Designing, testing, and implementing a production process for rapidly prototyping software applications.

TIM Field Study under Prof. Subhas Desa

Ralph Miller

March 23nd 2023

VANTAGE

Contents

1 Executive Summary

Introduction:

In today's fast-paced software industry, companies are under increasing pressure to quickly develop and prototype software applications to stay competitive. This executive summary presents key findings from a study on the use of low-code tools and agile methodologies to design, test, and implement a production process for rapidly prototyping software applications.

Problem:

The traditional software development prototyping process can take 3-4 months to complete, which is too slow for many companies who need to quickly test and validate their ideas in the marketplace. This study explores alternative methods for rapidly prototyping software applications to reduce development time and cost.

Approach:

In order to find the best practice for rapidly prototyping software, A structured process was created in order to solve each different sub-goal of the problem statement. A cycle of breaking down each goal, researching the best tools, identifying solution methods, implementation of these methods into a working prototype for the customer, and drawing relevant conclusions about the method used to solve the goal.

Key Findings:

The key findings of this study indicate that using low-code tools and agile methodologies allows software companies to rapidly prototype software at a significantly lower cost compared to traditional methods. By implementing these techniques at Vantage the prototyping time was reduced from 3-4 months to just a few weeks. The use of low-code tools also allows the engineering team at Vantage to develop software applications with fewer resources and less specialized skills.

Conclusion:

In conclusion, the findings of this study demonstrate that using low-code tools and agile methodologies for rapid software prototyping is an effective way for software companies to reduce development time and cost. Companies that adopt these methods can quickly test and validate their ideas, stay ahead of the competition, and ultimately achieve greater success in the marketplace.

2 Introduction to Vantage

Company Background:

Vantage is a technology company based in Vancouver B.C and Longmont Colorado, that combines software and hardware to create a better guest experience for their customers and track data and present it in software tools to allow their customers to make better business decisions. Their customer base are mainly amusement park companies, i.e. water parks, surfing wave pools, hotels and resorts who contract Vantage to provide software and hardware for a better water park guest experience. Examples of Vantages hardware and software solutions are; Mobile Applications, Physical Kiosks, Wristband Sensors and Stations, and Operator Terminals.

Below are the 4 main business solutions that Vantage offers to its customers:

Guest Experiences	Vantage seeks to make the guest experience unique to make it memorable, to simplify the guest journey for a more seamless experience, to allow guests to spend less time waiting and more time playing, and to provide visitors peace of mind with a safer park environment.
Operations Management	Vantage seeks to take the pressure of the customers' team and minimize human error, to stay on top of the customers' venue conditions in real time, to regulate crowds by influencing guest behaviors, and to determine the customers staffing sweet spot and motivate employees.
Strategic Planning	Vantage seeks to help the customer discover unknown revenue opportunities, to help the customer identify areas of improvement in real time, over time, and ensure alignment with a birds-eye view of all departments.
Marketing	Vantage seeks to engage guests before their visit and long after they have left, to increase ancillary sales during guest visits, and to earn rave reviews by modernizing the guest experience.

2.1 The Customer

The Atlantis Dubai is a luxury hotel and resort located on the man-made Palm Jumeirah island in Dubai, United Arab Emirates. The hotel offers a range of luxurious accommodations, including guest rooms, suites, and underwater suites. The Atlantis Dubai is a popular tourist destination and has won numerous awards for its architecture, accommodations, and entertainment offerings.

Aquaventure is a water park located at the Atlantis Dubai resort on Palm Jumeirah island in Dubai, United Arab Emirates. The water park is one of the largest in the Middle East and features a variety of water slides and attractions for all ages, as well as a private beach and several swimming pools.

Atlantis Dubai has contracted Vantage to create a SAAS platform for their waterpark. This includes:

1. Guest Mobile Application
2. Operator Mobile and Web Applications
3. Kiosk Terminals
4. GPS Stations
5. Wristband System

2.2 My Role

I have been working at Vantage as an intern since July 2022. I work in the engineering team under the Head of Engineering who will be the sponsor for the field study. I help the team by rapidly prototyping mobile applications using a mix of programming and low code solutions like FlutterFlow built on the Flutter framework. The main product category I have worked in is Guest Experiences.

I have worked on two projects at Vantage so far, the first project was a mobile app for a Hotel and Resort in Dubai, Aquaventure, which fell under the category of Vantage's Guest Experience products. During this project I worked closely with the Head of Engineering to understand the structure, function, details, and deliverables of the project. I created from scratch a working application to solve the Guest Experience problem at Aquaventure.

3 Designing, testing, and implementing a production process for rapidly prototyping software applications

3.1 Objective

Develop and demonstrate the best practice for rapidly prototyping applications in a way that;

(A) Accelerates delivery, reduces complexity, and lowers cost.

(B) Ties together multiple clients with the same backend, understands and fulfills customer requirements and architecture.

(C) Develops a deployment model to distribute application changes downstream with continuous delivery.

In order to create a high-quality Guest Experience product at Vantage for the customer Atlantis Dubai.

3.2 Approach

In this field study, I will be researching tools and practices in order to create a working prototype of the Aquaventure Guest and Operator mobile applications for Vantage's customer Atlantis Dubai. In order to do this each goal in the objective is broken down into the three distinct parts listed above. The approach I will be using in order to find a solution to each part of the objective will be as follows.

