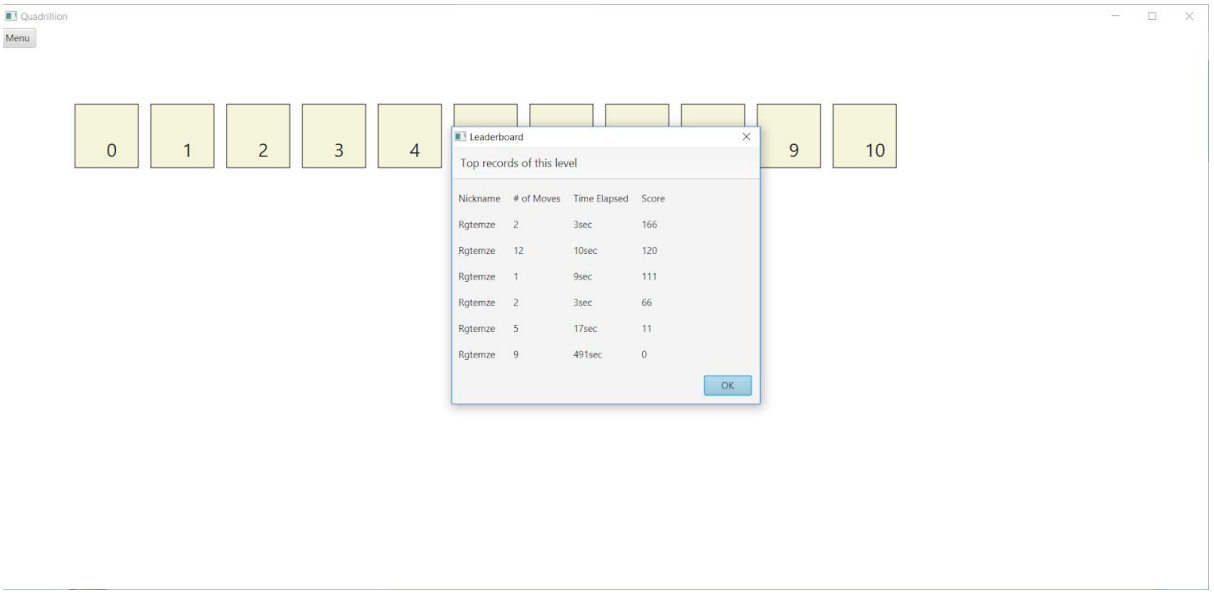

Figure 3: Leaderboard Pop-up

Figure 4: SelectLevel Screen

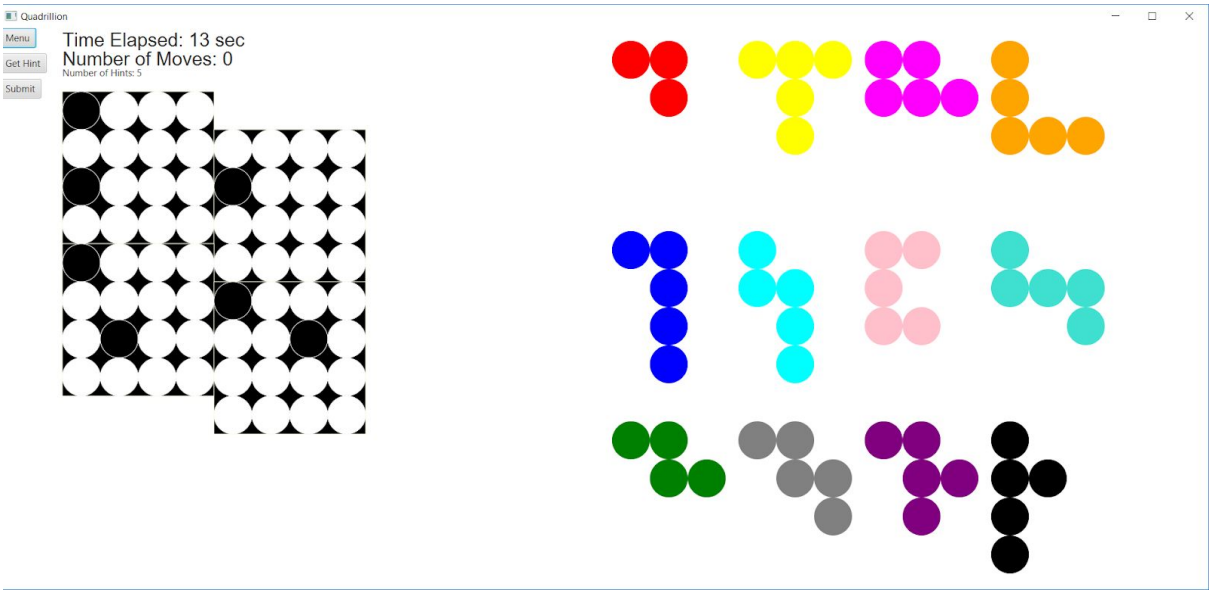

