

Kieran Standeven

0423212771

kieran.standeven@gmail.com

Gold Coast, Australia

Experienced mobile developer with 4 years of industry Flutter experience, including 2 years as a tech lead, alongside over a decade of native iOS and Swift development.

With a career in mobile that began in the earliest days of the iPhone, I'm passionate about building apps that go beyond functional code. Delivering smooth, intuitive, and high-quality user experiences is just as important as the technology behind them.

Work History Overview

Jumbo Interactive, Senior Mobile Developer (iOS, Android)

December 2024 - Present

QUT, Contract Mobile Developer (iOS, Flutter)

January 2023 - December 2024

CartonCloud, Mobile Technical Lead (iOS, Flutter)

September 2020 - December 2022

Trade Me, iOS Developer / Squad Master

March 2018 - August 2020

Air New Zealand, Senior Contract Mobile Developer

July 2016 - July 2017

Find my time - Ruby on Rails Website Creator

April 2017 - January 2018

Motim Technologies, iOS Developer

July 2014 - March 2016

Geoop, iOS Developer

January 2014 - July 2014

Fiserv, Mobile Developer

August 2011 - September 2013

My Apps (Unity, Cocos2d, C#, iOS, Flutter, QGIS)

*June 2011 - **Present***

Nuffie Productions, Mobile Developer

November 2010 - June 2011

Zodal, Mobile & Web Developer

October 2008 - November 2010

Work History Detail

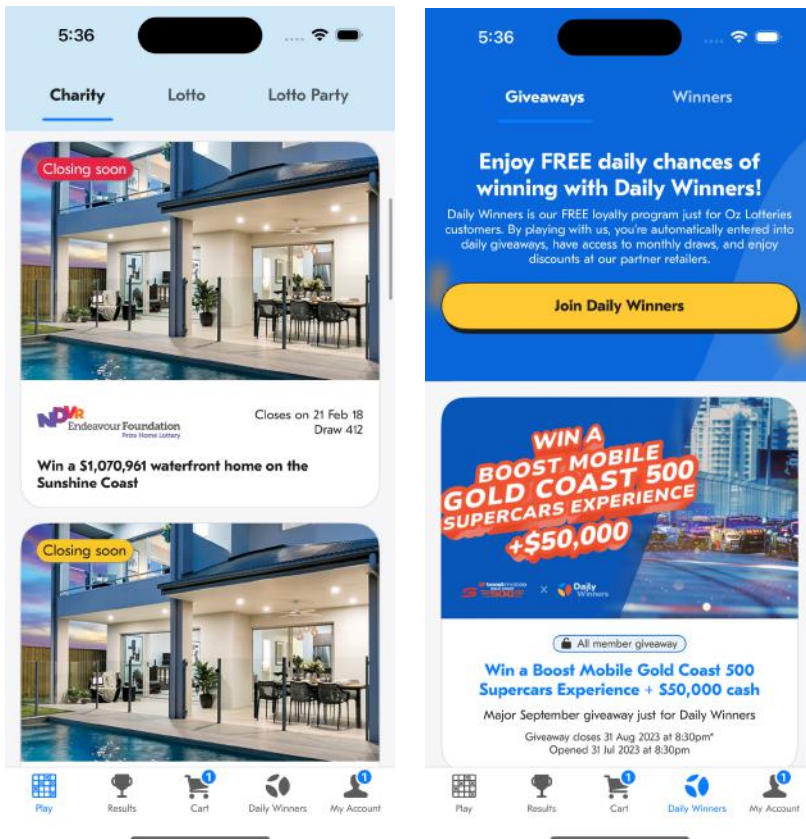
Jumbo Interactive January 2025 - present

My role at Jumbo is primarily focused on iOS development, with occasional contributions to Android. Key technologies I work with include Gerrit, SwiftUI, legacy UIKit and Objective-C codebases, and Android Compose.

When I joined the iOS team, the app had not yet begun migrating to SwiftUI and was still heavily reliant on LayoutKit (UIKit-based) alongside some Objective-C components. One of my main achievements early on was designing and implementing an approach that enabled SwiftUI-first development while still interoperating with the existing LayoutKit views.

Because LayoutKit was deprecated and increasingly difficult to maintain, this transition was a critical step for the long-term health of the codebase. The solution was built around an MVVM foundation, supported by repository and provider patterns to create clean boundaries between new SwiftUI features and legacy app infrastructure.

All the iOS developers in the company have transitioned from LayoutKit to using SwiftUI and these new tools.



Queensland University of Technology iOS and Flutter January 2023 - December 2024

I joined Queensland University of Technology (QUT) as a contract mobile developer, contributing to their Flutter application while specialising in native iOS development when required.