1. Break down what each goal means, define technical vocabulary.
2. Conduct research pertaining to the goal.
3. Identify a method to solve the stated goal.
4. Implement the method, including diagrams, pictures, ect.
5. Draw relevant conclusions about the method/tool used to solve the stated goal.

4 Part A: Accelerating delivery, reducing complexity, and lowering cost.

4.1 Breakdown of Goal

In this section the goal is to find a tool and method in order to accelerate the delivery of a mobile application prototype by reducing complexity and lowering cost.

4.1.1 Technical Vocabulary Defined:

1. Dart - a programming language designed by Lars Bak and Kasper Lund and developed by Google. The programming language is designed for client development such as for the web and mobile apps, and it can also be used to build server and desktop applications.
2. Flutter - a mobile app development platform created by Google built upon the Dart programming language.
3. low-code - a low-code development platform provides a development environment used to create application software through a graphical user interface.
4. FlutterFlow - a low-code development platform built upon the Flutter framework used to create mobile and web software applications.

4.2 Research/Solution Method

According to the article, How Long Does It Take to Build an iOS or Android Mobile App? it can take anywhere from 3 - 4 months for an application prototype to be built and deployed to IOS / Android. This is very time intensive and monetarily expensive. Vantage currently outsources some of its prototyping to teams overseas. The cost of a 3 month contract for an oversea team is around $40,000.

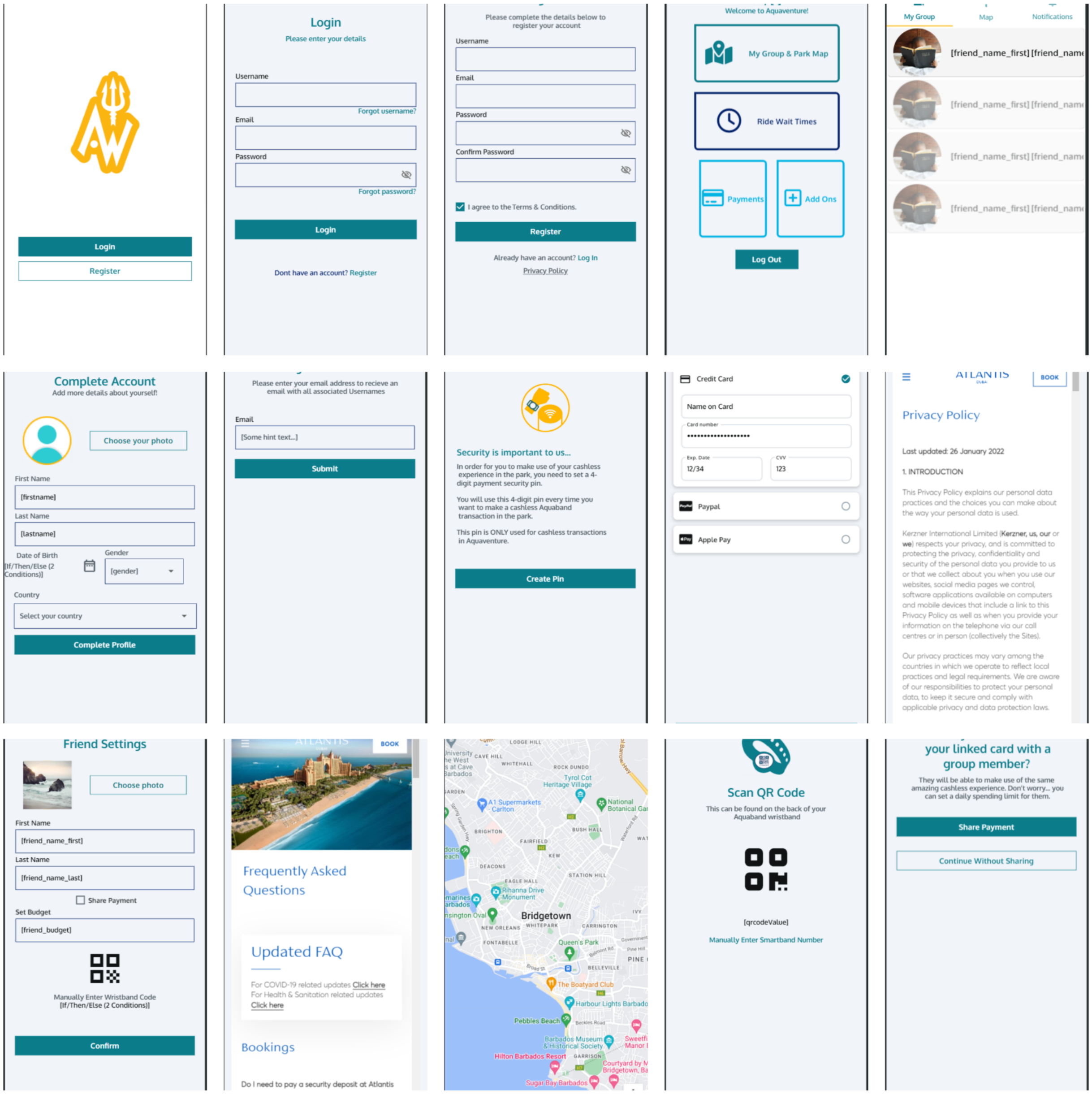
I have researched and found a platform for rapid development of mobile and web applications that could solve this prototyping problem called FlutterFlow. It is a low-code platform built upon Googles Flutter framework. It allows the developer to build mobile applications quickly and deploy them all from one terminal. It uses features such as a graphical app builder (building widgets, screens, ect without code) database connection, and allows the user to export the Dart code for further customization that FlutterFlow currently doesn't support.

