Self Reflection 3

What did I contribute in iteration 3?

I didn't contribute much towards the iteration 3 business model canvas, MVP canvas, learning cards, test cards, or Figma prototype. I did lead the effort for the MVP implementation and published that.

What did Himanshu contribute?

What did Rafael contribute?

Rafael did most of the business model canvas, test cards, learning cards and MVP canvas, and I'm pretty sure the competitor analysis as well. Kudos to him! And he did some of the styling with Patrick

What did Patrick contribute?

Patrick was our resident designer, and was responsible for the amazing styling present on the website now! The website would still look like "an excel spreadsheet" (CY interview 2) without him.

What did Liam contribute?

Liam was amazing, he did most of the Figma prototype by himself, as well as spending hours learning and implementing JS code to the MVP. His help was invaluable

What was the single most important lesson that I learnt in this course?

Many hands make light work. I don't normally program on interesting projects (outside of work) with other people, but it makes it a lot easier. I don't care much for CSS and styling, so Patrick taking on that role made the final product so much better

What did you learn about working in a team?

Github issues are actually quite useful. Github projects are great to get a nice overview

What did you learn about innovation?

Most of it comes down to *talking to the customers* and that aspect is in fact the most important part of innovation in my mind.

What was the best aspect of my participation in the team and why?

I wasn't required to "sign off on" or moderate every change to the MVP repo because I had scaffolded it well enough that other people could comfortably work on it. I did this because I knew my teammates were competent and could contribute code that was useful

Area of most improvement? Why?

Some things I said I would do I didn't get time to do, simply because I wasn't great at time management. I would spend 7 hours on a tiny issue, but not spend the couple hours it would take me to solve an important team-wide issue that would unblock much
I think this is a problem in myself because I like to do things correctly more than I want to see the outcome. I master the process without caring too much about the outcome. This is because I enjoy the art of programming, not so much the properties of the end products.