During my time on the project, I delivered a range of high-impact features and platform improvements, including:

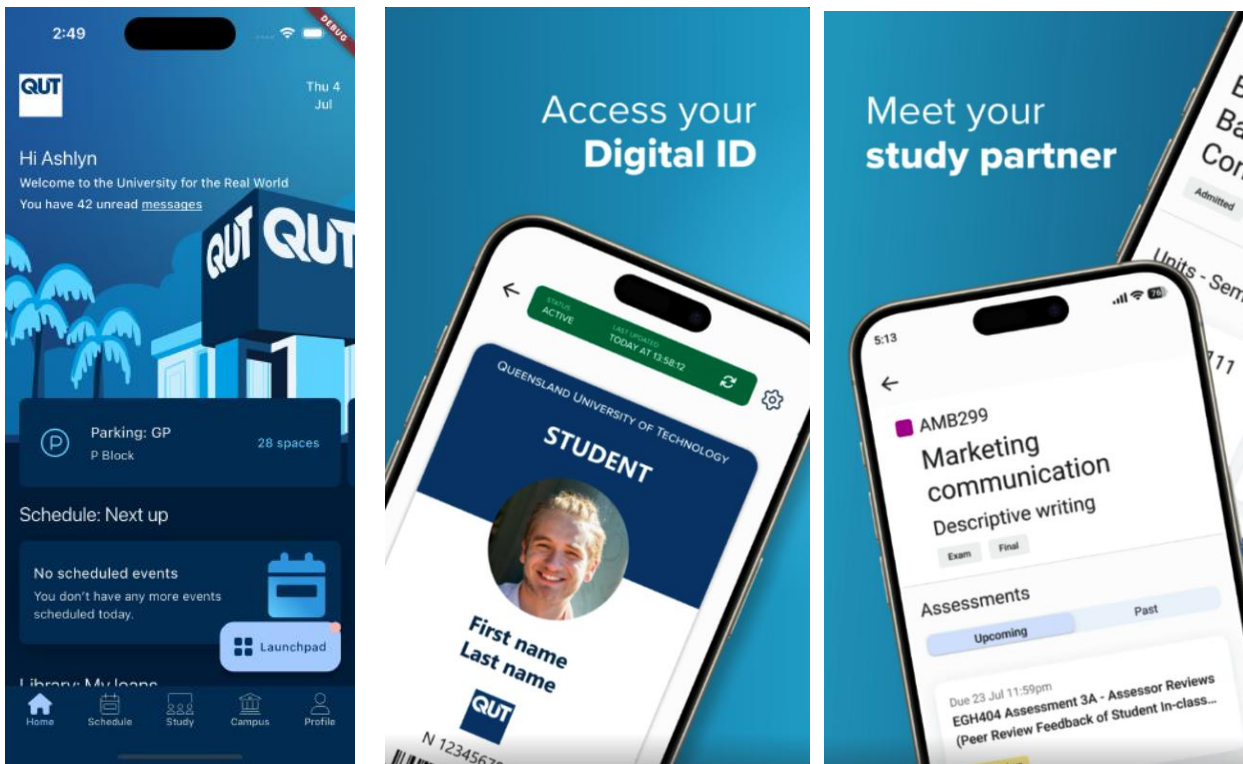
- **Academic insights:** Enhancing the way students view and track their grades and assessment results.
- **Library services:** Enabling staff and students to manage and renew library book loans directly within the app.
- **Campus parking:** Implementing real-time parking space availability for nearby parking facilities.
- **Transport updates:** Building a campus shuttle bus feature that displays live arrival times at each stop.
- **Media playback:** Developing a custom video player to support hosted university video content.

- **Digital student ID:** Migrating the physical ID card into a secure digital format, allowing staff and students to unlock doors and elevators across campus using their phone.
- **Material 3 redesign:** Leading a major restyling of the app to align with Material 3, with full support for light and dark mode. This required a significant renovation of the theming foundation — including colours, typography, spacing, and component styling — which later enabled improved tablet support.

As a result, the application is now better positioned for future development, with a more scalable design system and reduced need for large-scale UI overhauls moving forward.