4.3 Implementation

4.3.1 Accelerating Delivery of a Prototype:

At Vantage, I have worked on a project in which I prototyped a basic mobile application for the customer, Atlantis Dubai, in around 10 days. By using FlutterFlow, the low-code solution to mobile app development, I was able to build a prototype application that satisfied the customers requirements. Below are the requirements and screenshots of the prototype application I developed.

Aquaventure App (Project 1) Features Developed	Description of added features
Database Connection	• Google's Firebase DB platform • Store user-data (name, dob, email, wristband id, country of origin)
User Authentication and Registration	• Creation, deletion, and change of user profiles • Users can sign up via email • Uses secure authentication tokens
Add Friend to Party	• Creation, deletion, edit friend profiles • Share payment within party
Add Payment	• Add payment method to be used with wristband • Ability to share payment with friends in party
Connect Wristband	• Ability to connect a person specific wristband which is used for tracking, guest movement, ride wait times, purchasing food drinks ect.



Please complete the details below to register your account

Please enter your email address to recieve an email with all associated Usernames

Security is important to us...

In order for you to make use of your cashless experience in the park, you need to set a 4-digit payment security pin.

You will use this 4-digit pin every time you want to make a cashless Aquaband transaction in the park.

This pin is ONLY used for cashless transactions in Aquaventure.

Privacy Policy

Last updated: 26 January 2022

1. INTRODUCTION

This Privacy Policy explains our personal data practices and the choices you can make about the way your personal data is used.

Kerzner International Limited (**Kerzner, us, our** or **we**) respects your privacy, and is committed to protecting the privacy, confidentiality and security of the personal data you provide to us or that we collect about you when you use our websites, social media pages we control, software applications available on computers and mobile devices that include a link to this Privacy Policy as well as when you provide your information on the telephone via our call centres or in person (collectively the Sites).

Our privacy practices may vary among the countries in which we operate to reflect local practices and legal requirements. We are aware of our responsibilities to protect your personal data, to keep it secure and comply with applicable privacy and data protection laws.

For COVID-19 related updates Click here
For Health & Sanitation related updates Click here

Do I need to pay a security deposit at Atlantis

This can be found on the back of your Aquaband wristband

your linked card with a group member?

They will be able to make use of the same amazing cashless experience. Don't worry... you can set a daily spending limit for them.

Aquaventure App Screens, built in FlutterFlow

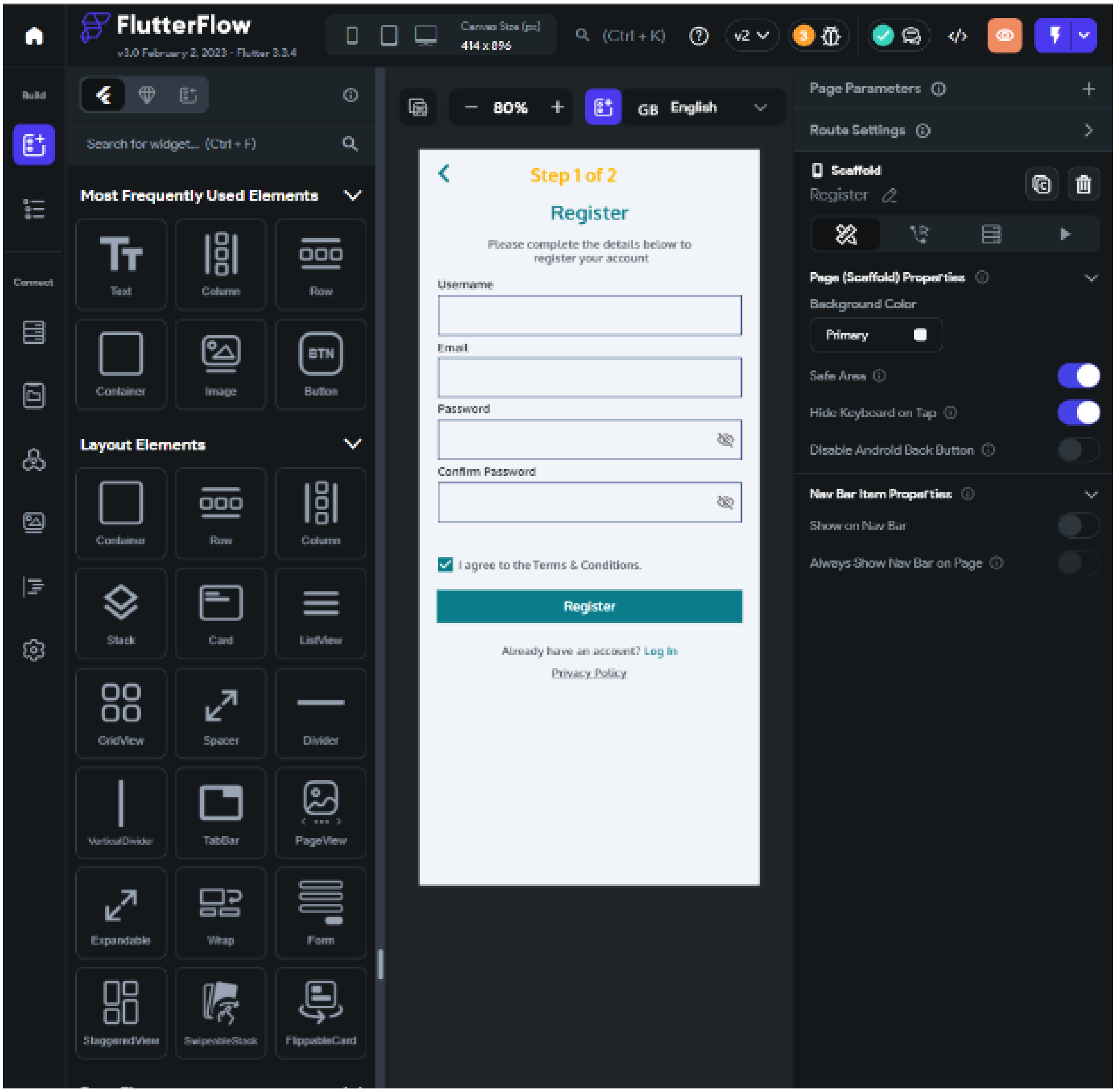
4.3.2 Reducing Complexity of a Prototype:

