Team: GAP

Members: Arushi Aggarwal (aa2555), Petros Georgiou (pag238), Grace Wei (gtw25)

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.

This project is a game inspired by Mastermind, a code cracking game between two players, where one player assumes the role of the code maker, and the other the code breaker. The code breaker guesses 4-digit codes that the code maker creates using keys, while providing feedback, until the code is guessed. The player plays against the computer using a graphical user interface that they navigate using keyboard keys. If the player is the code maker, then they create a code and the computer will guess for a limited number of tries. If the player is a code breaker, the computer generates a code by which the player must guess. Feedback is similar to the game Wordle, however positions are not revealed. Our plan has changed considerably since the start; we originally wanted to focus on a complex algorithm-based system that served to show users how those game-solving algorithms worked, however we transitioned into focusing more on the GUI and gameplay rather than the solver/algorithms. We developed an interest in exploring the GUI rather than the algorithms and wanted to explore that side more, and also thought that it would be more feasible to create a well-developed project. While the game runs both ways in terms of the computer and players can both play in how many rounds they want, we have focused on one algorithm with feedback and validation checking at each step. There is also a higher level of user interaction as the player can actually play the game and give feedback when the computer is guessing.

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, we have focused a lot on the GUI side of things as initially in MS2 it was a very simple terminal based game, but now the entire game runs on a GUI made using the Ocaml Graphics module. There is also now a backend storing game information which builds a record type data structure of the game board array, and different parts of the game which do backend and front integration with the GUI (Game.ml). We also programmed a respective feedback tool using a Pin module, added constraints as to what the user can input in the GUI and lots more verification checks, and created a much more visually appealing graphical interface to represent the game. We also made the 50 test cases and did more documentation on everything.

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

Arushi

- Developed and later reiterated the pseudo randomizer algorithm and the donald knuth algorithm (had to scrap last minute due to bugs, you can check it out in the git commits!) from scratch with references.
- Worked on the GUI in terms of getting user answer and user guess, frontend and backend integration for the board and feedback array, worked on UI and display, developed the win condition and the help screen
- Worked on game.ml to update board in terms of the backend and get computer guess
- Worked on the final User manual and the final MS3 Report
- 40 hours

Petros

- Attempted to research an algorithm and attempted to design it with Matthew Eichhorn but with the time constraint could not get another working algorithm.
- Instead, created a pin module, did some slight work on the GUI, some work on the computer representation of the game, and worked extensively on designing test cases for major modules, the Pin and Game, along with designing the testing document.
- Worked on user manual
- 20 hours

Grace

- Created base GUI outline, with screen transitions, key event handlers, and each page layout
- Worked on networking between front end and back end
- Implemented compilation unit for game module data type and functions
- Added a few OUnit test cases and QCheck test
- Revised yaml, install.txt and README.md files
- Worked on user manual and test plan
- ~40 hours

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 an entire team, we were less productive than we would have liked to have been. The plans that we made for our sprints kept getting prolonged, whether it was because of struggles getting it to work or not having enough time to finish. Additionally, due to our schedules, it was hard to find a time to really sit down together and deal with all of the merging and networking. When we finally came together, we saw that quite a bit of our code didn't work as intended.

Despite this, we worked endlessly to overcome those challenges and complete as much as we could. We did not meet the goals for our planned sprints, we still had a comprehensive set of ideas in terms of the GUI and the actual game play that we set out to accomplish and our lack of productivity in the middle of the semester forced us to accomplish our ideas closer to the deadline. While we made a huge jump from MS2 to MS3, the journey was not easy since the last couple of days were a bit stressful in terms of finishing our goals. We did accomplish everything we set out to do but this leaves us with a reminder how we should better focus and plan our time in future group projects.