This project used minimal libraries, most solutions were custom built, however some key technologies were:

- Firebase
- Dart/Flutter & Swift
- Mockito for a testing
- One Signal for app notifications
- Custom presenter and feature based architecture

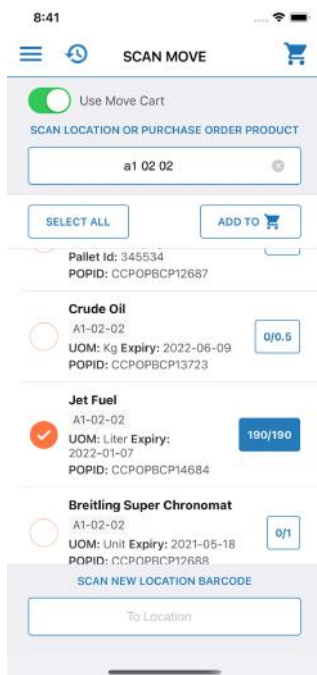


CartonCloud iOS/Swift September 2020 - December 2022

CartonCloud is a warehouse logistics and delivery company. Their mobile apps are used by warehouse workers and also delivery drivers to handle the end-to-end delivery of a product. The mobile code base initially needed some updating, so there were several initiatives I got going to help, including-

- Making build servers to build, test and release the apps.
 - Bitrise was the build server platform used for both iOS and android. This was later moved to a new Jenkins machine due to needing a more custom solution to run our automated tests.
- Appium UI automated test framework.
 - Some key technologies used were Java 11, Cucumber and the Page Object Model pattern.

- The UI tests ran in the Bitrise build servers, alongside the automated builds.
- Updating the iOS apps to start using Swift for new code.
 - Originally there was no Swift support in the app code, or in the unit tests. This was added and all new code going forward was written in swift.
- Integrated Flutter into the iOS and android app to be used for suitable new features, see below for more details.
- New architectures
 - As a way of making the code more consistent and readable, I created some architectures for the most important components in the app. One was a repository pattern for accessing the Database and Api (as this was not being done centrally). I also created class templates to be used based on the View Model pattern. This was so all new code being written had similar features and was easier for everyone to follow.
- The first new feature I added to the iOS native app was to help the warehouse workers move products around the warehouse, by scanning them with the mobile phone. The Feature was called “Scan Move 2”.



CartonCloud - Flutter May 2021 - December 2022

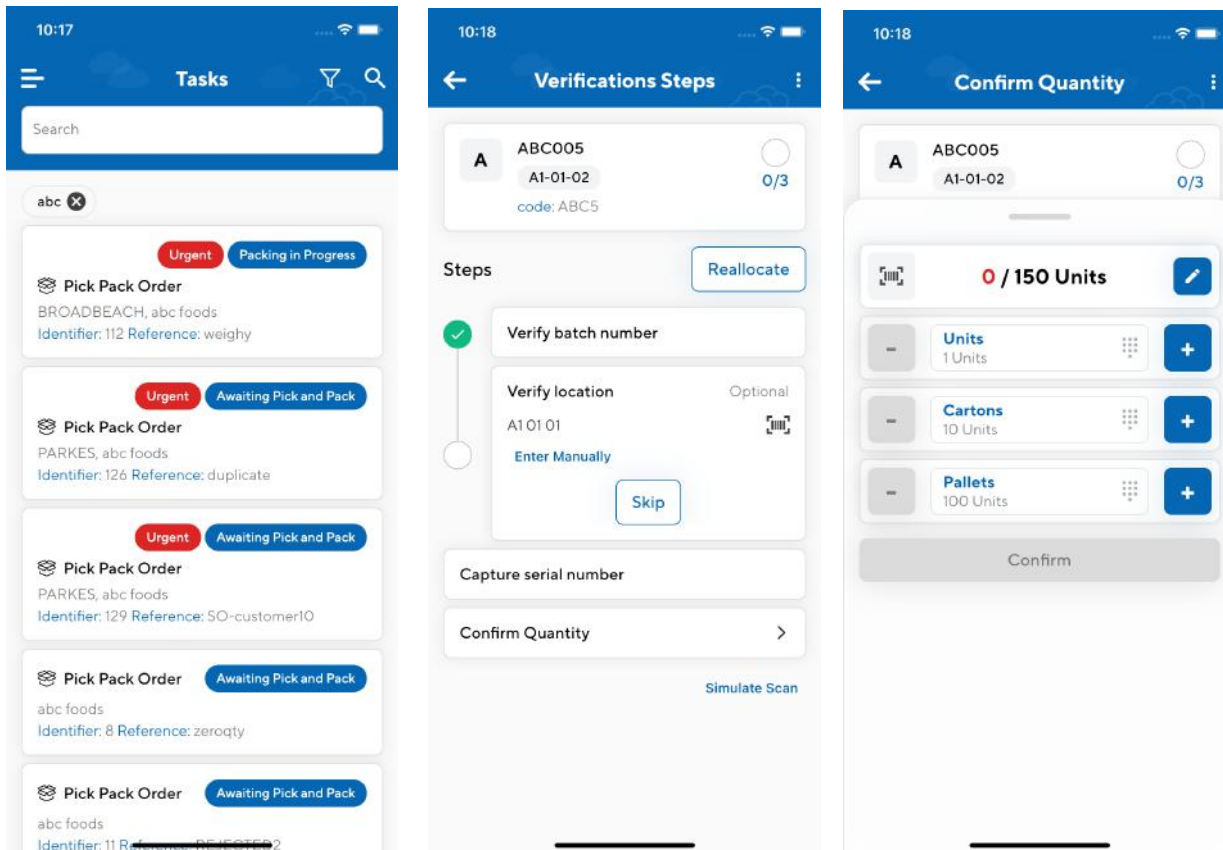
After gaining some experience with Flutter prior, my role was to initially investigate the potential of integrating Flutter into the current iOS and Android apps, and to see if it was a viable solution for new features moving forward. It turned out to be successful and as a result I moved to Flutter full time and a project began to start moving new features over to flutter, using a new api.

