Work Log

This is an ongoing document to track our progress and meetings.

Week 1

Date:	Wednesday, 04/22/20	Monday, 04/27/20	Wednesday, 04/29/20	Friday, 05/01/20
Daily Objective:	**Planning** Brainstorm ideas for 2-D game and begin write-up assignment.	**Planning / Analysis / Design** Discuss game design and analysis of implementation. Discuss classes and variables needed.	**Implementation** Begin building the basic skeleton of the project to achieve a Minimum Viable Product.	**Continue Implementation** *Continue* building the basic skeleton of the project to achieve a Minimum Viable Product.
Xiaolei Jiang	● Research player vs player and computer vs player games. ● Add ideas to write up draft document.	● Create a list of possible variables and functions needed for board class and discuss with the group. ● Discuss design mockup with the group and make changes.	● Begin working on adding base of board: this includes importing javafx packages to add labels for text, create buttons, and create game board panels.	● Debug and Research JavaFX board panels to use buttons instead of panels for the players to place stones. ● Technical hurdle: Original Gomoku game uses a line grid and dots at the intersections. JavaFX cannot handle that so we have to use buttons instead. ● Improve function so the board

				will print out 25 x 25 grid of “buttons/ titles, because originally I added them one by one and it wasted time and memory.
Kaylyn Torres	• Create a Google Drive folder for group to share documents and collaborate. • Create a write-up document for group to add project ideas and collab on write up draft. • Add ideas to write up draft	• Proof-read and finalize write-up. • Create mock up designs of gomoku game boards that will display how the board will change after each turn. • Discuss design with the group and make changes. • Make a list of variables and classes needed for the player class.	• Begin working on player class by adding basic mutator and accessor methods. • Add javadoc syntax to player class.	• Finalize the Player class and share code with the group for review to make sure our variables class names match. • Add project objective, approach and priorities, and mock-up designs to the Design Approach document draft. • Create a while loop to handle the choices for game mode: pvp and pve.
Zhanyao Xu	• Research player vs player and computer vs player games. • Add ideas to write up a draft document. • Assign tentative tasks to Kaylyn, Xiaolei and myself based on strengths and interests.	• Create a list of variables and classes needed for AI class.	• Create boardrunner class and initiate discussion on what should be included in the class and how we can collaborate. • Begin working on AI class and a function that will make AI randomly set stones on the game board.	• Build PlayerAction interface class. • Debug AI class by researching JavaFX board panel mouse tracking because AI cannot click and set stones by himself.

Week 2

Date:	Saturday, 05/02/2020 Sunday, 05/03/2020	Monday, 05/04/20	Wednesday, 05/06/20	Thursday, 05/7/20	Friday, 05/08/20
Daily Objective:	**Continue Implementation** We have completed a MVP of our Gomoku Game. Our next goal is to	**Improve Board Game / Begin Testing** Tweak the style and design and ensure runners work as expected.	**Finalize Implementation and Continue Testing**	Begin working on text documents for project: JavaDoc, UML, Design Approach,	Finalize documents and project files.
Xiaolei Jiang	● Finalize board grid and research player mouse movement where track mouse movement and GUI. ● Add to the project approach to the Design Approach document draft.	● Make the menu on the board runner better by adding label after each player turn and show text of the player's position. ● Set final button styles for the boardrunner and debug code with Kaylyn.	● Finalize the for loops that display items to screen ● Make sure mouse events will click and track player 1 and player 2 and the following mouse event add for loops	● Add cover letter to design approach ● Add project planning and implementation to design approach	

Kaylyn **Torres**	• Add labels in board runner that display rules as text on screen. • Add a function that will handle the player vs player action when the tiles are clicked. • The function checks if the position x and y selected on the board is empty, if the condition is true, set the tile on and make sure it matches the player • Create abstract player class	• Review and debug my code with Xiaolei and Zhanyao to make sure it handles their buttons and AI objects correctly. • Add a label for when the position is empty, to alert the player to click another tile • Add a label to display winner to screen • Debug code in boardrunner for a couple* hours **million*	• Fix button sizes because they displayed differently on my mac then my group's windows 7 demo. Make sure the buttons are the same on each platform. • Add javadoc syntax to all classes except AI class	• Continue adding javadoc syntax • Make UML Diagrams for Board,Player and BoardRunner Class	• Make UML Diagrams for AI, HumanPlayer Interface class • Add analysis of P v P and P v E to design approach • Proof-read and edit design approach document
Zhanyao Xu	• Debug AI: When the AI put a stone, the color of the stone didn't show up • Continue working on AI functionality to make AI click buttons by itself. • Build a new button that gets a random x and y position param on the board. The button will be added to the top of the title on the board to appear that the AI clicked it. • Build function that checks wins in AI.	• Buff the AI because he is too dumb. Instead of having the AI select a random x and y position, AI will check it's surrounding: up, down, left,right before placing a tile.	• Finalize AI buff • Make sure the entire project is clean of errors and bugs • Upload final project to google drive		

Week 3

Date:	Sunday, 05/09/2020	Monday, 05/11/2020
Daily Objective	**Finalize Project**	**Finalize Project** Practice for Presentation
Xiaolei Jiang	● Prepare talking points for presentation and how to explain the code	● Meet with group on BlackBoard Collaborate to discuss presentation and assign tasks for who will discuss the components of code
Kaylyn Torres	● Clean up and format code ● Create Jar file ● Add all documents into a zip on drive ● Create mini presentation for project	● Meet with group on BlackBoard Collaborate to discuss presentation and assign tasks for who will discuss the components of code ● Share presentation with group and make changes
Zhanyao Xu	● Prepare talking points for presentation and how to explain the code	● Meet with group on BlackBoard Collaborate to discuss presentation and assign tasks for who will discuss the components of code