**Progress Report**

**- Increment 2 -**

**Group Phaze 5**

# **1)** **Team Members**

Faith Miller:

FSUID: fmm15c

GitHub: maithfiller

Mackenzie Knight:

FSUID: mlk17

GitHub: m-knight99

Eduardo Antonini:

FSUID: ea17h

GitHub: edcorant

Katie Rombeiro:

FSUID: klr16f

GitHub: klr16f

Tristan Garcia:

FSUID: tjg16d

GitHub: tjgarcia0427

**2)** **Project Title and Description**

*Phaze 5* is inspired by the popular card game *Phase 10*, but with an online gameplay. Like in *Phase 10*, players in *Phaze 5* will try to complete a given number of phases until someone completes the last phase. Whoever finishes the last phase first will be the winner — unless two or more players finish the last phase in the same round, in which case the player with the smallest amount of points becomes the winner.

**3)** **Accomplishments and overall project status during this increment**

The most significant accomplishment during this increment is the completion of the backend game implementation for the web app. The game implementation is currently independent from the frontend. The game implementation uses prompt() and alert() at this moment for testing purposes instead of pulling input and outputting to the web app. Testing of the game implementation was completed using CodePen and we were able to successfully get through a full game with 2 phases.

The frontend accomplishments during this increment include taking in the number of players and outputting text fields for the player usernames based on the number inputted by the user in a modal. Additionally, the homepage design has been planned out, which includes a series of modals that pop up to grab game choices from each player. The creation of modals has also been started within the frontend.

In comparison to the initial scope and functionality of the project, our project has been changed in a few ways. Firstly, the functionality has been slightly scaled back, as our game implementation has gone from having 5 phases to having 2 phases. This decision was made in order to reduce the amount of time taken to test our project, as it is predicted to take about an hour and a half for 3-5 players to finish the game with 5 phases. Additionally, we are no longer using routing to go from one page to the next. We are now using a link in order to navigate from the homepage to the how to play page.

**4)** **Challenges, changes in the plan and scope of the project and things that went wrong during this increment**

One challenge we all faced as a group was when we all either went home, or physically all separated, because of COVID-19, causing the in-person communication aspect within our group to be severed. Another challenge that we are coming to face is the connection aspect between the front-end and the back-end (game implementation) of our code. All of us are fairly new to Atom, along with the languages React and JavaScript. We are still trying to figure out and understand how to send user input through the website to our game implementation, and send output related to the game back to the website. We plan to do more research and ask questions in order to overcome this. Another challenge we are facing is how long, time wise, this game itself takes to play, therefore taking hours of our time just to test our code. We are currently in the process of fixing this issue by either coding less phases of the game or by using flags within our testing to not have our time wasted. Another challenge that Katie and Faith faced, who focused on the front end implementation, was the struggle to transfer state variables between components. In the end, we found a way around it by keeping and merging all of the information into one file instead of using multiple files for the Homepage and PlayGame page.

**5)** **Team Member Contribution for this increment**

**Faith Miller:**

1. Faith wrote the section “Plans for the Next Increment” on the Progress Report.
2. Faith contributed to the “Functional Requirements” and “Non-functional Requirements” sections of the Requirements and Design Document.
3. Faith wrote the section “Non-Execution-based Testing” on the Implementation and Testing Document.
4. Faith contributed to the code that takes in the number of players and outputs the number of player usernames that is needed depending on the number inputted by the user. Faith also contributed with trying to figure out the process to pass state variables between components. Faith contributed to the code of the alteration of the HomePage, creating the first “Play Game” modal gathering the player data. Faith also added to the modal css and did minor error checking for the UI.
5. Faith contributed to the Video portion of the project by actually filming the Presentation in the video and by filming the front-end demo of the project.