For this, I developed an architecture using modern approaches and popular libraries that also supported the existing integrations of the current apps (Bluetooth scanners, google analytics, Firebase etc). It was also important the solution was future-proofed as much as possible for future expansion. The UI and screens for the new app were also developed by me using XD.

Some of the core libraries used in the solution were:

- Riverpod
- Freezed
- Hive Database

- Retrofit



Above: The Cartoncloud Flutter application, showing the various screens involved in packing a warehouse order

Trade Me March 2018 - August 2020

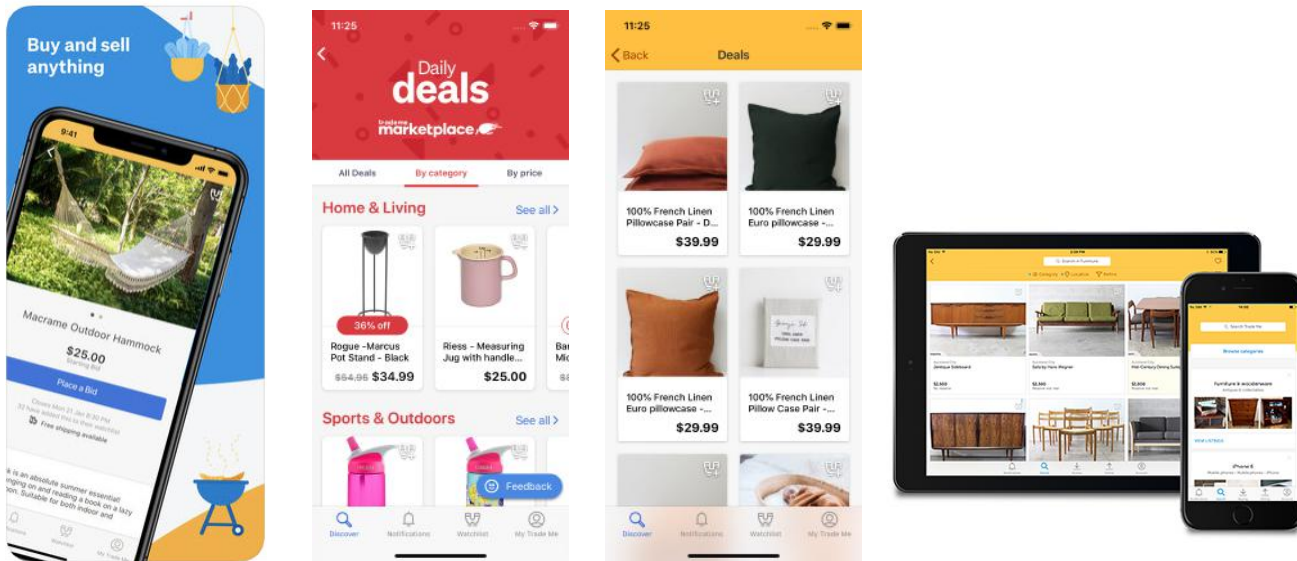
Trade Me is an online marketplace that was originally a way for people to sell their second-hand goods, now it supports new goods, job advertisements and services as well. My primary responsibilities were on the `yellow app`, which is Trade Me's core mobile application. All new code was written in Swift using MVVM architectures and RxSwift, this was happening on top of an original Objective-C code base. I participated in code reviews using BitBucket pull requests, this review process was used to add new features and to ensure code quality. My team introduced numerous new features to the app, including promoting store sellers in an effort to sell more new goods, this was an area where support had been light. We also initiated the new "Daily Deals" feature which was an entirely new section where new good sellers could post their latest deals - using a newly added API.

We ran many experiments (A/B tests) on the application in an effort to find and reduce friction in the user experience. Some of the experiments failed and some resulted in changes being rolled out to all users. One such feature included displaying the "Afterpay" price for *supported* listings in the search results. We discovered this small change increased buyer conversion by 10%.