In programming terms, complexity of algorithms can be derived in Big O notation, but for projects at scale complexity can be derived from many different sources; I believe that using a low-code/no-code solution such as FlutterFlow can solve some of the problems that cause complexity when creating a Prototype.

4.3.3 Technology:

FlutterFlows slogan is “Build applications faster than ever, Create beautiful UI, generate clean code, and deploy to the app stores or web in one click. Fully extensible with custom code.” It is a technology that uses an easy to use graphical interface to design app widgets and functionality. This reduces the complexity of design by magnitudes, as the programmer is no longer designing with code, but visually akin to drawing. This opens doors to employees who are not as specialized in programming but are familiar with app design to be able to create application prototypes.

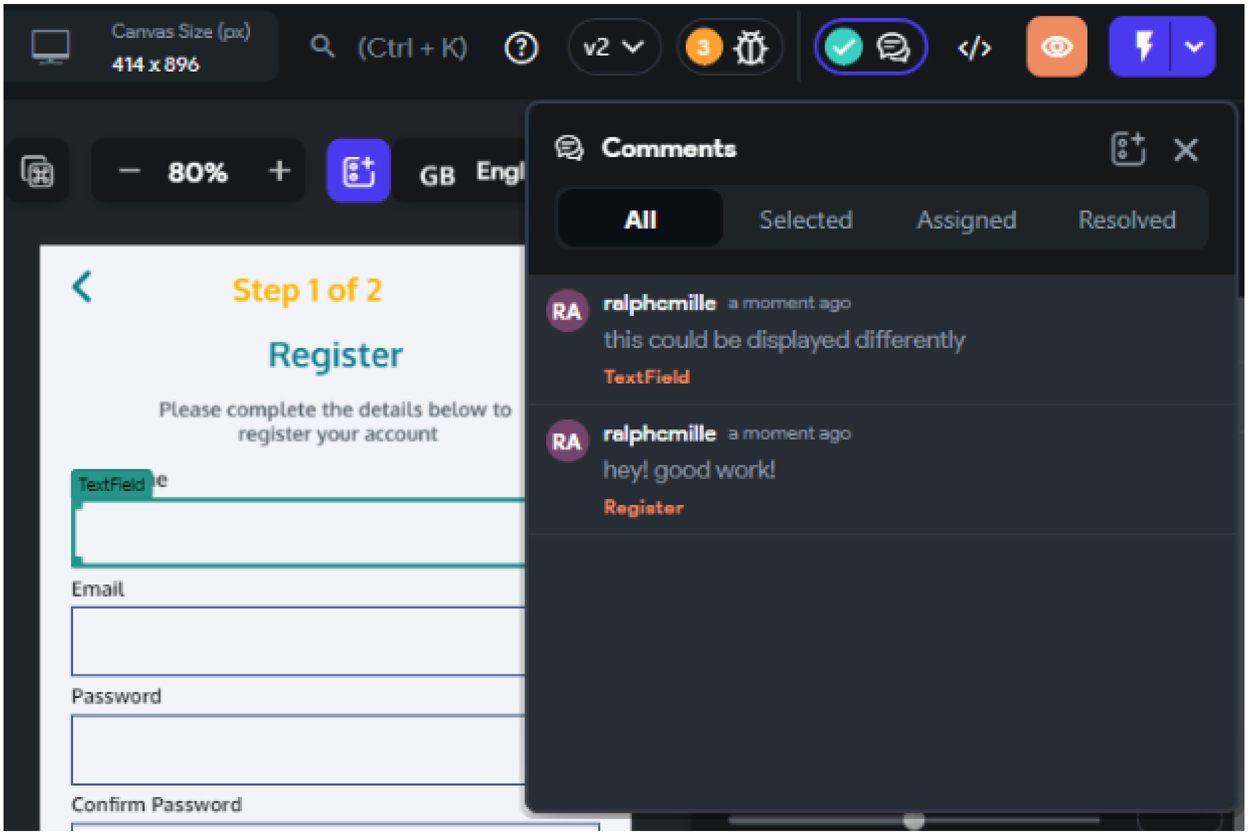
Below is an image showing the graphical tool used to create an app page in FlutterFlow.



