**Team Project Sprint #1**

Instructions

Please read the instructions carefully. All members of your team should discuss the instructions together to ensure that everyone is on the same page.

**Objectives**

Create a brief project description, specify all requirements (i.e., all user stories and acceptance criteria) of the target software that allows a human player to play against either a human or a computer opponent, and implement the primitive functions (i.e., board object and visualization, and piece placement for both players). Each team should meet at least once a week. One meeting may serve multiple purposes in the Scrum process.

**Deliverables and Grading Policy**

1. Project Report (**20 points**)

The project report should include the following sections:

* 1. Project description (micro-charter), which should result from group discussion **(1 point)**.
  2. User stories using the template discussed in class. **(2 points)**

Provide a complete list of user stories and estimated efforts for the target software that allows a human player to play against either a human or a computer opponent. The planning poker approach to the effort estimation should be performed by the entire team.

* 1. Acceptance criteria using the template discussed in class. **(10 points)**

Provide complete acceptance criteria for each of the user stories related to all the functions for a human player to play a complete game either a human or a computer opponent. Note that, although some of the user stories will be implemented in the future sprints, their acceptance criteria need to be defined in the first sprint. You may continue to improve the user stories and acceptance criteria in the next sprint.

* 1. Implementation tasks **(5 points)**

Describe the production code, automated test code or manual test case for each user story and acceptance criterion related to the implementation of the primitive functions, i.e., board object and visualization, and piece placement for both players. For each acceptance criterion of every user story for the primitive functions, you need to implement at least one test (either test code or manual test case).

* 1. Minutes of ALL meetings, including, but not limited to: project/sprint planning meeting, stand-up meeting, backlog grooming, retrospective meeting, and pair programming (or development) session. **(2 points)**
  2. A table of buddy ratings. Individual members may email their buddy ratings to the instructor or teaching assistant.

Each team only needs to submit one report. For an individual member to receive the credit for this part of the project, the team’s project report must include explicit evidence of his/her contribution (e.g., his/her name is listed as a developer).

2. Demonstration **(5 points)**

Submit a 5-minute video, clearly demonstrating that:

1. your project has implemented the working software for the primitive functions, i.e., board object and visualization, and piece placement for both players.
2. for each acceptance criterion of every user story for the primitive functions, your project has implemented either an automated test method or performed an acceptance test manually.
3. your project has some unique features or enhancements (optional).

Grading of the demonstration is based on completion of the required functions (**2 points**), and overall presentation (**3 points**) using the following evaluation rubric:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Poor | Fair | Good | Very Good | Excellent |
| Was the demonstration logically organized |  |  |  |  |  |
| Were points made clearly and concisely |  |  |  |  |  |
| Were the instructor’s questions, if any, answered satisfactorily |  |  |  |  |  |

**Team Project Sprint #1**

Report Template

Team Name: Forgetful Wanderers

Team Members: Zach Gharst, Thomas Yang, Ken Dozier, Joe Soria, Thomas Tran

1. **Project Micro-Charter (no more than one page)**

Our project name is “M-3” which stands for “Milling Milking Milkmen”. We use software methodologies such as Scrum, testing, and version control to create and deliver finished products in a reasonable timeline. Through this project, we aim to create a fun and unique take on the well-known game “Nine Men’s Morris” as a product; players will have the choice of playing against a friend on the same device or against a computer opponent using a point-and-click (or touch) interface. This entertainment product will get exposure in the games marketplace and create an online presence for the project customer. Our project customer could release this game on the mobile, console, or PC marketplaces to attract an online customer base. We will measure the success of the product through metrics such as daily active users, daily active minutes, releasing-early-and-often, and customer engagement. We have several milestones: (1) functional primitive functions, (2) usable product with all logic functions, and (3) polish and unique-ness. To achieve these milestones, we will have to avoid certain risks. There is a risk that the value of our product backlog becomes too low to justify continuing with our project. There is also a risk that the lower experience of our team could result in a slower timeline. We can mitigate these risks by are usage of software methodologies.