On top of the iOS development, I also built a ruby on rails application as an internal tool that helped people view and manage the daily deals consumed by the app in a user-friendly way.

Squad Master at Trade Me - I was nominated to be the squad master by a team of 8 members. This was a leadership-focused role that involved representing the team when required, facilitating sprint planning and retrospective sessions, and also being another point of contact for any project-related issues. During my time as squad master, I refined

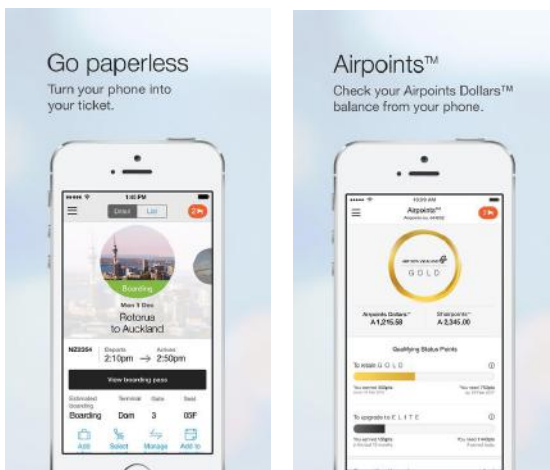
the teams' sprints into 2-week time frames and introduced velocity measuring. This way we were able to make accurate estimates for our work while also keeping everyone's workloads manageable.



Above: Screenshots of the Trade Me application

Air New Zealand Contactor July 2016 - July 2017

I joined Air New Zealand as a Senior Contract iOS developer to work on their iOS mobile app. The code base was very large and had good test coverage. Any new code was written with an emphasis on code quality and testing, because of this I gained a lot of experience writing unit and automated tests and participating in code reviews using *crucible*. Some key technologies used for the automated tests were *calibash* and *ruby*. These tests were part of a continuous integration pipeline initially running on *Jenkins*, but later moved to *Go*. I worked on many different aspects of the app including UI enhancements and maintenance on outdated libraries including RestKit and Cocoapods. I also added some new features including 'in-app ratings', that allowed the user to submit a rating and a comment to us directly from the app. It was very successful and is providing a constant stream of ideas and opinions from the user base. My last task there was to develop multi-PAX boarding passes, this allowed families or a group of friends to scan onto their flight together using a single device.



Above: Air New Zealand's application.

Find my time April 2016 - December 2017

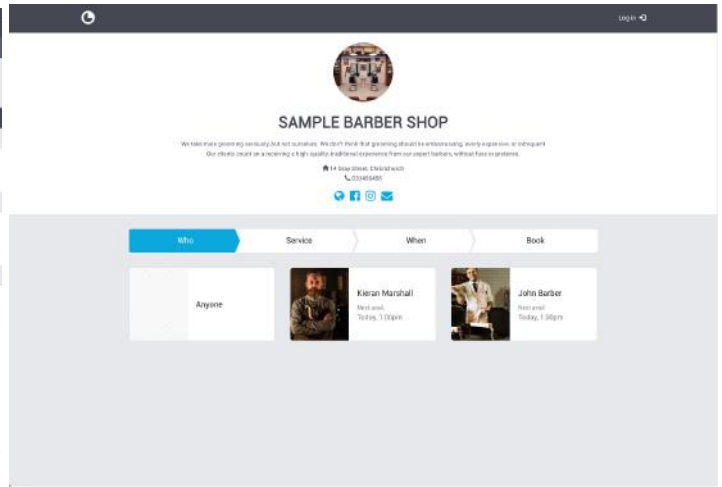
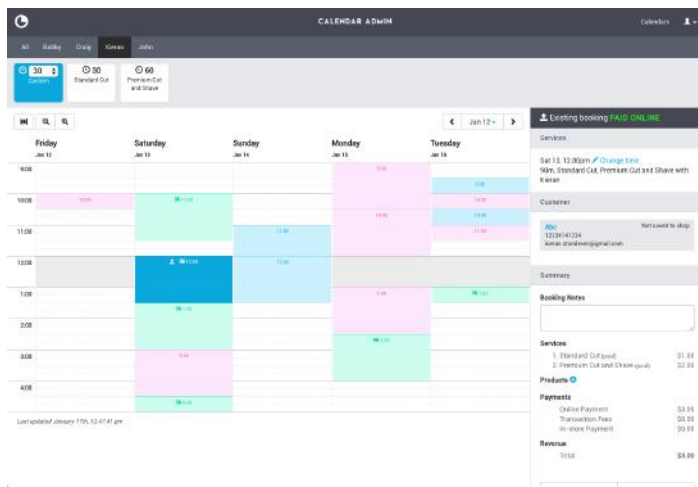
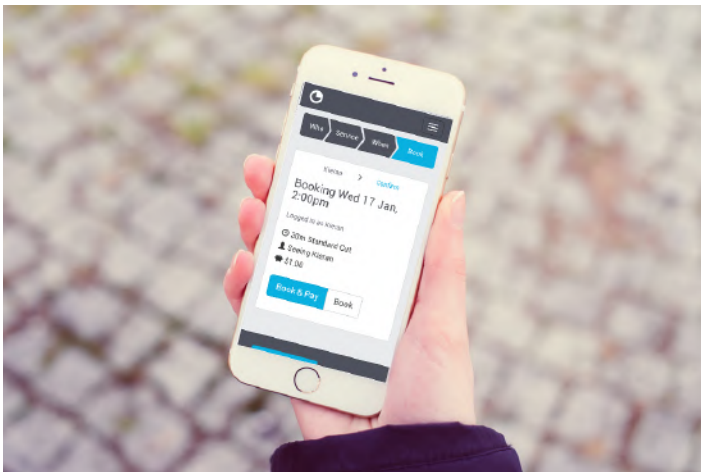
This was a project I started to gain some web development and business experience. The site is built on Ruby on Rails, uses a Postgres database and is hosted on Heroku. A large amount of javascript was also written to power the calendar's user interface.

