Oscar So (ons4) & Aidan Chalnick (ac2597) & Tushar Khan (tak62)

- **Vision:** In one paragraph, what is your current vision for the system you are building? How has it evolved from previous sprints?

We think that given how we had to change our topic halfway through reaching the deadline for MS1, we are really proud of what we have researched, learned, and accomplished. We built a platform that allows users to login, play games, and send messages to other users on the platform. Having to change our project from an interpreter in our own language to an application platform initially hindered our progress a lot but we ended up successfully creating a mini, unaesthetic, communication platform in MS1, which we have further expanded upon for MS3. We added a security encryption system in MS2 to make the communication system more secure. In MS3, we developed more add-ons and added extensions to the platform, such as: emoji functionality, a very basic checkers board game with an AI, a minesweeper game, and passwords for users so that they cannot pretend to be anyone but themselves.

- **Summary of progress:** Write a one or two paragraph description of what your team accomplished during the previous sprint. What functionality did you work on? What did you show off in your demo?

As mentioned above, we developed more features for our application which helped expand what we wanted our app to be like. In the demo, we showed the TA our application connected between 3 computers and showed every feature implemented from passwords, to emojis, to checkers. One thing that was recommended was that we also add on texts and parse them as emojis. For example, if the user sends “:)” it should produce the emoji instead. We took this advice and worked on that after the demo as well. Furthermore, we also ensured that we had well-documented code and .mli files that would be really helpful for other readers looking at our code. Lastly, we wrote test suites and updated our makefile.

- **Activity breakdown:** For each team member, give a bulleted list of the responsibilities that team member had and the activities in which they participated during the sprint.

Oscar:

- Added Emoji Features
- Worked on App Interface
- Added checkers and minesweepers into app.
- Fixed lots of bugs by catching errors
- Test Suite
- Added password option for users when sending messages.
- Decoupled and tidied code
- Updated makefile
- Progress Report

Tushar:

- Created Checkers Game
- Created Minesweeper Game
- Created Checkers AI
- Updated makefile
- Test Suite
- Progress Report
- Worked on more encryption
- .merlin and .ocamlinit files

Aidan:

- Worked on GUI to try and create and interface for our messenger, though ultimately scrapped as it was incompatible with ANSITerminal and Lwt_IO
- Progress Report
- Test Suite

- **Productivity analysis:** As an entire team, how productive were you? Did you accomplish what you planned? Were your estimates of what you could do accurate, or far off? Write a paragraph addressing those questions.

Given our time constraints, our team was thoroughly productive. Despite not fully realizing the online messenger we sought out for, we managed to develop a solid way of communication between two computers and further package text-based

games within what is essentially a communication app. Our estimates of what kind of system to build were slightly ambiguous due to communication protocols being foreign to all three of us, however, we are ultimately content with the system we have produced.

- **Scope grade:** Give your team a scope grade for this sprint—Satisfactory, Good, or Excellent—based on your experience of those levels of scope in the assignments thus far in this course. Write a paragraph or two providing a detailed justification of why you gave yourself that grade. Please be honest: we want you to reflect candidly on your progress. Your sprint grade is not going to be based on what you self-assign here.

Scope Grade: Satisfactory.

We gave ourselves satisfactory because our system is sort of a barebones model for what a messaging application system can accomplish. We successfully implemented communication between computers as well as support for emojis. Furthermore, all of our messages are encrypted and users are password protected. Additionally, our messenger supports two basic games: checkers and minesweeper. However, all of these features can only be demonstrated through an unaesthetic terminal and some features could stand to be fleshed out further. It is for these reasons we gave ourselves satisfactory scope: we have created a broad and functional proof of concept that has room for improvement.