Authors: Thomas Tran, Zach Gharst

1. **User Stories**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **User Story Name** | **User Story Description** | **Priority** | **Estimated effort (hours)** | **Actual effort (if completed)** | **Status (completed, toDo, inProgress)** | **Developer names** |
| 1 | Game menu | \*\*As a player, I need a menu that allows me to start a new game or exit the program | Low | 1 |  | To do | Thomas Tran |
| 2 | Opponent | As a player, I need a choice between a computer opponent or another player. | Low | 5 |  | To do | TBD |
| 3 | Player ordering | One player will be assigned white pieces and the other player will be assigned black pieces. | Med | 1 |  | In Progress | Zach Gharst |
| 4 | Starting Board | As a player, I need an empty board and an opponent to start playing the game. | Very High | 2 | 2 | Completed | Zach Gharst |
| 5 | Player’s first turn | To play the game, we need to determine which player goes first (first player is white pieces). | Med | 1 | 1 | Completed | Zach Gharst |
| 6 | Phase 1 (placement) | As players, we need to alternate placing 9 pieces (each) on the board per player. | Very High | 3 | 2 | In Progress | Zach Gharst, Thomas Tran |
| 7 | Valid placement of piece | To make a move, as a player, I need a board that indicates which spots are open and available to click on. | Very High | 3 | 1 | In Progress | Ken Dozier,  Joe Soria |
| 8 | Mill Rule | As a player, when I have a mill (three pieces in a row), I can eliminate an opponent’s piece that is not in a mill unless the opponent only has pieces that are part of a mill. | High | 3 |  | In Progress | Zach Gharst,  Thomas Tran |
| 9 | Phase 2 (Play) | After all pieces have been placed, the board state should switch to the second (play) phase. | High | 1 |  | To do | Ken Dozier, Joe Soria |
| 10 | Movement | During phase two, if it is my turn, I want to select one of my pieces and move it to an adjacent vacant spot. (then #18 check for mill) | High | 3 |  | To do | Thomas Yang |
| 11 | Phase 3 (flying) | Once a player has three pieces left, that player enters phase 3 and can fly (move 1 piece per turn to any open spaces on the board). | High | 3 |  | To do | Thomas Yang |
| 12 | Exit the game | As a player, I want an option to forfeit the game early (forfeit button) | Low | 2 |  | To do | Ken Dozier,  Joe Soria |
| 13 | UI turn check | As a player, I would like an indicator of whose turn it is currently. | Low | 1 |  | To do | Thomas Tran |
| 14 | Score display | As a player, I would like to know what my current score is. | Low | 1 |  | To do | Thomas Tran |
| 15 | Undo button | As a player, I want an undo button to go back one turn (AI only). | Low | 2 |  | To do | Joe Soria |
| 16 | Board changes | As a player, I want different boards to play on (different board design) | Very Low | 4 |  | To do | Joe Soria |
| 17 | Request draw | As a player, I would want a to request a “draw” when pieces are on stalemate condition. | Very Low | 1 |  | To do | Joe Soria |
| 18 | Reset | As a player, I would like to easily reset the board and start a new game. | Med | 1 | 0.5 | Completed | Zach Gharst |
| 19 | History | As a player, I would like a tab or section to show recent moves made | Very Low | 2 |  | To do | Zach Gharst |
| 20 | UI polish | As a player, I would like a “highlight”/glowing light for vacant spaces. | Low | 1 |  | To do | Thomas Tran |
| 21 | Win Condition 1 | After each player’s turn, if the opponent has 2 pieces remaining, the player wins. | High | 2 |  | To do | Zach Gharst |
| 22 | Win Condition 2 | After each player’s turn, if the opponent cannot make a valid move, the player wins | High | 2 |  | To do | TBD |