Some key features of the site included

- Runs on all devices (responsive web)
- Supported SMS booking reminders
- Online payment support using 'Stripe'
- Credit card billing using 'Stripe'
- Shop revenue reporting using 'Google Charts'
- Customer management and reporting
- In-store product sales

Find my time was designed to be a modern solution to online booking for any sole trader or business that provides services in allotted time frames. It hosts all of its businesses on a single website, in this way it could be the first place someone goes when they want to find a barber, personal trainer, beautician, or any other service in their area.

This project is no longer live, but you can view the demo site at this link <https://safe-anchorage-demo.herokuapp.com/>

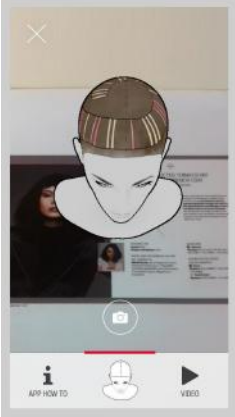






Above: The Find my time demo web application.

Motim Technologies (Christchurch) July 2014 - March 2016

My role at Motim was to specialise in iOS, maintaining existing applications and developing new applications. I gained some augmented reality experience here working with Vuforia's libraries. All development was native and many of the applications had backends, supported push notifications and included app analytics. Some of the applications I worked on include **Goldwell**, **Skrillex Alien Ride**, **Hummingbird Coffee**, **Big Sean** and **Rinnai Fire**. Not all of the apps are still maintained by Motim but some are still active.

iOS Apps worked on at Motim

Goldwell	Skrillex Alien Ride	Hummingbird	Big Sean	Rinnai Fire
				

Geoop/Geo Workforce Solutions (Auckland)

February 2014 – July 2014

Geoop is a job management tool for mobile platforms that allow businesses to manage their staff, clients, jobs and finances. The mobile applications are additionally equipped with GPS tracking and job location services to allow businesses to keep track of their staff and job locations.

My initial tasks at Geoop were maintaining the existing version and adding new features to it. I am now involved in the design and implementation of the new application which is built from the ground up. The goal is to design an application that is more stable, user-friendly and takes advantage of the new operating systems.

M-Com/Fiserv (Auckland) August 2011 – September 2013

Started in September 2011 as a mobile developer specialising in iOS. I gained my initial Android development experience here, as well as a large amount of JavaScript/Web experience.

ANZ

From November 2012 until April 2013 I have been working on the new ANZ iPhone application that has been re-written and re-designed from the ground up. This project has given me the opportunity to help design the architecture of an application and write new, highly specialized UI components (including a 'card spinner control', '3D cover flow' and a 'transaction search tool').

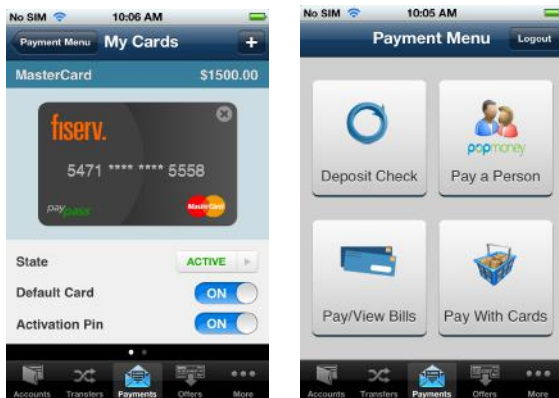


Above: The ANZ application (at the time I was working on it).

