CSCI 5448 - Object Oriented Analysis and Design
Project 6 - Interim Report

1. **Status Summary:**

Team Members:

Prashanth Vamanan Srinivasan
Ashwin Viswamithiran

Project Title: ASCII Attack

Work Done:

In the first sprint for the past two weeks, we designed and implemented the core features and elements of our 2D game. We have created well-structured classes with specific responsibilities that handle various interactions within the game.

The game manager class is the central piece of our application which delegates responsibilities such as creating blocks to the block manager, setting up ui elements to the UI manager, and managing levels to the level manager. We also have modeled the buttons and the blocks used in the game as their own classes to be reused across the application wherever needed. Apart from this, we also have separate classes for viewing game rules and a welcome screen that contains various interactive buttons for the player.

Tasks Accomplished and breakdown of work:

- Loading screen with a delay of three seconds when the game is launched - Prashanth
- Welcome screen with background image, four interactable buttons (New Game, Rules, Leaderboard, Quit) - Ashwin
- Adding functionality to view the rules for the game and a back button to return to the welcome screen - Ashwin
- Loading the main game screen upon click of New Game button - Prashanth
- Spawning blocks at regular time intervals and stop spawning when the specific number of blocks for the level have been spawned - Prashanth

- Designing and placing the main game screen UI - Ashwin

Changes or Issues Encountered:

Our design has stayed pretty consistent from what we described in Project 5. We are actively using the class diagram as our reference point as we are building out the various features of our application. Since we spent a good amount of time on our design process detailing the various functional requirements, we created classes that help to achieve these requirements.

Though we have added a couple of more classes for viewing the rules and a generic button class to model the buttons throughout the various game screens, the rest of our design has not changed much and is helping us to build the various functionalities that support our application.

Since PyGame was a new library we haven't used before, we had a bit of a learning curve. We also needed to understand some game specific terminologies such as a game loop and how to manipulate the individual pixels for objects in our game. Further, we spent time on using a little bit of math calculations and formulas, to position and move objects within our game window.

Patterns:

During the design phase of our semester project, we planned to implement four design patterns namely singleton, factory, command and decorator. In the first sprint, we have implemented the singleton pattern.

Implementing the singleton pattern was beneficial to our application, since there are certain classes where we needed to ensure that only one instance of it was available throughout the entire game. These include Game Manager, UI Manager, Block Manager and Level Manager.

These classes drive other core features of our game and it is essential to ensure that only one instance of each of them is available at any point since these classes are memory-intensive as they contain references to other classes. The singleton pattern helps us to achieve this requirement.

Furthermore, in cases where we needed to paint the UI screen in a while loop, making that class as a singleton ensured that we are not creating multiple instances of it within a loop which can slow down the game play and consume more memory. In the next sprint, we plan to implement the remaining patterns which will help us to create code which is loosely coupled and easily extensible for future changes.

2. **Class Diagram:**

   The class diagram can be found here. The classes that have been implemented are color-coded with Green.

3. **Plan for Next Iteration:**

   At this point we have implemented the main features of the game. Since we have accomplished this, in the second sprint we will focus our attention on implementing the remaining three patterns as discussed above. We have already started leaderboard functionality as well and it is moving at a brisk pace.

   We believe that we are on track to complete the remainder of the project by dividing and working through the functionalities together as a team. Once the patterns are implemented, we need to implement the game over functionality, update the UI elements in the main game loop, load new levels and provide an option to restart the game.

   Apart from this, we also intend to add some final polish to our game through sound and visual effects. For the final project delivery, we envision delivering a fully working game supporting all the features we described during our design phase. Though the remaining features may seem a lot of work, we are confident that we can complete them for the final deliverable.