Figure 5: PlayGame Screen

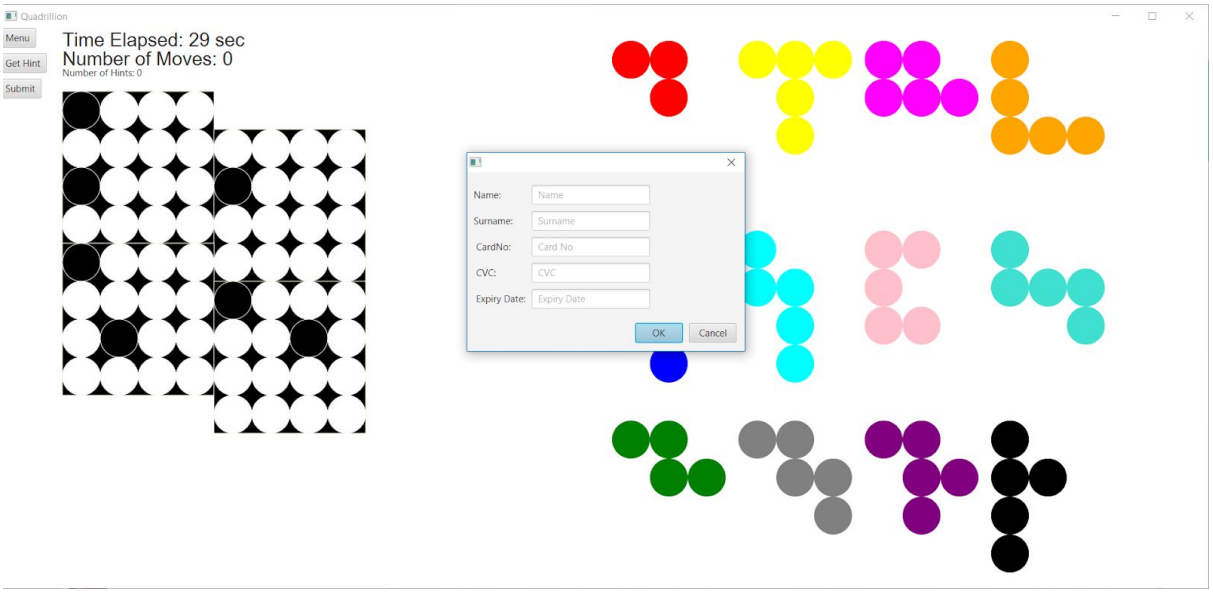

Figure 6: Hint Purchase Popup

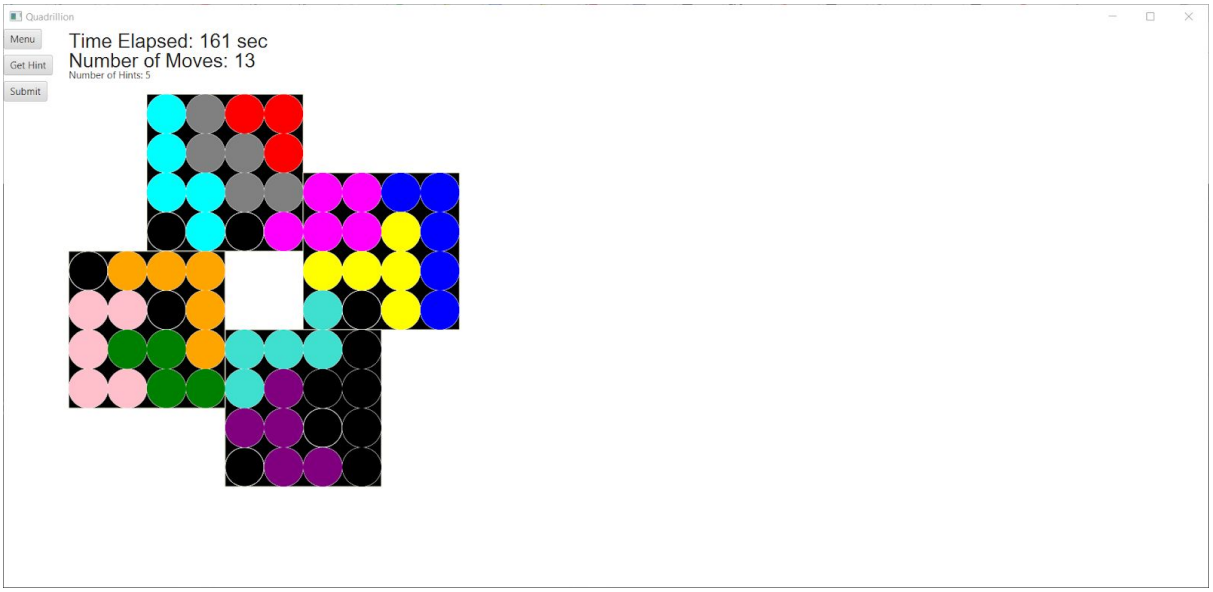

Figure 7: An example of Completed Level