NFC prototype

Starting in May 2012 I was assigned to a new Proof of Concept project as the sole Developer. My task was to explore the uses of NFC (Near Field Communication) and demonstrate its ability to complete a contactless payment using a mobile device. The target device was an iPhone equipped with an external NFC sleeve. Two existing companies (ABNote and ViVOTech) had already started development in this field and I collaborated with both to build the prototype. I was able to integrate their libraries into our existing software with complete functionality.

Once the prototype was complete, I presented it to clients in meetings and to the company in the monthly meeting (using a big screen projector to show it in action).



Above: The completed NFC prototype. The MasterCard shown is test card downloaded from the ViVOTech server, an NFC Card reader can read the account details from the iPhone device and complete a transaction.

Scotiabank

When I first arrived at M-Com (September 2011) I was assigned to work on an iPhone application for Scotiabank (Canadian bank). I added numerous new payment features to the application in a large release cycle and worked closely with a team of web, platform and mobile developers.

Initially the application was built in PhoneGap (a mobile framework used to build html rendered applications), however Scotiabank wanted to see more Native functionality. I was able to merge numerous native screens into the application by making many changes to the architecture and generally improving the performance and the user experience.



Above: The Scotiabank application, allows users to access their accounts to make payments, check balances, find their nearest atm branch and more.

My Apps

- **Pin Masters**

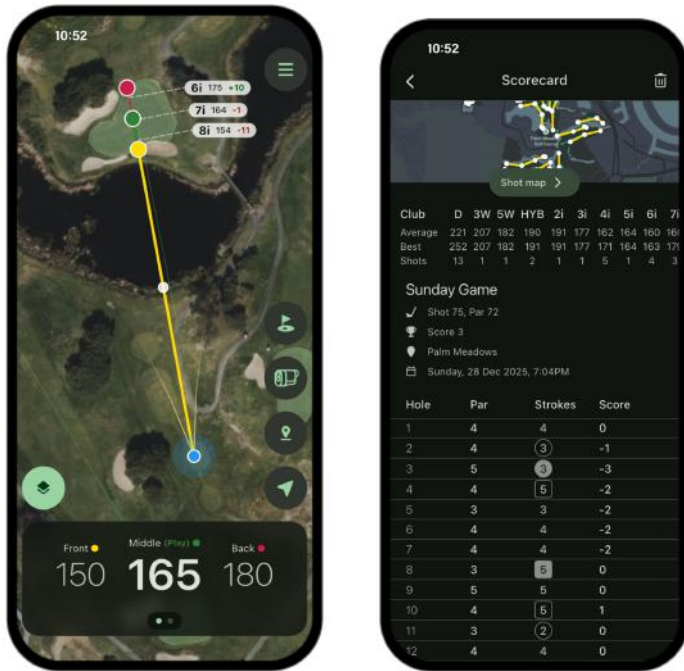
An application built to be used on the golf course. I struggled to find an application that was able to provide basic distance measurements without also requiring an account, paying a subscription, or navigating through a complicated interface. I built this application as a remedy to those issues, for myself, but it is also available on iOS and Android for others to use for free.

The key feature is being able to open the app anywhere on the golf course and it will automatically show you where you are and how far it is to the hole without needing to interact with it.

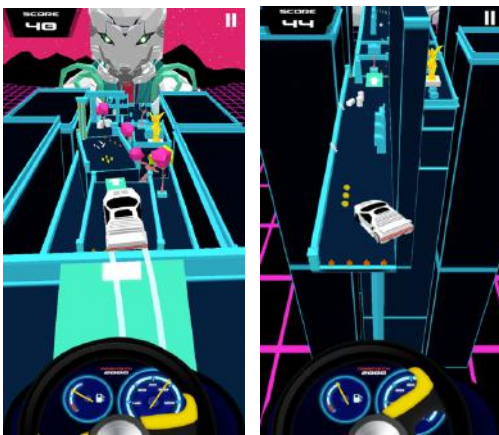
Application homepage: <https://pinmasters.app>.

Key technologies:

- Riverpod state management
- Hive database
- Flutter map (fleaflet)
- Mapbox for map image hosting and custom styling
- Open street maps (or OSM) for golf course GPS data
- QGIS for managing golf course mapping data

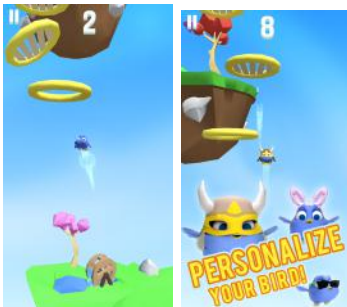


- **Sky Driver iOS (Unity)**



Above: Sky driver, a swipe controlled racing game built in Unity released late 2018., The aim is to survive as long as you can, while keeping your gas topped up and collecting coins to score points.

