**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: Overwork and Understaffed

Team Members: Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds

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

Provide a brief description about the project, including the following elements:

Project name: Nine Men’s Morris

Vision statement: Our vision for this project is to create a functional and enjoyable version of the game Nine Men’s Morris using Python. The game will be simplistic and intuitive for users.

Mission statement/Project Purpose: The purpose of this project is to create a fully functional version of the Nine Men’s Morris that is easily accessible and playable by board game enthusiasts with computer access.

Elevator pitch: With this project, we aim to give board game enthusiasts with basic computing abilities a functional simulator of the game Nine Men’s Morris. In this game, we want to give them functionality that allows them to play both by themselves against a computer, or with one of their friends on the same computer.

Business value: Our Nine Men’s Morris game is being create on Python with Pygame and Pytest, which is an easily recognizable and more commonly used language. This should allow for this project to be updated and edited by other developers easily. In addition, keeping the game simplistic should create less maintenance issues.

Customers and users: We hope that most board game enthusiasts of any age with basic computer access and computing skills will be able to use our project.

Metrics: To measure business value, the metric we will measure by is customer satisfaction.

Milestones:

Sprint 1 (September 22nd – October 6th)

Board Visualization: September 29th

Piece Visualization: October 4th

Turn-based Gameplay: October 6th

Piece Differentiation: October 6th

Sprint 1 Project Report: October 8th

Sprint 1 Presentation Recorded: October 11th

Sprint 1 Deadline: Midnight, October 13th

Sprint 2 (October 20th – November 3rd)

Player vs. Player Functionality: October 22nd

Mill Functionality: October 27th

Phase 1 Production: October 29th

Phase 2 Production: October 29th

Phase 3 Production: October 29th

Start Screen Functionality: November 3rd

End Screen Functionality: November 3rd

Sprint 1 Project Report: November 5th

Sprint 1 Presentation Recorded: November 8th

Sprint 1 Deadline: Midnight, November 10th

Spring 3 (November 15th – December 1st)

Computer Selection: November 17th

Computer Functionality: December 1st

Sprint 1 Project Report: December 3rd

Sprint 1 Presentation Recorded: December 6th

Sprint 1 Deadline: Midnight, December 8th

Risks: Our team has a lack of experience, both in using Scrum as a developmental process of coding, as well as coding video game projects like this one that may pose some challenges to the completion of this projects. In addition, the time constraints of this project will not hopefully pose much risk, but could add some risk to the timely completion of this task.

Authors of this micro-charter: Ruby Rios, with guidance and feedback from Joshua Koni, Elizabeth Nastoff, and Marley Symmonds

1. **User Stories**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **User Story Name** | **User Story Description** | **Priority** | **Estimated effort (hours)** | **Actual effort (if completed)** | **Status (completed, to Do, in Progress)** | **Developer names** |
| 1 | Board Visualization | As a user, I want to be able to visualize the board so I can see the game. | Very High – Sprint 1 | 3 | 4 | Completed | Elizabeth Nastoff |
| 2 | Piece Visualization | As a user, I want to be able to place my pieces so I can see the game. | High – Sprint 1 | 3 | 5 | Completed | Joshua Koni |
| 3 | Turn-based Gameplay | As a user, I want the system to track whose turn it is so I can only play on my turn. | High –  Sprint 1 | 2 | 3 | Completed | Joshua Koni,  Marley Symmonds |
| 4 | Piece Differentiation | As a user, I want to know which color I am so I can know which pieces are mine. | High – Sprint 1 | 1 | 2 | Completed | Joshua Koni |
| 5 | Start Screen Functionality | As a user, I want to be able to select whether I want to play against a computer or another player, so I can choose who I play against. | High– Sprint 2 | 3 |  | To Do |  |
| 6 | End Screen Functionality - Close | As a user, I want to be able to close the game so I can end the application when I am done. | High– Sprint 2 | 1 |  | To Do |  |
| 7 | End Screen Functionality - Restart | As a user, I want to be able to reset the game so I can start a new game. | High – Sprint 2 | 1 |  | To Do |  |
| 8 | Player vs Player Functionality | As a user, I want to be able to play against another player on the same device so I can play against my friends when we are using the same computer. | High– Sprint 2 | 4 |  | In Progress | Marley Symmonds |
| 9 | “Placing Pieces” Phase Production | As a user, I want to be able to place my pieces anywhere on the board where there is not a piece currently during the Placing Pieces phase of the game, so that I can play the game. | High – Sprint 2 | 4 |  | In Progress | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff |
| 10 | “Moving Pieces” Phase Production | As a user, I want to be able to move my pieces one space on the board where there is not a piece currently during the Moving Pieces phase of the game, so that I can play the game. | High– Sprint 2 | 4 |  | In Progress | Joahua Koni |
| 11 | “Flying” Phase Production | As a user, when I have only three pieces left, I want to be able to move my pieces anywhere on the board where there is not a piece currently during the Flying phase of the game, so that I can play the game. | High – Sprint 2 | 4 |  | To Do |  |
| 12 | Mill Functionality | As a user, when I have moved three pieces into a “mill” (all of the pieces are in consecutive spaces in a row or column), I want to be able to remove one of my opponents’ pieces, so that I can play the game. | High – Sprint 2 | 2 |  | In Progress | Joahua Koni |
| 13 | End Screen – Win/Loss | As a user, I want to know if I have won or lost so I can know if the game is done. | Medium – Sprint 2 | 1 |  | To Do |  |
| 14 | Computer Selection | As a user, I want to be able to play against the computer so I can play solo. | Medium - Sprint 3 | 1 |  | To Do |  |
| 15 | Computer Functionality | As a user, I want the computer to place and move pieces so I can play against it. | Medium – Sprint 3 | 10 |  | To Do |  |