1. **Acceptance Criteria (AC)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Story ID and Name** | **AC**  **ID** | **Description of Acceptance Criterion** | **Status (completed, toDo, inProgress)** | **Developer Names** |
| 1 Game Menu | 1.1 | GIVEN the program is initiated to run THEN a menu should open AND have a selection to exit or enter game. | To do | Thomas Tran |
|  | 1.2 | Given the menu has start game button, When the start button is pressed, Then the menu will proceed to choosing and human or computer opponent. | To do | Thomas Tran |
| 2 Opponent | 2.1 | GIVEN the program has initiated THEN the menu will ask for an enemy option AND I can select whether to play human or computer. | To do | Thomas Tran |
|  | 2.2 | GIVEN the user made a choice for compute opponent THEN another menu should open for computer difficulty AND have a selection for “EASY, INTERMEDIATE, HARD” | To do | Thomas Tran |
| 3 Player ordering | 3.1 | GIVEN that there’s only 2 valid players AND a white piece always goes first THEN a random factor will decide which color the players will be. | In Progress | Zach Gharst, Ken Dozier |
|  | 3.2 | GIVEN that players have been assigned a color WHEN the game has started THEN no two player should be assigned the same color. | In Progress | Zach Gharst, Ken Dozier |
|  | 3.3 | GIVEN that the players are chosen randomly to be white WHEN the game has started THEN there should be an equal chance for either player to be assigned white. | In Progress | Zach Gharst, Ken Dozier |
| 4 Starting Board | 4.1 | GIVEN that the game has started THEN the board should be empty AND appropriate players with selected color is ready. | Completed | Zach Gharst, Ken Dozier |
| 5 Player’s first turn | 5.1 | GIVEN that the game has started, WHEN the board is ready, THEN an it is white’s turn to place a piece. | Completed | Zach Gharst |
|  | 5.2 | Given that the game has started, When the player has placed a piece, the other player will have a turn to place a piece | Complete | Zach Gharst |
| 6 Phase 1 (placement) | 6.1 | GIVEN that the game has started WHEN a player makes a move THEN only the appropriate player’s turn can put a piece in the board. | Completed | Zach Gharst, Thomas Tran |
|  | 6.2 | GIVEN that a player clicks on the board WHEN it is not that player’s turn THEN it should ask them to wait for their turn. | To do | Zach Gharst, Thomas Tran |
| 7 Valid placements of piece | 7.1 | GIVEN that the player clicks on a vacant space WHEN it is that player’s turn AND it is phase 1 THEN a man should be removed from their pool AND placed on the vacant space. | Completed | Ken Dozier |
|  | 7.2 | GIVEN that the player clicks on an occupied space WHEN it is that player’s turn AND it is phase 1 THEN the player should be alerted that they must place a man on a vacant space. | In Progress | Ken Dozier |
|  | 7.3 | GIVEN that the player clicks on a vacant space WHEN it is that player’s turn AND it is phase 2 THEN the player should be alerted that they are out of unplaced men AND should instead click on a man to move. | To Do | Ken Dozier,  Joe Soria |
| 8 Mill Rule | 8.1 | GIVEN that a player completes a mill (three men in a continuous vertical or horizontal line) WHEN they have completed a vital move THEN they should be presented with the option to click on a piece to be eliminated. | In Progress | Zach Gharst,  Thomas Tran |
|  | 8.2 | GIVEN that a player has clicked on an opposing man in a mill WHEN they have formed a mill THEN the game should check to see if there are any men not in a mill that can be removed first. | In Progress | Zach Gharst, Thomas Tran |
|  | 8.3 | GIVEN that a player has clicked on an opposing man in a mill WHEN they have formed a mill AND all of the opposing men are in a mill THEN that man should be removed from the board. | In Progress | Zach Gharst, Thomas Tran |
| 9 Phase 2 (Play) | 9.1 | GIVEN that the black player has ran out of pieces WHEN it is first phase THEN second phase should be enabled. | Completed | Ken Dozier |
| 10 Movement | 10.1 | GIVEN that a player makes a valid move WHEN it is second phase and the player has finished their move THEN that players turn is ended and the other player’s turn starts. | To do | Thomas Yang |
|  | 10.2 | GIVEN that a player makes a valid move WHEN the player moves a piece to an empty intersection space THEN the piece will move to that empty intersection space. | To do | Thomas Yang, Joe Soria |
|  | 10.3 | GIVEN that a player makes an invalid move WHEN the player tries to make that invalid move the game will say no, stop, don’t do that (not literally) THEN the player will continue to make that invalid move until they give up and make a valid move. | To do | Thomas Yang, Zach Gharst |
| 11 Phase 3 (flying) | 11.1 | GIVEN a player has only three men remaining, AND desperate measures are called for. THEN player's men are allowed to 'fly' to any vacant cell, not just adjacent ones. | To do | Thomas Yang |
|  | 11.2 | GIVEN if one player is down to three men AND the other player still has more than three THEN only the player with three men is allowed to fly. | To do | Thomas Yang |
| 12 Exit the game | 12.1 | GIVEN a player wins, if his opponent AND cannot move, or is down to two men THEN the game announces winner. | To do | Ken Dozier,  Joe Soria |
|  | 12.2 | GIVEN that both players are down to three men AND neither player can capture anything in specific set of moves THEN the game ends in draw. | To do | Ken Dozier |
| 13 UI turn check | 13.1 | GIVEN that the player’s turn THEN an indicator should alert player for turn AND initiate action. | To do | Joe Soria |
| 14 Score Display | 14.1 | GIVEN that a piece is eliminated THEN a score system should be shown AND inform who’s winning | To do | Joe Soria |
|  | 14.2 | GIVEN a tally of remaining pieces WHEN a player makes a move AND the number of pieces remaining on the bag used THEN it should prompt how many pieces I have left. | Completed | Zach Gharst |
|  | 14.3 | GIVEN that no pieces are being eliminated WHEN a player makes a move THEN the score should not change. | To do |  |
| 15 Undo (Cheaters) Button | 15.1 | GIVEN that a user is playing against the computer WHEN user places regrets placing a piece THEN user should have an undo button to make better decisions in life. | To do | Joe Soria |
| 16 Board changes | 16.1 | GIVEN that a user is tired of playing on same boring board THEN user should have other selection board theme AND it changes the board. | To do | Joe Soria |
| 17 Request a draw | 17.1 | GIVEN that both players are down to three men AND neither player can capture anything in specific set of moves THEN the game ends in draw. | To do | TBD |
| 18 Reset | 18.1 | GIVEN that a player wants to reset the game WHEN the player is playing against a computer THEN the player should be able to press a button AND the game restarts. | Completed | Zach Gharst |
|  | 18.2 | GIVEN that a player wants to reset the game WHEN the player is playing multiplayer THEN the game should ask him to forfeit first. | To do | Zach Gharst |
|  | 18.3 | GIVEN that a player does not wish to reset the game WHEN they are playing THEN the game should retain the state of the game and not reset. | In progress | Zach Gharst |
|  | 18.4 | Given that the reset button is pressed, When all players agree, Then the board should reset to be empty. | In progress | Zach Gharst, Th |
| 19 History | 19.1 | GIVEN that a player makes a valid move WHEN it is their turn THEN that turn(move) should be shown in the history tab AND recent moves is displayed. | To do | Zach Gharst |
|  | 19.2 | GIVEN that a player does an undo WHEN the game is going THEN the history should remove undone actions. | To do | Zach Gharst |
|  | 19.3 | GIVEN that a player makes an invalid move WHEN it is their turn THEN the history should not add that move to the history tab. | To do | Zach Gharst |
|  | 19.4 | GIVEN that a player makes an invalid move WHEN it is not their turn THEN the history should not add that move to the history tab. | To do | Zach Gharst |
| 20 UI polish | 20.1 | GIVEN that a player’s turn with highlighted (vacant) spaces WHEN a player attempts to place a piece on unhighlighted space THEN player should be notified for illegal move or nothing happens. | To do | Joe Soria |
| 21 Win Condition (1) | 21.1 | GIVEN that a player has fewer than two pieces remaining WHEN a player makes a mill THEN the game ends and the other player wins. | To do | Zach Gharst |
|  | 21.2 | GIVEN that there are more than two pieces for both players WHEN a turn ends THEN the game should continue and not end. | To do | Zach Gharst |
|  | 21.3 | GIVEN that a player has fewer than two pieces remaining WHEN a player makes a mill THEN the game should ask a player what they want to do: new game or quit. | To do | Zach Gharst |
| 22 Win Condition (2) | 22.1 | GIVEN that the player’s turn ends, WHEN the game checks if the opponent has 2 pieces remaining and/or cannot make any more valid moves THEN the game ends and the player wins. | To do | TBD |
|  | 22.2 | GIVEN that the player can make a valid move WHEN their turn begins THEN the game should not end. | To do | TBD |