**Mackenzie Knight:**

1. Mackenzie wrote the “Accomplishments and overall project status during this increment” section of the Progress Report.
2. Mackenzie contributed to the “Functional Requirements” and “Non-functional Requirements” sections of the Requirements and Design Document.
3. Mackenzie wrote the “Execution-based Functional Testing” section of the Implementation and Testing Document.
4. Mackenzie wrote and tested the showBoards(), showHand(), dropCard(index), clearBoard(), clearHand(), board1(), board2(), and makeSkipTrue() functions in the Player class of the game implementation. Mackenzie contributed to the error checking for an empty draw pile within the draw() function. Mackenzie contributed to the isASet(arr1, arr2) function in the PlayGame class of the game implementation. Mackenzie also contributed to the game implementation testing, which included contributing to various small changes within the PlayGame class.
5. Mackenzie wrote the Plans for Increment 3 slide of the presentation designed for the video. Mackenzie also filmed the backend demo for the video.

**Tristan Garcia**

1. Tristan contributed w/ Katie on the “Challenges, changes in the plan and scope of the project and things that went wrong during this increment” section of the progress report.
2. Tristan contributed to the “Use-Case Diagram” section of the Software Requirements and Design document
3. Tristan wrote the “Platforms, APIs, Databases, and other technologies used” section of the Software Implementation and Testing document.
4. Tristan contributed to the moveCardsToBoard1() and moveCardsToBoard2() functions in the Player class. Tristan wrote and contributed to all functionality within the PlayGame class of the game implementation. Tristan also contributed to the game implementation testing, which included making various small changes within the PlayGame class.
5. Tristan wrote the “Change in Scope” slide of the video presentation.

**Katie Rombeiro**

1. Katie contributed w/ Tristan on the “Challenges, changes in the plan and scope of the project and things that went wrong during this increment” part of the progress report
2. Katie wrote the “Overview” and the “Assumptions and Dependencies” sections of the Software Requirements and Design document
3. Katie wrote the “Execution-based Non-Functional Testing” section in the implementation and testing document
4. Katie contributed to the code that takes in the number of players and outputs the number of player usernames that is needed depending on the number inputted by the user. Katie also contributed with trying to figure out the process to pass state variables between components. Katie contributed to the code of the alteration of the HomePage, creating the first modal gathering the player data. Katie also added to the modal CSS and did minor error checking for the UI.
5. Katie contributed to the general overview of the video presentation

**Eduardo Antonini**

1. Eduardo composed the “Project Title and Description” section of the Progress Report.
2. Eduardo handled the research and writing of the “Class Diagrams” and “Operating Environment” sections of the Software Requirements and Design document.
3. Eduardo authored the “Programming Languages” section of the Software Implementation and Testing document.
4. Eduardo wrote several of the Player class member functions in their entirety, such as addPoints( ), handSize( ), addPhase( ), points( ), sortHand( ) and isBoardEmpty( ). Still in the Player class, Eduardo worked with his group mates to write some functions collectively — his contributions include, but are not limited to, most lines of the draw( ), moveCardsToBoard1( ) and moveCardsToBoard2( ) functions, as well as protocoling and coding a solution to the edge case where a player attempts to draw a card from an empty deck in the draw( ) function.
5. Eduardo wrote the State of the Project slide of the video presentation.

**6)** **Plans for the next increment**

In the next increment, our group is going to continue connecting the front end of the Play Game page to the Game Implementation aspect of the program. We will also be continuing testing the entire project to make sure everything is always working even if some of us are fixing or changing code. In addition, we will also be finalizing the game because there is less than a month left until the project is due. We will be conducting code reviews, so that each person gets their code checked by another person to check for any mistakes or issues. We will also be figuring out if we have extra time when finished and possibly set up the server, so that the game doesn’t have to just be played on a single computer on localhost. Also, if extra time allows, our group will try to learn how to add in visually appealing animations to the cards on the user interface.

**7)** **Link to video**

<https://www.youtube.com/watch?v=lTYkxQruvTE>