1. **Acceptance Criteria (AC)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Story ID and Name** | **AC**  **ID** | **Description of Acceptance Criterion** | **Status (completed, to Do, in Progress)** | **Developer Names** |
| 1 Board Visualization | 1.1 | When an option is selected from the start screen for player opponent, then 16 lines and 24 circles will appear on the screen in the formation of the Nine Men’s Morris board. | Board completed, selection screen to do | Elizabeth Nastoff |
| 2 Piece Visualization | 2.1 | Given a board is displayed on the screen, when the screen is clicked on a circle where there are not any existing pieces and it is the “Placing Pieces” phase, then a piece is placed on the screen at that location. | Completed, Placing Pieces phase In Progress | Joshua Koni, Elizabeth Nastoff |
|  | 2.2 | Given a board is displayed on the screen, when the screen is clicked on a circle where there is an existing piece and it is the “Placing Pieces” phase, then a piece will not be placed on the screen at that location. | Completed, Placing Pieces phase In Progress | Joshua Koni, Elizabeth Nastoff |
|  | 2.3 | Given a board is displayed on the screen, when the screen is clicked in a location where a piece cannot go and it is the “Placing Pieces” phase, then a piece will not be placed on the screen at that location. | Completed, Placing Pieces phase In Progress | Joshua Koni, Elizabeth Nastoff |
| 3 Turn-based Gameplay | 3.1 | Given a game is started, when it is the user’s turn, then they will be able to make an action according to what phase the game is in. | Completed | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff |
|  | 3.2 | Given a game is started, when it is the opponent’s turn, then they will be able to make an action according to what phase the game is in. | Completed | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff |
| 4 Piece Differentiation | 4.1 | Given a board is displayed on the screen and the board is clicked in a place where there are not any existing pieces, when it is the user’s turn and it is Phase 1, then the piece placed is in the color of the user. | Differ color pieces- Completed,  Phase 1 to do | Joshua Koni,  Marley Symmonds,  Elizabeth Nastoff |
|  | 4.2 | Given a board is displayed on the screen and the board is clicked in a place where there are not any existing pieces, when it is the opponent’s turn and it is Phase 1, then the piece placed is in the color of the opponent. | Differ color pieces- Completed,  Phase 1 to do | Joshua Koni,  Marley Symmonds,  Elizabeth Nastoff |
| 5 Start Screen Functionality | 5.1 | When the application has been opened, then a screen will be displayed with buttons for choosing an opponent. | To Do |  |
| 6 End Screen Functionality – Close | 6.1 | Given that a game has ended and an end screen appears, when the user clicks the “close” button, the application will close. | To Do |  |
| 7 End Screen Functionality – Restart | 7.1 | Given that a game has ended and an end screen appears, when the user clicks the “restart” button, the application will reset (the board will appear on the screen with no pieces on it, a new coin toss will be performed, the game will start in phase 1)  . | To Do |  |
| 8 Player vs Player Functionality | 8.1 | Given that a game has started and the option to play against another player has been selected on the start screen, when the user is playing the game, then they will be able to play against another player on the same computer with full game functionality. | In Progress | Marley Symmonds |
| 9 “Placing Pieces” Phase Production | 9.1 | Given that a game has started, when it is the user’s turn. less than nine pieces of the players have been placed on the board, and the place where the user clicks does not have any existing pieces on it, then the user will be able to play down a piece in this space. | In Progress | Joahua Koni,  Marley Symmonds,  Elizabeth Nastoff |