FlutterFlow's Graphical Tools.

4.3.4 Communication:

FlutterFlow has two main tools for communication within its platform, one commonly used tool is GitHub for source control, employees can make application changes in FlutterFlow and then push changes to a GitHub repository where a project manager can review the changes and merge them into the main project prototype. Another tool of communication within the FlutterFlow platform is the commenting feature which allows team members to comment on specific widgets allowing for certain issues to be easily resolved. An example of this is shown below.



RA ralphcmille a moment ago
this could be displayed differently
TextField

RA ralphcmille a moment ago
hey! good work!
Register

Communication tools built into FlutterFlow.

4.3.5 Interdependencies:

A large part of causing complexity within a project is multiple different interdependencies and the configuration of these dependencies. FlutterFlow has a solution for this. All of the interdependencies are built into the platform in a low-no-code fashion. This means most integrations are working and available after little to no configuration. This includes support for;

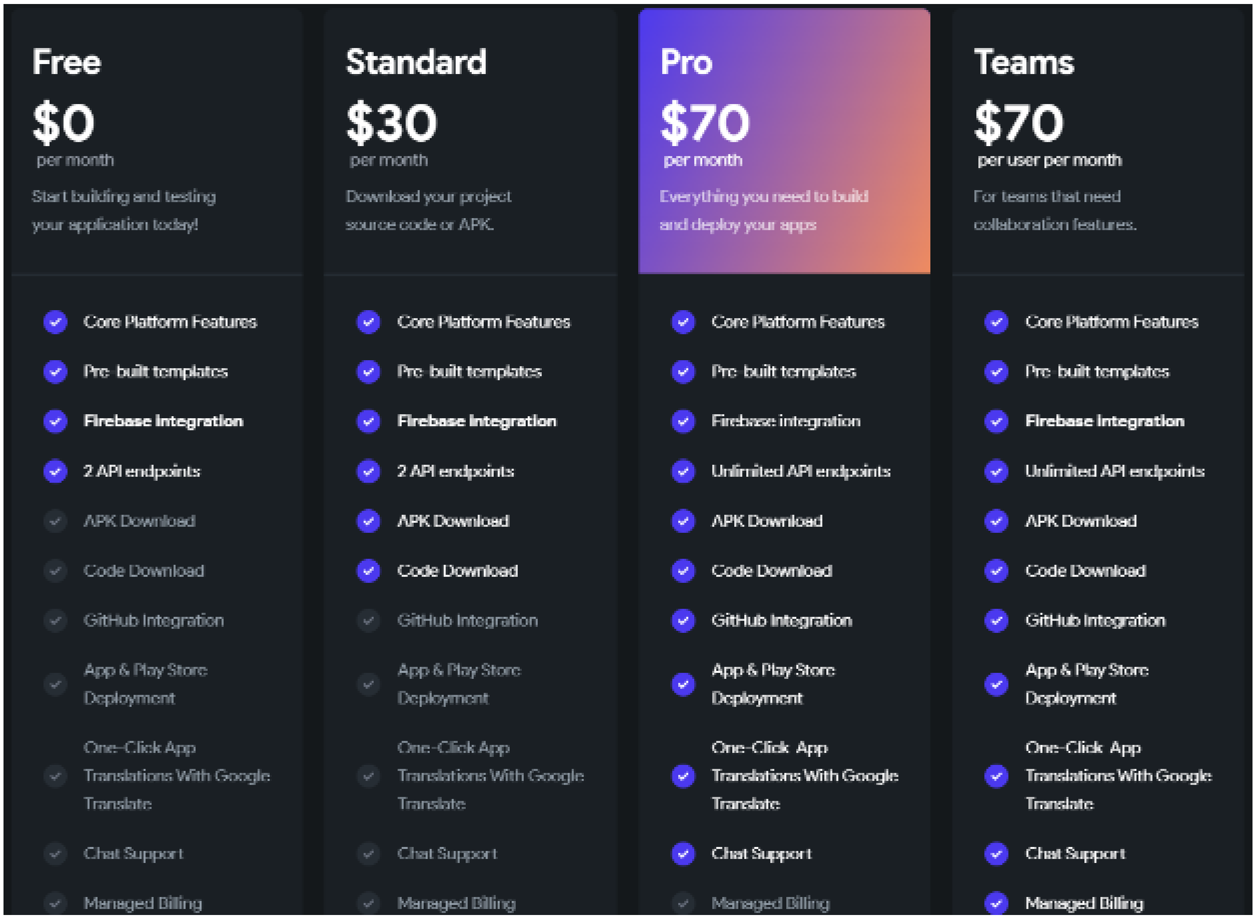
- Database Configuration with Google Firestore.
- Privacy and Security App Permissions.
- Authentication
- One click, Mobile and Web Deployment
- Stripe, Braintree and RevanueCat Integration for InApp Purchases
- Supabase DB Integration
- GitHub
- Google Analytics
- Remote Configuration
- Google Maps
- Ad Mob

- And many more...