1. **Implementation Tasks**

Summary of production code

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Story ID and Name** | **AC ID** | **Class Name(s)** | **Method Name(s)** | **Developer Name(s)** | **Status** | **Notes (optional)** |
| 4 | 4.1 | Board/BoardManager | Start(), InitGame(), CreateIntersections() | Zach Gharst, Ken Dozier | Completed |  |
| 5 | 5.1 | BoardManager | InitGame() | Zach Gharst | Completed |  |
| 6 | 6.1, 6.2 | BoardManager/Intersection | Phase1(), OnMouseDown() | Zach Gharst, Thomas Tran | Completed | Currently showing remaining pieces in plaintext |
| 18 | 18.1 | BoardManager | ResetBoard() | Zach Gharst | Completed | Currently the button R is to reset; there could be a button in the future if requested. |
| 10 | 10.110.210.3 | BoardManger/Intersection | Phase2(), PieceMovement() | Thomas Yang, Zach Gharst | To do |  |
| 11 | 11.111.2 | BoardManger/Intersection | Phase3(), Flying() | Thomas Yang | To do |  |
| 13 UI turn check | 13.1 | TextManager | Update() | Zach Gharst | In Progress | Will be added upon |

Summary of automated test code (directly corresponding to some acceptance criteria)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Story ID and Name** | **Acceptance Criterion ID** | **Class Name (s) of the Test Code** | **Method Name(s) of the Test Code** | **Description of the Test Case (input & expected output)** | **Status** | **Developer Name(s)** |
|  |  |  |  |  |  |  |