|  | 9.2 | Given that a game has started, when it is the opponent’s turn. less than nine pieces of the opponents have been placed on the board, and the place where the opponent clicks does not have any existing pieces on it, then the opponent will be able to play down a piece in this space. | In Progress | Joahua Koni,  Marley Symmonds,  Elizabeth Nastoff |
| 10 “Moving Pieces” Phase Production | 10.1 | Given that both players have placed all nine of their pieces, when the user clicks on a piece that is theirs and then click on a spot on the board that does not have any existing pieces on it, then the user will be able to move a piece to this space (the piece will disappear from its original position and reappear where the user has clicked). | In Progress | Joshua Koni |
|  | 10.2 | Given that both players have placed all nine of their pieces, when the user clicks on a piece that is theirs and then click on a spot on the board that does not have any existing pieces on it, then the user will be able to move a piece to this space (the piece will disappear from its original position and reappear where the user has clicked). | In Progress | Joshua Koni |
| 11 “Flying” Phase Production | 11.1 | Given that the user has only 3 pieces remaining on the board, when it is the user’s turn and the place where the user clicks does not have any existing pieces on it, then the user will be able to move a piece to this space (the piece will disappear from its original position and reappear where the user has clicked). | To Do |  |
|  | 11.2 | Given that the opponent has only 3 pieces remaining on the board, when it is the opponents turn and the place where the opponent clicks does not have any existing pieces on it, then the opponent will be able to move a piece to this space (the piece will disappear from its original position and reappear where the user has clicked). | To Do |  |
| 12 Mill Functionality | 12.1 | Given that a game is going, when 3 pieces of the users are aligned consecutively vertically or horizontally, then the player has formed and mill and gets to remove one of their opponent’s pieces from the board. | In Progress | Joshua Koni |
|  | 12.2 | Given that a game is going, when 3 pieces of the opponents are aligned consecutively vertically or horizontally, then the opponent has formed and mill and gets to remove one of the user’s pieces from the board. | In Progress | Joshua Koni |
| 13 End Screen – Win/Loss | 13.1 | Given that a game has ended, the user has won, and the game was player vs player, when the end screen is displayed, then it will say that the user has won. | To Do |  |
|  | 13.2 | Given that a game has ended, the opponent has won, and the game was player vs player, when the end screen is displayed, then it will say that the opponent has won. | To Do |  |
|  | 13.3 | Given that a game has ended, the user has won, and the game was player vs computer, when the end screen is displayed, then it will say that the user has won. | To Do |  |
|  | 13.4 | Given that a game has ended, the opponent has won, and the game was player vs computer, when the end screen is displayed, then it will say that the user has lost. | To Do |  |
| 14 Computer Selection | 14.1 | Given that a game has started and the start screen is displayed, when the user selects to play against a computer, then the user will begin a game against a computer. | To Do |  |
| 15 Computer Functionality | 15.1 | Given that a game has started and the option to play against a computer has been selected on the start screen, when the user is playing the game, then they will be able to play against another player on the same computer with full game functionality. | To Do |  |