4.3.6 Lowering Cost of a Prototype:

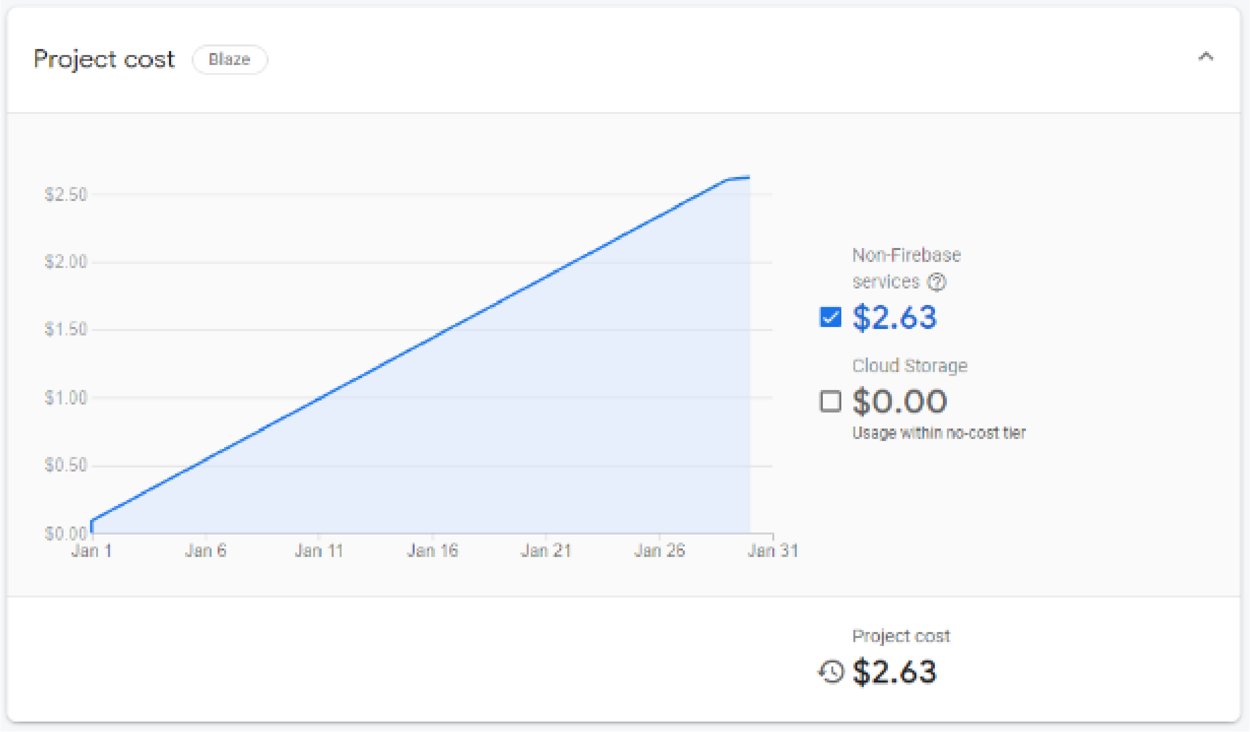
A large barrier to creating many prototypes is cost. If something is able to be created cheaper, and quickly then the quality of a prototype and its effect on the main product are increased. FlutterFlow and its integrations are inherently very inexpensive.

A business plan to use FlutterFlow for a team comes out to $70 per team member per month. For Vantage, our team of two, that are using this service to rapidly prototype applications, only costs $140 per month, or $1680 per year. Other misc costs such as database storage are very inexpensive as well, over the last 6 months of prototyping with these tools, it has cost a whopping $2.63 in addition to paying for FlutterFlow.



Free	Standard	Pro	Teams
$0	$30	$70	$70
per month	per month	per month	per user per month
Start building and testing your application today!	Download your project source code or APK.	Everything you need to build and deploy your apps	For teams that need collaboration features.
✓ Core Platform Features	✓ Core Platform Features	✓ Core Platform Features	✓ Core Platform Features
✓ Pre-built templates	✓ Pre-built templates	✓ Pre-built templates	✓ Pre-built templates
✓ Firebase Integration	✓ Firebase Integration	✓ Firebase integration	✓ Firebase Integration
✓ 2 API endpoints	✓ 2 API endpoints	✓ Unlimited API endpoints	✓ Unlimited API endpoints
APK Download	✓ APK Download	✓ APK Download	✓ APK Download
Code Download	✓ Code Download	✓ Code Download	✓ Code Download
GitHub Integration	GitHub Integration	✓ GitHub Integration	✓ GitHub Integration
App & Play Store Deployment	App & Play Store Deployment	✓ App & Play Store Deployment	✓ App & Play Store Deployment
One-Click App Translations With Google Translate	One-Click App Translations With Google Translate	✓ One-Click App Translations With Google Translate	✓ One-Click App Translations With Google Translate
Chat Support	Chat Support	✓ Chat Support	✓ Chat Support
Managed Billing	Managed Billing	Managed Billing	✓ Managed Billing

FlutterFlow Pricing.



Database storage costs over ∼ 6 months.