Summary of manual test cases (directly corresponding to some acceptance criteria)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Story ID and Name** | **Acceptance Criterion ID** | **Test Case Input** | **Test Oracle (Expected Output)** | **Status** | **Notes** | **Developer Name(s)** |
| 4 | 4.1 | Start program | The board sprite, empty intersection points, and nine pieces available for each player. | Completed |  | Zach Gharst |
| 5 | 5.1 | Start program | By default, as a business decision, the white pieces always go first. If a different player wants to go first, they should choose white pieces. | Completed | Shows in plaintext on the left the current player’s turn. | Zach Gharst |
| 6 | 6.1 | Mouse click on vacant intersection points while pieces are waiting to be played and it is that player’s turn | A man is added as appropriate | Completed |  | Zach Gharst |
| 6 | 6.2 | Mouse click on vacant intersection points while pieces are waiting to be played and it is that player’s turn | Men are no longer allowed to be clicked and phase 2 is initiated. | Completed | Phase 2 is initiated, but phase 2 itself isn’t implemented. | Zach Gharst |
| 7 | 7.1 | Mouse click on vacant intersection points while pieces are waiting to be played, and it is that player’s turn | A man is placed and the remaining pieces is reduced by 1. | Completed |  | Zach Gharst |
| 18 | 18.1 | Keyboard press R | All men are removed from the board, it is now white’s turn, both players have 9 pieces again, and it’s currently phase 1. | Completed; Found Bug | Bug found at end of sprint; bugfix at start of sprint 2. | Zach Gharst |
|  | 18.4 | Placing pieces after resetting. | After the board resets, every cell becomes vacant. | In Progress  Bug found. | Cannot place men after resetting board | Thomas Tran |
| 10 | 10.1 | Player makes a valid move, then the player turn will end | Player’s piece will move to a valid intersection point and then the player’s turn will end | To do |  | Thomas Yang |
| 10 | 10.2 | Mouse click to move to an empty adjacent intersection point on the board, and it is that player’s turn | The piece will move to that empty adjacent intersection point on the board, and it’s currently phase 2 | To do |  | Thomas Yang |
| 10 | 10.3 | Mouse click to move a piece to an invalid intersection point will not work | The piece will not move to that invalid intersection point and the player will have to pick a valid intersection point | Completed | Sorta done, it’s only a small implementation that needs to be done. | Thomas Yang, Zach Gharst |
| 11 | 11.1 | When player has only three pieces remaining, phase three will be enabled for player | The player with flying enabled will be able to move their piece to any empty intersection point on the board, and it’s currently phase 3 for the player | To do |  | Thomas Yang |
| 11 | 11.2 | If a player has three pieces and the opponent has more than three pieces flying is enabled for only the player with three pieces. | Player with three pieces will have flying enabled, opponent with more than three pieces will not have flying enabled | To do |  | Thomas Yang |

Summary of other automated or manual tests (not corresponding to the acceptance criteria)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Number** | **Test Input** | **Expected Result** | **Class Name of the Test Code** | **Method Name of the Test Code** | **Status** | **Developer Name(s)** |
|  |  |  |  |  |  |  |

1. **Meeting Minutes**

Report the minutes of all meetings, including, but not limited to: project/sprint planning meeting, stand-up meeting, backlog grooming, retrospective meeting, and pair programming session.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Time and Duration** | **Place** | **Participant Names** | **Purpose of the Meeting** | **Specific Action Items** |
| 9/29/2020 | 9:50pm, 100 mins | Discord server | All | GitHub repo,  Unity setup,  User Stories | Worked on creating a repo. Worked on getting Unity and GitHub integrated.  Created User Stories |
| 10/6/2020 | 10:00pm, 90 mins | Discord server | All | Sprint 1 report | More User Stories  Acceptance criteria  More GitHub integration. |
| 10/13/2020 | 9:45pm | Discord server | All | Sprint 1 report | Acceptance criteria |

1. **Buddy Ratings**

If you don’t feel comfortable to include your ratings in this report, you may email your ratings to the instructor or grader.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *Rating giver* | *Rating receiver* | | | | |  |
|  | Ken Dozier | Zach Gharst | Joseph Soria | Thomas Tran | Thomas Yang |
| Ken Dozier | X | 1.0 | 1.0 | 1.0 | 1.0 |
| Zach Gharst | 1.0 | X | 1.0 | 1.0 | 1.0 |
| Joseph Soria | 1.0 | 1.0 | X | 1.0 | 1.0 |
| Thomas Tran | 1.0 | 1.0 | 1.0 | X | 1.0 |
|  | Thomas Yang | 1.0 | 1.0 | 1.0 | 1.0 | X |
|  | *Average* |  |  |  |  |  |