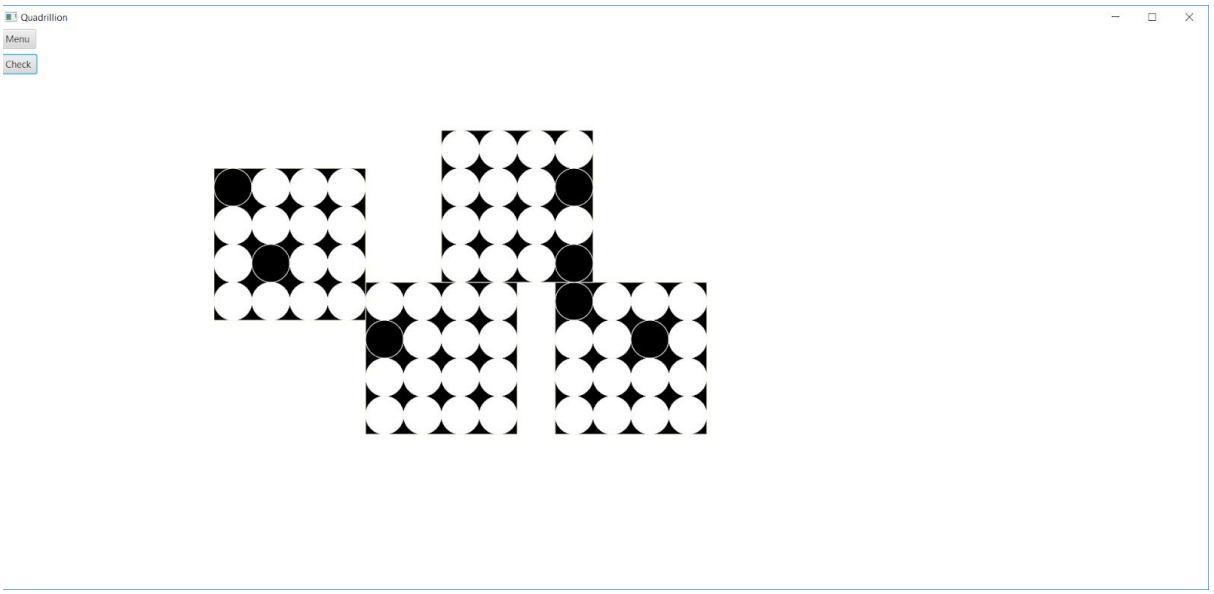

Figure 8: ComposeLevel Screen

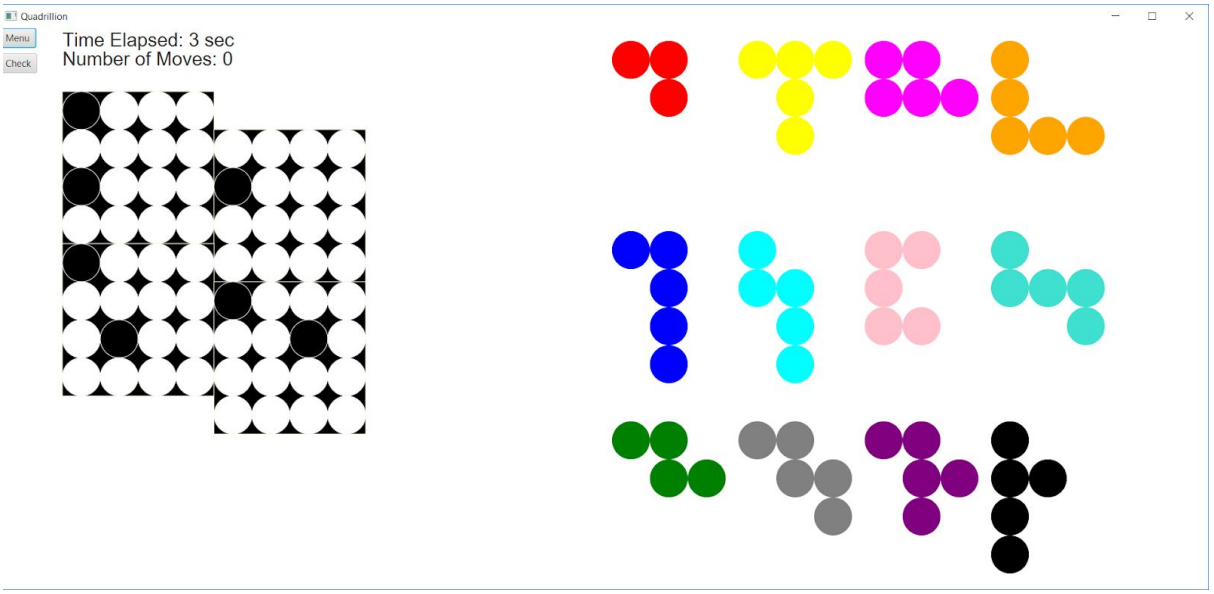

Figure 9: PlayRanked screen