1. **Implementation Tasks**

Summary of production code

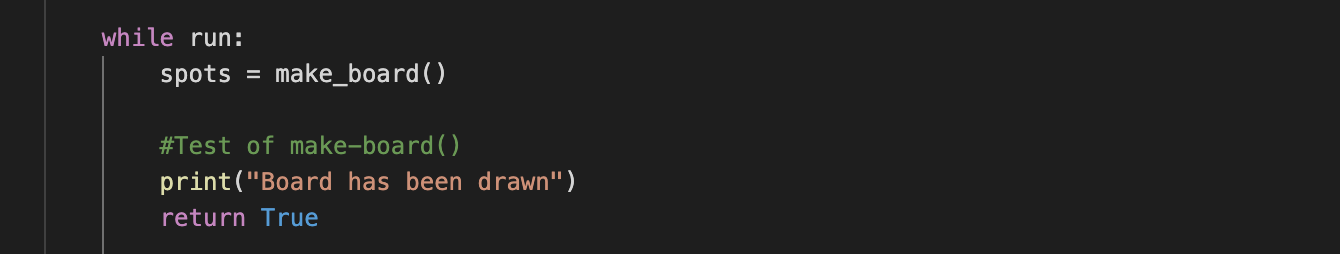
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Story ID and Name** | **AC ID** | **Class Name(s)** | **Method Name(s)** | **Developer Name(s)** | **Status** | **Notes (optional)** |
| 1 Board Visualization | 1.1 |  | def main()  def make\_board() | Elizabeth Nastoff | Completed | Start Sceen to be made in next sprint |
| 2 Piece Visualization | 2.1 |  | def main()  def Piece\_Location(), def make\_board() | Joshua Koni, Elizabeth Nastoff | Completed |  |
|  | 2.2 |  | def main()  def Piece\_Location(), def make\_board() | Joshua Koni, Elizabeth Nastoff | Completed |  |
|  | 2.3 |  | def main()  def Piece\_Location(), def make\_board() | Joshua Koni, Elizabeth Nastoff | Completed |  |
| 3 Turn-based Gameplay | 3.1 | Gamer | def main() | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff | Completed | This code will be further edited in Sprint 2 |
|  | 3.2 | Gamer | def main() | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff | Completed |  |
| 4 Piece Differentiation | 4.1 |  | def main()  def Piece\_Location(), def make\_board() | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff | Completed |  |
|  | 4.2 |  | def main()  def Piece\_Location(), def make\_board() | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff | Completed |  |
| 8 Player vs Player Functionality | 8.1 | Gamer | def main() | Marley Symmonds | In Progress |  |
| 9 “Placing Pieces” Phase Production | 9.1 |  |  | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff | In Progress |  |
|  | 9.2 |  |  | Joshua Koni, Marley Symmonds,  Elizabeth Nastoff | In Progress |  |
| 10 “Moving Pieces” Phase Production | 10.1 | Pieces, Gamer | def main() | Joshua Koni | In Progress |  |
|  | 10.2 | Pieces, Gamer | def main() | Joshua Koni | In Progress |  |
| 12 Mill Functionality | 12.1 | Pieces, Gamer | def main() | Joshua Koni | In Progress |  |
|  | 12.2 | Pieces, Gamer | def main() | Joshua Koni | In Progress |  |

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)** |
| 2 Piece Visualization | 2.1 |  | Piece\_Location(x, y. spots\_Dict),  Piece\_Location\_Test(spots\_Dict) | Piece\_Location() returns true is the x and y values input are in the range of an item in spots\_Dict. Piece\_Location\_Test() checks that this works for every coordinate on the board  The expected output is null. | Working | Ruby Rios |
|  | 2.3 |  | Piece\_Location(x, y. spots\_Dict),  Piece\_Location\_Test(spots\_Dict) | Piece\_Location() returns true is the x and y values input are in the range of an item in spots\_Dict. Piece\_Location\_Test() checks that this works for every coordinate on the board  The expected output is null. | Working | Ruby Rios |
| 3 Turn-based Gameplay | 3.1 |  | def CheckTurnTest (turn) | def CheckTurnTest (turn) gets which turn the game is at, and checks that the color of the piece and the player putting down that piece are correct.  The expected output is null. | Working | Ruby Rios |
|  | 3.2 |  | def CheckTurnTest (turn) | def CheckTurnTest (turn) gets which turn the game is at, and checks that the color of the piece and the player putting down that piece are correct.  The expected output is null. | Working | Ruby Rios |
| 4 Piece Differentiation | 4.1 |  | def CheckTurnTest (turn) | def CheckTurnTest (turn) gets which turn the game is at, and checks that the color of the piece and the player putting down that piece are correct.  The expected output is null. | Working | Ruby Rios |
|  | 4.2 |  | def CheckTurnTest (turn) | def CheckTurnTest (turn) gets which turn the game is at, and checks that the color of the piece and the player putting down that piece are correct.  The expected output is null. | Working | Ruby Rios |

