**Team Name**: Saucy South  
**Members**: Sanjum Sahni (ss3873), Sreya Jonnalagadda (sj652), Srinithi Krishnamoorthy (sk2693)

**PM**: Brandon Lu bl533

**Vision**

In one paragraph, what is your current vision for the system you are building? How has it evolved from previous plans? It's okay for you to make changes, even big ones.

Our vision for our final project is to put a twist on the classic board game Battleship, where the user attempts to identify the coordinates of various caravans of camels. In our first version, we had followed a format much closer to that of the original Battleship but our version of the game had the user guess where the camels may be within the grid (denoted by a 8x8 board of cacti). This was done by taking in user input for coordinate guesses and providing them with appropriate feedback, such as a hit (denoted by a camel) or a miss (denoted by a X). In our final version, the game has evolved to contain many more functionalities, such as a client/server connection, leaderboard log, user input specific hints, and etc. In general however, the final vision was to make Camel Caravan a multiplayer game where users (pairs of 2) can connect with a game log server to start playing; also the objective of the game is now to find the camels the fastest and before the other player.

**Summary of progress**

Write a one or two-paragraph description of what your team accomplished between MS2 and MS3. What functionality did you work on?

Between MS2 and MS3, our team focused on developing several more key functionalities that form the game logic as well as the user interface of our game. We left off with our terminal reading in the user input and then processing that input to provide basic feedback regarding the accuracy of the guess. From there, we changed our game so it handled camels of both size 1 and size, with three total camels being placed at all times (with two of three camels representing a camel of size 2). After that the feedback provided to the user based on their input was improved. In particular, we provide three main types of responses: answer hints, proximity hints, and general feedback. The answer hints help reveal the actual answer of each camel’s coordinates in increments and are given if the user enters either hint1, hint2, or answer into the terminal. The proximity hints take in the user's guess and in turn returns a hint as to where the closest camel is in comparison to the guess. Finally, the general feedback helps the users keep a track of their guesses by letting them know if they have gotten all camels right or how many remaining camels they need to guess.

The next functionality we implemented was creating a client/server connection where the server is represented as a game log which the clients can choose to connect to if they want to get feedback, such as time played. On the client/user side the game alternates between player 1 and 2 until the first person to guess all 3 camels wins. At the end of the game we also implemented a leaderboard and gamelog text file which the users can choose to download to see how they compared to other players and how fast they completed their games.

**Activity breakdown**

For each team member, give a bulleted list of the responsibilities that team members had, the activities in which they participated, the features they delivered, and the number of hours they spent working.

Sanjum Sahni MS1 to MS2:

* -  Designed and implemented the game board in the terminal
* -  Enhanced user guess-checking algorithm by checking for accuracy
* -  Enhanced feedback mechanism for hits and misses
* -  Debugged game and user interface logic
* -  Number of Hours: 13

MS2 to MS3:

* -  Created functionality to add more than one caravan of camels to the board
* -  Added timer to check how long it takes for each pair of players to complete the game
* -  Developed some of the OUnited test cases for some of the functions
* -  Before, we only had ship.ml and ship.mli. I split it up so now we have ship.ml, ship.mli, board.ml, and board.mli
* -  Added functionality to check the user’s guesses and tell them if the guess is right or not
* -  Number of Hours: 25 hours

Sreya Jonnalagadda

MS1 to MS2:

* -  Designed and implemented the game board in the terminal
* -  Worked on the random placement algorithm for the camel
* -  Developed OUnit test cases to test functions implemented thus far
* -  Debugged game and user interface logic
* -  Number of Hours: 12

MS2 to MS3:  
- Developed OUnit test cases to test functions implemented

-  Added helper manual for users to ask for before game begins

* -  Implemented answer hints prompts, such as hint1, hint2, answer
* -  Implemented proximity hints for user based on guess input, such as warm or hot
* -  Implemented leaderboard and gamelog txt files for user
* -  Implemented ranking system in leaderboard to report lowest and highest scoring team
* -  Number of Hours: 25

Srinithi Krishnamoorthy MS1 to MS2:

* -  Implemented basic user guess-checking algorithm, checking for validity
* -  Implemented the feedback mechanism
* -  Debugged game and user interface logic
* -  Number of Hours: 10

MS2 to MS3:

* -  Implemented OUnit Test suite in test\_battleship.ml
* -  Implemented the game from single-player to multiplayer on the client end
* -  Implemented basic server and client functions for the game that allows players to connect to a larger server that contains the leaderboard with other clients
* -  Refactored functions in board.ml to be more efficient
* -  Implemented test cases for update\_player\_guess, handle\_hint and read\_user\_guess within the test suite
* -  Numbers of Hours: 26

**Productivity analysis**

As an entire team, how productive were you? Did you accomplish what you planned in your sprints? Were your estimates of what you could do accurate, or far off? Write a paragraph addressing those questions. Please be honest: we want you to reflect candidly on your progress so that you can make more accurate estimates between MS2 and MS3. Your grade is not going to be based on how positive or negative you are here.

As a team, we made great progress between MS2 and MS3. We were able to get our full functioning Camel Caravan game done by finishing building the user interface and setting up more advanced game logic. In particular, after MS2 we compiled a list of tasks we wanted to accomplish and went beyond that list by the end of MS3. While some sprints were more complicated than others which meant a slight underestimation of time, we were still able to fix those challenges and adapt to the new changes we made. For example, implementing the client/server feature was challenging at first due to the format of a gameboard and the multi-player feature, but we were able to accomplish that component and even add more functionality to it, such as a scoreboard text file and a game duration timer. Adding on, once we began implementing testing, our overall team productivity increased significantly as we were able to understand where to make necessary changes and efficiently continue building and fixing our program. Testing not only helped with functionality but also helped our team gain a better understanding of our game logic, which we then used to continue adding more advanced features. Overall, our team has felt that the progress made by the end of MS3 was significantly important as we were able to accomplish everything planned and beyond.