4.4 Relevant Conclusions

In conclusion, rapid prototyping using software applications tools that accelerate delivery, reduce complexity, and lower cost is an essential part of the software development process. These tools allow software development teams to quickly test and validate their ideas, reduce time-to-market, and improve overall product quality. With the advent of new technologies and methodologies, such as low-code tools like FlutterFlow, rapid prototyping has become even more effective and efficient. By leveraging these tools and methodologies, businesses can rapidly innovate and stay competitive in an increasingly fast-paced and dynamic market. Therefore, it is imperative for software development teams to adopt these rapid prototyping tools as a key practice to deliver high-quality software applications that meet the customer needs and expectations.

5 Part B: Tying together multiple clients using the same back end. While also incorporating customer requirements and architecture.

5.1 Breakdown of Goal

In this section the goal is to find a tool and method in order to have the ability to easily access the same backend with more than one mobile application. For this project the customer requests that two applications be built, a guest application and an operator application. In order for these two applications to communicate, we need a backend that ties together multiple clients.

5.1.1 Technical Vocabulary Defined:

1. Frontend and Backend - refer to the separation of concerns between the presentation layer (frontend), and the data access layer (backend) of a piece of software, or the physical infrastructure or hardware. In the client–server model, the client is usually considered the frontend and the server is usually considered the backend, even when some presentation work is actually done on the server itself.
2. Document Reference Database - document database is a type of nonrelational database that is designed to store and query data as JSON-like documents.
3. Firestore Database - Firestore is a NoSQL document database built for automatic scaling, high performance, and ease of application development.
4. Operator Application - Mobile application for waterpark operators.
5. Guest Application - Mobile application for waterpark guests.

5.2 Research

There are many options for a database management solution. A modern database is usually cloud based instead of on site. There are several commercial database solutions out there such as solutions provided by Oracle, AWS, ect. For this application, FlutterFlow offers great and inexpensive pay as you go integration with Google Firestore for managing the database of a mobile application that is integrated into the software platform.

5.3 Solution Method

A clear choice was made for which database service to use in order to efficiently and easily tie together clients with the same backend. We chose to use Google's Firestore database which is based off of a document reference database system. Which is integrated easily into Flutter, and is cost effective. On the Aquaventure project one example of tying together multiple clients using the same backend that I have worked on is for the Operator Control Access that allows for guest admittance at certain admittance gates within the waterpark at Atlantis Dubai.

5.4 Implementation

We have two application clients, the guest mobile application, and the staff operator terminal. Both of these clients use the same registration database for creating guest and staff accounts. In order to login to the Operator Terminal an account must be flagged to be of Operator Privilege, this is checked against the profile data and if the user is an operator they are let in, if they are not the user is denied access.

The process for generating a ticket is done via the Guest Application where the guest purchases a ticket and then this ticket is assigned to the guests ID and stored in the database. Then when the guest approaches the ticketing gate a aquaventure staff uses the operator terminal application to scan and admit the guest. This is an instance of using two separate mobile applications that both talk to the same backend.

5.4.1 Logical Process for Ticket Entry

The logical process for what I explained in the previous section is as follows:

Step	Backend	Data
1) Guest Purchases a Ticket	a unique redeemable ticket is created in the firestore database under the guests unique ID	• (Data Created) • TicketID • GuestID • Date/Time Purchased • Day of Access
2) Operator Scans Ticket at Gate	the unique redeemable ticket is passed to the database and checks are done to ensure - one this is a valid ticket that exists in the database - this guest ID belongs to the ticket	• Ticket Information
3) Ticket Data is Moved	• A Transfer of information occurs. • the ticket is removed from the active ticket database • an event is stored into the database logging the entry.	• (Data Transferred to Event DB) • Date and Time • Location of Admittance • guestID of admitted guest • operatorID of staff member who scanned and admitted ticket