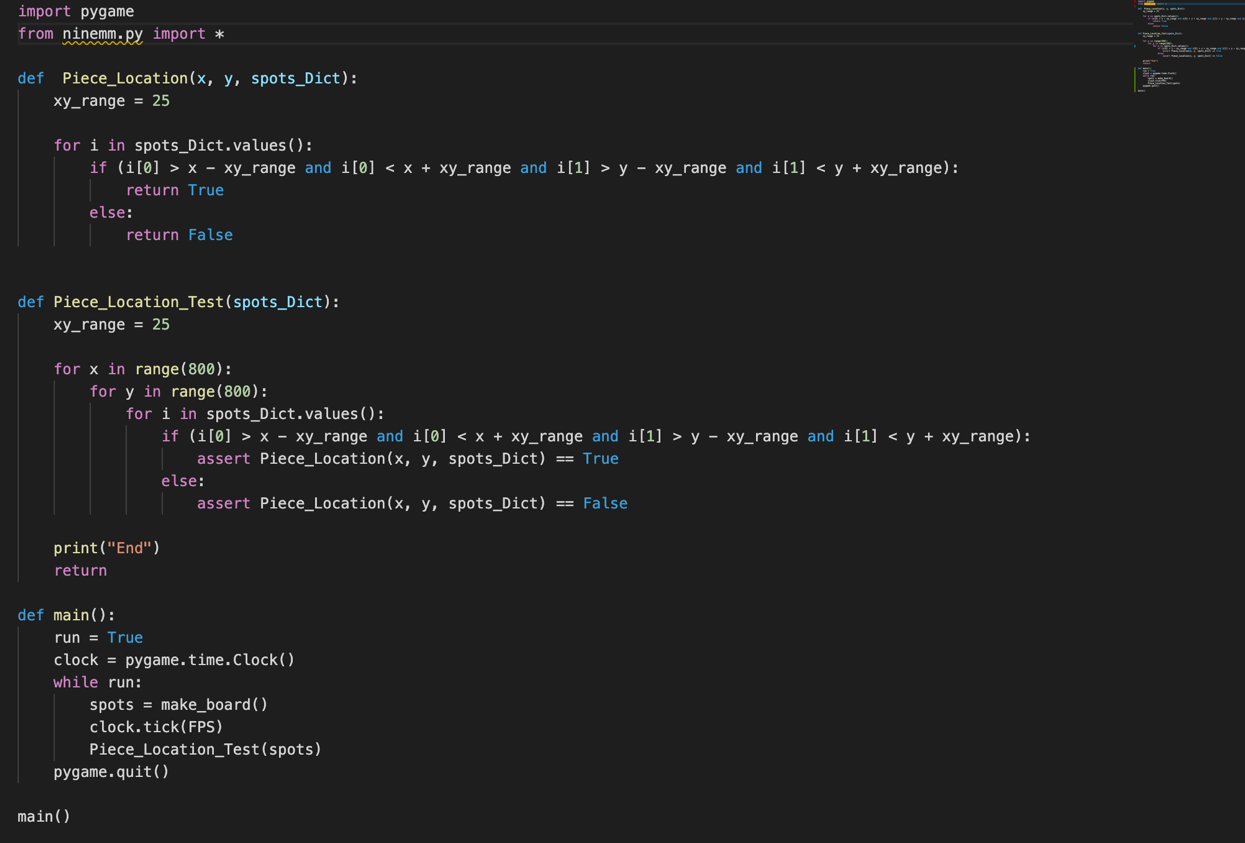
Summary of manual test cases (directly corresponding to some acceptance criteria)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Story ID and Name** | **Acceptance Criterion ID** | **Test Case Input** | **Test Oracle (Expected Output)** | **Status** | **Notes** | **Developer Name(s)** |
| 1 Board Visualization | 1.1 | print(“Board has been drawn”)  return True | “Board has been drawn”, before the game closes | Working | More tests done on board functionality in other tests. | Joshua Koni, Ruby Rios |
| 2 Piece Visualization | 2.2 | print("Piece has already been placed!")  Input cases for all 24 circles checked | For each circle, on first click:  x, y, i[0], i[1]  YEAH BOI  Upon repeat click:  x, y, i[0], i[1]  YEAH BOI  Piece has already been placed! | Working | Test worked when first implemented. Currently, with the beginnings of a piece counter, the test does not work, as the counter only allows 9 pieces to be placed down by each player. | Ruby Rios |

