

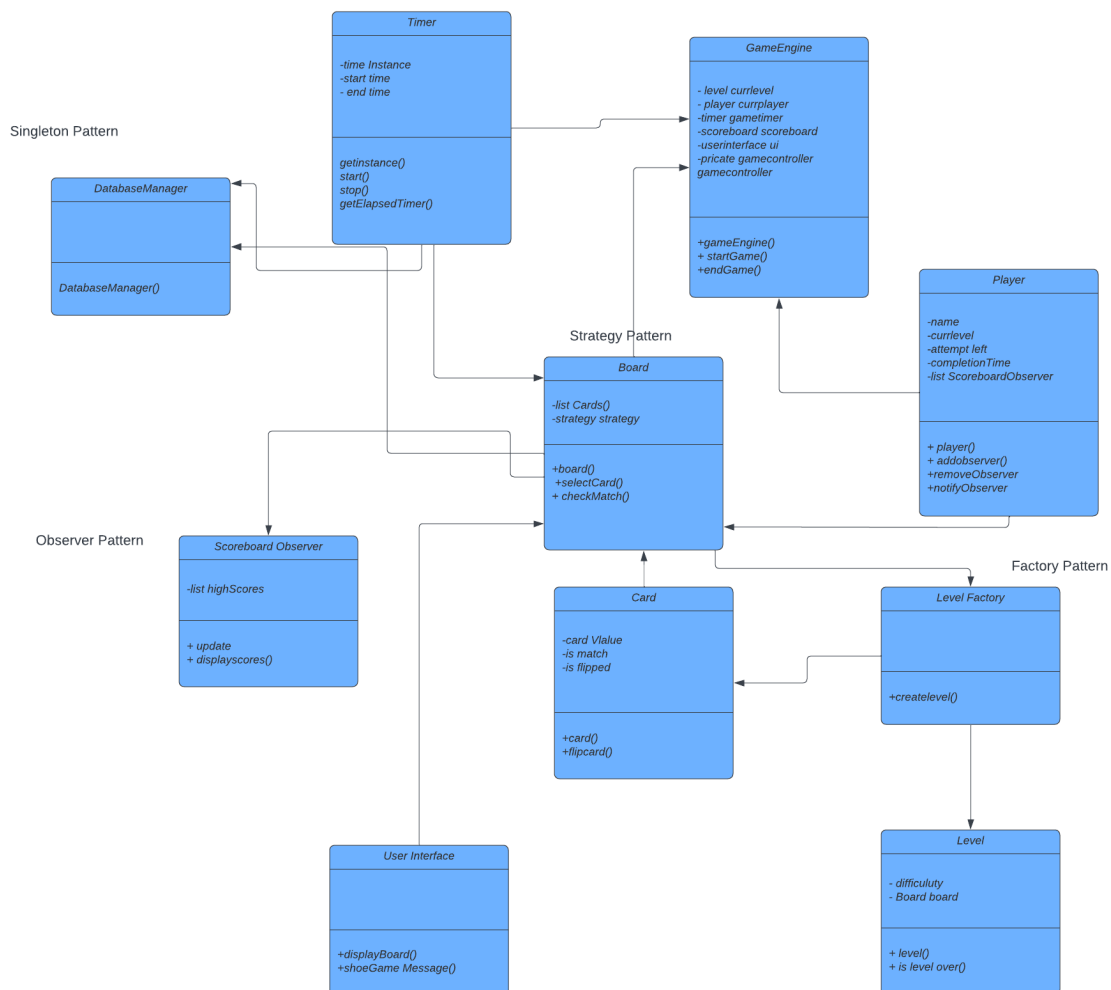
Project 7 Deliverables

1. Name of project and names of all team members
 - a. Owen Crenshaw and Franco Ruscitti
 - b. Memory Match Master
2. Final State of System Statement
 - a. A paragraph on the final state of your system: what features were implemented, what features were not and why, what changed from Project 5 and 6
 - i. Our java matching game is in its final stage of development, with several key features already successfully implemented and functioning while there are still some that are not. The User Interface (UI) is functional and displays our game board with a set of cards and a fully functional timer which works as expected on click, the cards flip when prompted (clicked), and the cards show their hidden value when flipped. In developing our project we have made significant progress, yet there are a few features which are still in the refinement stage. These features include the scoreboard, the functionality of flipping the cards back over when there is no match, and the activation of the observer for state reporting and some other patterns. The scoreboard was not implemented because we had major issues with reading and writing to a file which caused significant loss of time and energy. The flipping of the cards was not implemented because we were having major issues with our UI inside our code structure which made it very difficult to get the visual functionality working. This was the challenge that influenced the pace of our development of the game, trying to perfect the card flipping mechanism in our code and UI. Here we encountered a critical roadblock in this area of our development which required our focused attention as it was necessary for the games overall functionality. We could not get the other functionalities of the code up and running without the basic foundational functions working. When working on the ARCANE project, we quickly rushed into all the steps without having the basics working. This approach really handicapped us in the future weeks. For this project we wanted the basics working before getting ahead of ourselves and having to rework code. The evolution from Project 5 and 6 to our current iteration has been marked by significant advancements, particularly in two key areas. First we developed a more robust

