Game Logic and Mechanics Team Timeline:

Iteration 1 (Planning & Design)

• Feb 22 – Mar 7:

- **From the use cases written in Week 1, create a Use Case Diagram to show the interaction between the user (player) and the system (game logic).**
- **Deliverables: Use case diagram (for three selected games initially), Class structure diagrams for each of the games selected.**
- **Responsibilities: Team members should divide the task of defining game rules for different games.**

Iteration 2 (Development & Integration)

• P2:

- **Team drafts and signs their Accountability Contract.**
- **Submit the contract to the group Dropbox.**
- **Team works collaboratively, following the agreed-upon roles and guidelines.**
- **Regular check-ins help ensure members meet their obligations.**
- **Evaluate each member's performance and complete Individual Adjustment Sheet and submit to group Dropbox.**

• Mar 8 – Mar 14:

- **Milestone 1: Define core game mechanics and begin coding for Chess, Go, Connect4, and Tic Tac Toe.**
    - **Chess:**
        - **Task: Implement basic game logic for chess, focusing on piece movement (e.g., Pawns, Knights, etc.), including capturing and promotion mechanics.**
        - **Task: Implement win conditions, such as checkmate, stalemate, and draw (e.g., insufficient material, fifty-move rule).**
        - **Task: Code for castling and en passant moves.**
        - **Deliverable: Functional Chess game engine with basic mechanics, including unit tests for piece movements and captures.**

- **Go:**
    - **Task: Implement basic game logic for Go, including placing stones, capturing groups, and calculating territory.**
    - **Task: Implement rules for ko (capture-stalemate situation) and suicide.**
    - **Task: Develop the board state management and display logic for a Go game board.**
    - **Deliverable: Functional Go game engine with basic mechanics and unit tests for stone placement and capturing.**
- **Connect4:**
    - **Task: Implement Connect4's game logic, focusing on player turns, gravity-based piece dropping, and win detection (horizontal, vertical, and diagonal).**
    - **Task: Code for detecting full columns and preventing moves beyond the bottom row.**
    - **Deliverable: Fully functional Connect4 game engine with turn-based mechanics and win detection.**
- **Tic Tac Toe:**
    - **Task: Implement basic game logic for Tic Tac Toe, focusing on player moves, win detection (horizontal, vertical, diagonal), and game-ending conditions (draw or win).**
    - **Task: Code for handling input validation (preventing moves in already occupied spaces).**
    - **Deliverable: Functional Tic Tac Toe game engine with turn-based mechanics and win detection.**
- **Unit Testing:**
    - **Task: Create unit tests for all the basic game mechanics and functions.**
    - **Task: Test core functionality for two games (choose 2-3 games for initial testing).**

        - **Deliverable: Unit tests executed successfully with initial bug fixes, if applicable.**

- **Mar 15 – Mar 21:**
    - **Milestone 2: Integration of game logic into the matchmaking system and continued development.**
        - **Chess:**
            - **Task: Integrate Chess logic into the matchmaking system, ensuring players can be matched up for online play.**
            - **Task: Ensure that multiplayer Chess matches sync properly between players, reflecting the correct game state.**
            - **Deliverable: Multiplayer Chess functionality with working turn-based gameplay.**
        - **Go:**
            - **Task: Integrate Go logic into the matchmaking system and test multiplayer functionality.**
            - **Task: Ensure proper synchronization of game state between players (stone placements, captures, etc.).**
            - **Deliverable: Multiplayer Go functionality with game state synchronization.**
        - **Connect4:**
            - **Task: Integrate Connect4 logic into the matchmaking system and test multiplayer interaction.**
            - **Task: Ensure that game state syncs properly when players take turns.**
            - **Deliverable: Connect4 with multiplayer support and synchronized gameplay.**
        - **Tic Tac Toe:**
            - **Task: Integrate Tic Tac Toe logic into the matchmaking system and test for synchronization in multiplayer play.**

        - **Task: Ensure proper win detection and smooth transition from one player's turn to the next.**
        - **Deliverable: Multiplayer Tic Tac Toe with full synchronization and turn-based play.**

Iteration 3 (Testing & Finalization)

• All TAs and your instructor need to be added as Developers.

• Mar 22 – Mar 28:

- **Milestone 1: Completion of intensive testing and optimization.**
    - **Game Logic and Mechanics Team:**
        - **Task: Conduct intensive testing of all implemented games, especially multiplayer aspects (turn handling, game state synchronization, etc.).**
        - **Task: Create detailed bug reports, track issues, and perform performance optimizations.**
        - **Task: Verify that all game mechanics are functioning according to the design specifications.**
    - **Game Logic and Mechanics Team:**
        - **Task: Test the core mechanics of the games (e.g., Chess piece movement, Connect Four win detection) for accuracy and performance.**
        - **Task: Ensure mechanics function correctly across multiple game instances (local and multiplayer).**
- **Milestone 2: Optimization of code and finalization of game logic.**
    - **Game Logic and Mechanics Team:**
        - **Task: Fix bugs, optimize code for performance, and finalize the game logic for all games.**
        - **Task: Ensure that all win conditions, game states, and interactions work seamlessly.**
    - **Game Logic and Mechanics Team:**

        - **Task: Final round of testing for mechanical aspects, ensuring smooth game flow.**
        - **Task: Perform regression testing on previously fixed issues to ensure no new bugs arise.**

• Mar 29 – April 4:

- **Milestone 3: Final round of testing and integration.**
    - **Game Logic and Mechanics Team:**
        - **Task: Conduct final testing on game logic to ensure smooth operation with multiple players, including edge case scenarios.**
        - **Task: Implement final adjustments based on feedback from other teams and test cases.**
        - **Deliverable: Finalized game logic with all bugs fixed, including multiplayer synchronization issues.**
    - **Game Logic and Mechanics Team:**
        - **Task: Validate all mechanics again after final code adjustments.**
        - **Task: Perform multiplayer testing to ensure mechanics interact correctly in an online environment.**

• Deliverables:

- **gitlab_link.txt**
- **README.md**
- **git_log.csv**
- **Changes made: Summary of changes made after reviews from other groups.**
    - **Recommended: before/after sections of diagrams**
- **group_demo.mp4: Video demo submission. Zoom to record and edit. If D2L denies submission due to large size, upload to Yuja, rename to 'yuja_video_link.txt" in D2L.**
- **test_suite.html: Test cases.**

Post-Iteration (Final Review & Demo)

- **April 5 – April 11:**
    - **Task: Prepare for group demonstration video, showcasing all game logic implemented and tested.**
    - **Deliverables: Complete demo of the game's rules and multiplayer capabilities.**
    - **Responsibilities: Final review and testing before demo submission.**