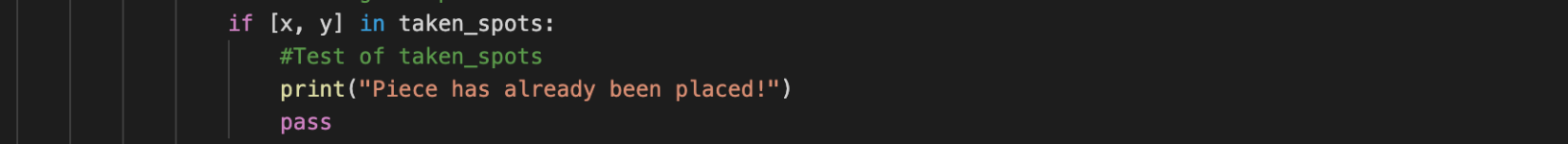
AC 1.1:



AC 2.1, 2.3:



AC 2.2 (The AC 2.2 Terminal Results are in a folder on the GitHub in Documentation (too long to put in here)):



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** |
| 9/1 | 1:00-1:30 PM  30 minutes | Zoom | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Team Introductions, Play Nine Men’s Morris | Research key decision items. |
| 9/3 | 1:20-2:15 PM  55 minutes | Zoom | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Discussion about project requirements, Make Key Decisions | Research project organization methods, as well as python video game making |
| 9/11 | 12:00-12:35 PM  35 minutes | Zoom | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Further discussion about project requirements, pygame and pytest discussions | Continue with researching specific project components and requirements. Waiting for lecture on user stories (next week). |
| 9/17 | 1:00-1:45 PM  45 minutes | Zoom | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Create and distribute user stories in Jira | Each individual to get started with their user story assignment. Waiting for lecture on acceptance criteria (next week). |
| 9/23 | 8:00-11:00 PM  3 hours | Microsoft Teams | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | A pair programming session to help initialize the project in GitHub and layout project components. | No action items were created at this meeting. Further steps for the project will be assigned at the meeting tomorrow. |
| 9/24 | 12:50-1:00 PM  10 minutes | Microsoft Teams | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | A standing meeting to create action items for the weekend | Every individual was assigned tasks to further where they are with their individual user story. |
| 9/29 | 12:50-1:00 PM  10 minutes | Microsoft Teams | Elizabeth Nastoff, Ruby Rios, Marley Symmonds | A standing meeting to create action items for the week. | Every individual was assigned tasks to further where they are with their individual user story. |
| 10/1 | 9:00-9:45 PM  45 minutes | Microsoft Teams | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Collaborate and get feedback on code and sprint report | Future pair programming meeting on continuation with piece placement work. With the feedback on the sprint report, make corrections.  Start on presentation for Sprint 1. |
| 10/6 | 1:00-1:15 PM  15 minutes | Microsoft Teams | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Check in on where individuals are with their portions of the project | Action Items are the continuation of the last meeting’s:  Future pair programming meeting on continuation with piece placement work. With the feedback on the sprint report, make corrections.  Start on presentation for Sprint 1. |
| 10/8 | 12:00-3:00 AM  3 hours | Microsoft Teams | Elizabeth Nastoff, Marley Symmonds | Pair Programming Session to work on piece placement | No action items were created at this meeting. Further steps for the project will be assigned at the meeting tomorrow. |
| 10/9 | 2:30-3:00  30 minutes | Microsoft Teams | Joshua Koni, Ruby Rios, Marley Symmonds | Go through User Stories, determine what is still needed for Sprint 1 | Work on writing test code for acceptance criteria.  Finish Sprint Report 1.  Start on presentation for Sprint 1. (Meeting planned for Sunday evening) |
| 10/11 | 10:30 PM – 3:30 AM  5 hours | Microsoft Teams | Joshua Koni, Ruby Rios | Work on writing test code for acceptance criteria,  Work on Sprint 1 Report | Talk to other team members about test code.  Finish test code.  Finish Sprint Report 1. |
| 10/11 | 11:00 PM - ? | Microsoft Teams | Joshua Koni, Elizabeth Nastoff, Ruby Rios, Marley Symmonds | Create Presentation Video | Everyone is to look at all of the components of what is being submitted for Sprint Report 1 before it is turned in.  Start thinking about Sprint Report 2. |

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* | | | | |
|  | Joshua Koni | Elizabeth Nastoff | Ruby Rios | Marley Symmonds |
| Joshua Koni | X | 1.0 | 1.0 | 1.0 |
| Elizabeth Nastoff | 1.0 | X | 1.0 | 1.0 |
| Ruby Rios | 1.0 | 1.0 | X | 1.0 |
| Marley Symmonds | 1.0 | 1.0 | 1.0 | X |
|  | *Average* | 1.0 | 1.0 | 1.0 | 1.0 |