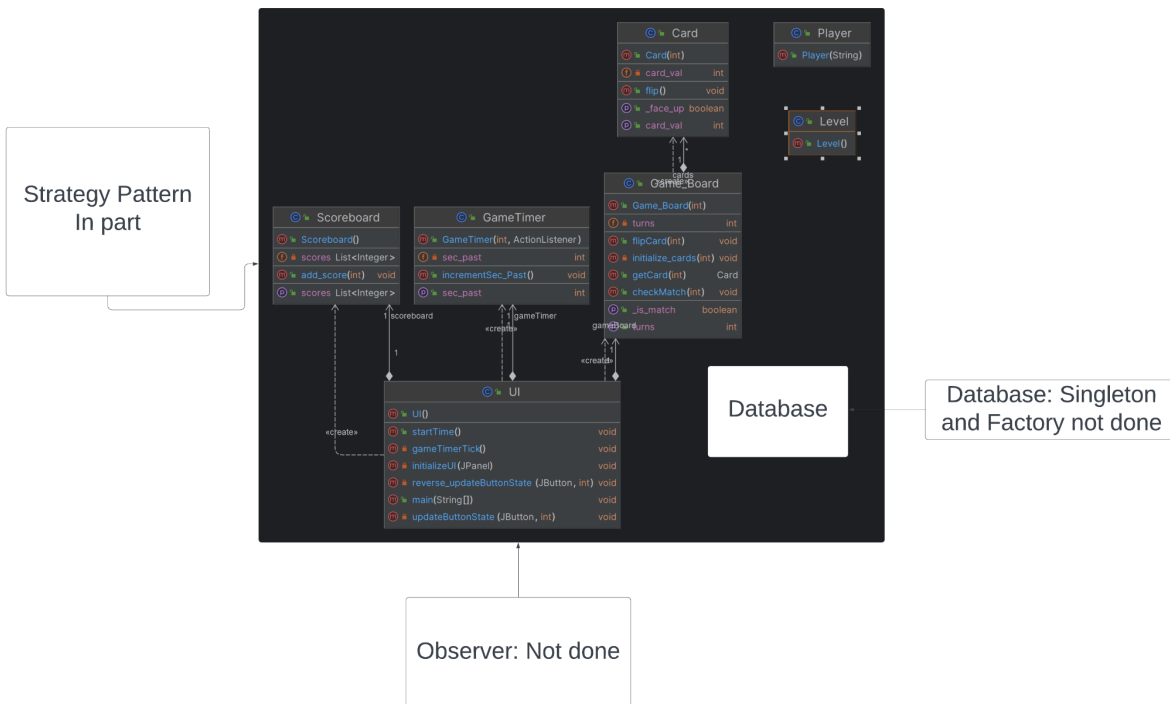
framework for managing the game's card flipping mechanics specifically for trying to flip the cards back to their original state when there is no match. We worked very extensively on this so that is one of the main changes. Secondly, another big improvement is the integration of a functional timer that activates when the first card on the UI is clicked in the user interface. This feature added a more dynamic trait to the gameplay by further engaging players with a sense of urgency. This implementation of the timer also showcases our progress in creating a more interactive and responsive UI effectively increasing the overall quality of the game.

3. Final Class Diagram and Comparison Statement

UML for Project 5



UML for Project 7



Key Changes in system since Project 5 and 6:

The major changes from project 5 and 6 to now in project 7 are as follows. We got rid of the game engine class and pushed that functionality into the UI class. All of the classes were reworked into different attributes and methods. We also planned for our database to be a .txt file instead of an external database system. This simplified the project scope and reduced work. As we said above, in the previous paragraph we have a lot more work completed since project 5 and 6 such as a functional timer, UI, and card functionality. Further, the UI structural change was the development of a fully functional feature which now effectively supports the visual aspects of card flipping and matching as well as a display of a timer indicating how long the user takes to match the cards. Another change we made in our UML was the removal of the level class. Initially, our design was to include multiple levels each with its own set of cards increasing in difficulty to match. However, in project 7 assignment we decided to effectively streamline our approach by focusing on a single level. This allowed us to focus more on the mechanics of the game as it pertained to the only level. Also we decided to move the strategy pattern from the gameboard to the scoreboard class. This is because we can have different scores depending on how many matches in a row the user gets. This directly relates to the strategy pattern.

4.

a. 3rd Party Code VS Original Code Statement

- i. All of the code in this project is entirely originally developed exclusively by franco and owen
 - b. Tutorial for SWING
 - i. <https://docs.oracle.com/javase/tutorial/uiswing/>
 - ii. [Java Swing Tutorial - javatpoint](#)
 - iii. <https://www.javatpoint.com/java-awt>
 - c. We used these main imports in our project
 - i. `import javax.swing.*;`
 - ii. `import java.awt.*;`
 - iii. `import java.awt.event.ActionEvent;`
 - iv. `import java.awt.event.ActionListener;`
 - d. We used the following frameworks
 - i. We used IntelliJ for our code environment
 - e. We used lucid Chart for assistance in visualizing the desired functionality of the game
5. Statement on the OOAD process for your overall Semester Project
- a. We had a negative experience as we struggled with working with flipping the cards using JButton
 - b. We had a positive experience getting the timer working as it started when we clicked on the first card
 - c. We had a negative experience and struggled with fitting the patterns of this class into our project. It felt forced to try and put these patterns in.