

TakuMahi

Software Presentation

What is TakuMahi?

TakuMahi is the fresh solution to rostering at AskOtago.

You will now be able to view your upcoming shifts, have your shifts updated and changed, see all your employees shifts and it's all dressed up in new and improved UI design.

We have built this system to meet our stakeholder needs. Our Stakeholders are:

- University of Otago Executive Management
- AskOtago Management
- AskOtago Employees
- Developers

What are the Problems?

Employees have problems with knowing their work shifts.

This occurs due to the miscommunication between the employees with their employers as communication is not linear.

In order to combat this problem, we identified key features we need to implement.

Our Journey: Features

At the start of our project, we had Identified many key Features that we wanted to include to keep our stakeholders happy.

- Users will be able to log into an online website
- Employees will be able to view their shifts
- Employees are able to select suitable times to work
- Authorised users would be able to alter employees shifts
- Managers will be able to view shifts by role/department
- People would be able to connect to the server from anywhere with an internet connection
- The webpage will be designed in a similar fashion to other Otago IT systems.
- Authorised users would be able to create new accounts for staff

File Management

As the days ticked on, our project naturally built up in size and complexity.

To ensure a high quality product we made sure to continuously track files we had deemed important to us. These Files were identified as our Configuration Items.

Such Configuration Items were:

- Source Code
- Project Issues
- External libraries used in the project
- Git & GitBucket

Maintaining High Code Quality:

By Incorporating Continuous Integration throughout our project, we ensured that our product would stay functioning. Using the GitBucket CI as our main tool, we were able to test our branches in the test suite.

We would also upload our own branches and have the Sprint Master at the time merge them into the main branch. By incorporating many small changes we can ensure the stability of the main branch.

Of course we ensured our minimum standard requirement of merging to the main branch is that it would pass all tests.

Managing Our Risks

Throughout this project we used an Agile Framework to ensure we produced a viable product.

By implementing this early, we could identify our issues and risks from our previous sprints and update a new direction in the next sprint with our obtained knowledge.

By having both asynchronous & synchronous communication methods we were able to schedule our project functions in through our Scrum sheet.

What We Have Learned

Building TakuMahi was a big project for all of us and we have learnt much from our time with together.

We now know to be more thorough in our system specifications. To ensure simplicity and reduce spaghetti code.

We can also be more confident in our time estimation skills, knowing how to identify how much time some projects may take.