- **Happy Hoops iOS (Unity)**



Above: A game I released in September 2016. This is a simple infinite-gameplay game that I built in Unity. Most of my time and effort went into the 3D artwork using 'blender'. This game started as an extremely basic prototype that turned out to be very successful with my flatmates and friends.

- **Spirit Ball iOS (Unity)**



Above: A game I finished in February 2016 using the Unity game engine, it is similar to Greedy Bones (below) but dressed up with many more features. I worked with in-app purchases and advertisements for the first time as a way of monetizing a free game (now it is entirely free though). I am planning to release it on Android as well.

- **Greedy Bones iOS (Cocos2d)**



Above: Rotate the maze to roll the character around and collect all the coins. There are various traps and obstacles set up to create a mix of problem solving and action gameplay.

I created the game artwork and the maps (stored in XML) using my own custom map builder application. Cocos2d and Chipmunk power the graphics and physics and Photoshop CS5 was used for the artwork.

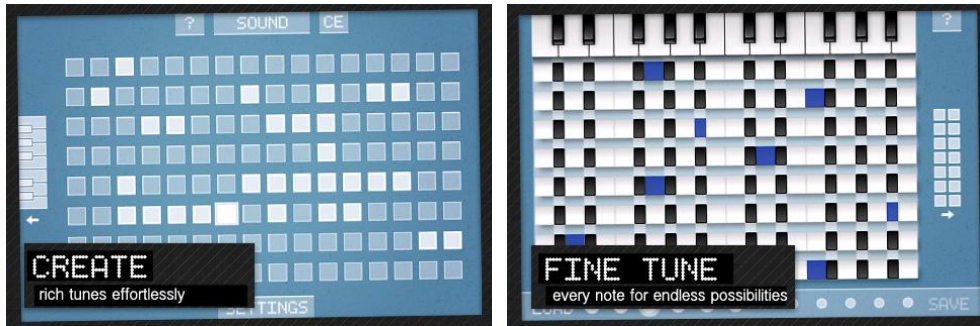
- **Soccer Mayhem iOS (Cocos2d)**



Above: Control the streakers to avoid the security guards, score goals, grab fruit power-ups and streak for as long as you can before exiting the stadium.

Cocos2d powers the game and it also features a physics engine so the soccer ball can roll around bouncing off players.

- GridDJ (iPhone application)



Above: A music application that allows the user to build melodies using a tone matrix. Being a relatively complex concept I pushed for usability and a fast, clean looking application. I designed all the graphics used in the application using Photoshop CS5.

Nuffie Productions/CrichHQ (Wellington) November 2010 – June 2011

I was part of a large project that built an iPhone application used to score cricket, it was named 'CrichHQ'. I developed libraries to help display cricket statistics in graphical formats such as 3D wagon wheels and bar graphs.

Zodal (Christchurch) October 2008 - April 2010

I worked for Zodal during University studies, full time in summer and part time during the year. Projects there included an iPhone application named 'mPass' for Air New Zealand, an iPhone game named 'Tiny Tint Terrors' and a web service for finance application called 'Link Marketing Services'.

The original Air New Zealand iOS application (mPass):



Above: The first version of mPass, where I initially started developing iOS applications and gained valuable experience working with a team to release a major application.

My main contributions to the application were in the design of the network layer that communicated with Air New Zealand's web service (using JSON) and building the encrypted barcode (boarding pass) page. I also helped with the design of the sql database.

Education

- Primary School Education, Saint Teresa's Private School, Wellington.
- 2005 NCEA level 3, University Entrance, Timaru Boys' High School.
- 2006 Diploma of Multimedia, Natcoll Design Institute, Christchurch.

- 2010 Bachelor of Science Majoring in Computer Science, **University of Canterbury**.

Skills

- Flutter, Dart
- Apple Development: extensive experience in iOS development in Swift and Objective-C.
- Android Development
- Ruby on Rails / javascript: Building and designing web applications
- Unity: C# game development.
- JIRA, Confluence, Bitbucket
- Agile processes
- Web: HTML, Javascript, CSS/SCSS.
- Programming: theoretical and practical using Java, C, C#, C++, Cocoa/Objective-C.
- Human and Computer Interaction: fundamentals of human behavior and interface design.
- Multimedia: Practiced skills in 3D Modeling, animation, interactive design and graphic design.
- Data Structures and Algorithms: theoretical and practical knowledge of computer science fundamentals.
- Computer Graphics: theoretical and practical using OpenGL.
- Software Engineering: group projects, group dynamics and project lifecycles.