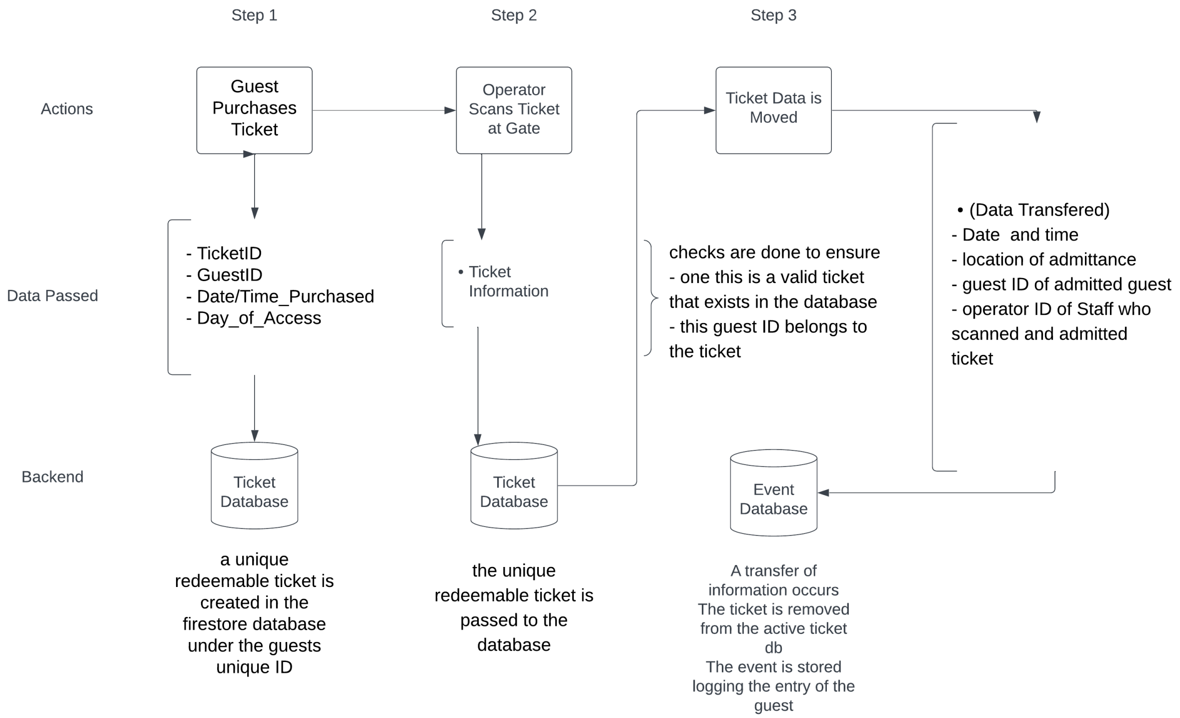
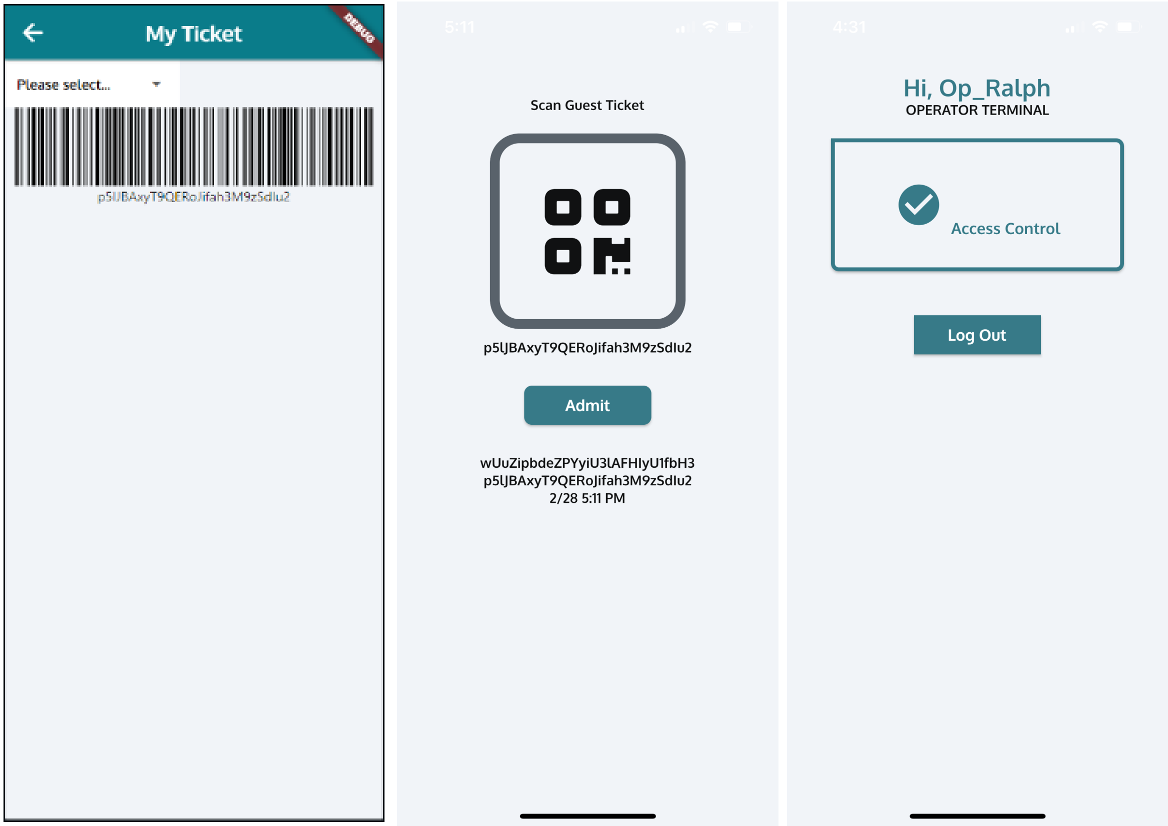

Step 1
Step 2
Step 3
Actions
Guest Purchases Ticket
Operator Scans Ticket at Gate
Ticket Data is Moved
Data Passed
- TicketID
- GuestID
- Date/Time_Purchased
- Day_of_Access
• Ticket Information
checks are done to ensure
- one this is a valid ticket that exists in the database
- this guest ID belongs to the ticket
• (Data Transfered)
- Date and time
- location of admittance
- guest ID of admitted guest
- operator ID of Staff who scanned and admitted ticket
Backend
Ticket Database
Ticket Database
Event Database
a unique redeemable ticket is created in the firestore database under the guests unique ID
the unique redeemable ticket is passed to the database
A transfer of information occurs
The ticket is removed from the active ticket db
The event is stored logging the entry of the guest

Diagram of data flow



My Ticket

Please select...

p5lJBAxyT9QERoJifah3M9zSdlu2

Scan Guest Ticket

p5lJBAxyT9QERoJifah3M9zSdlu2

Admit

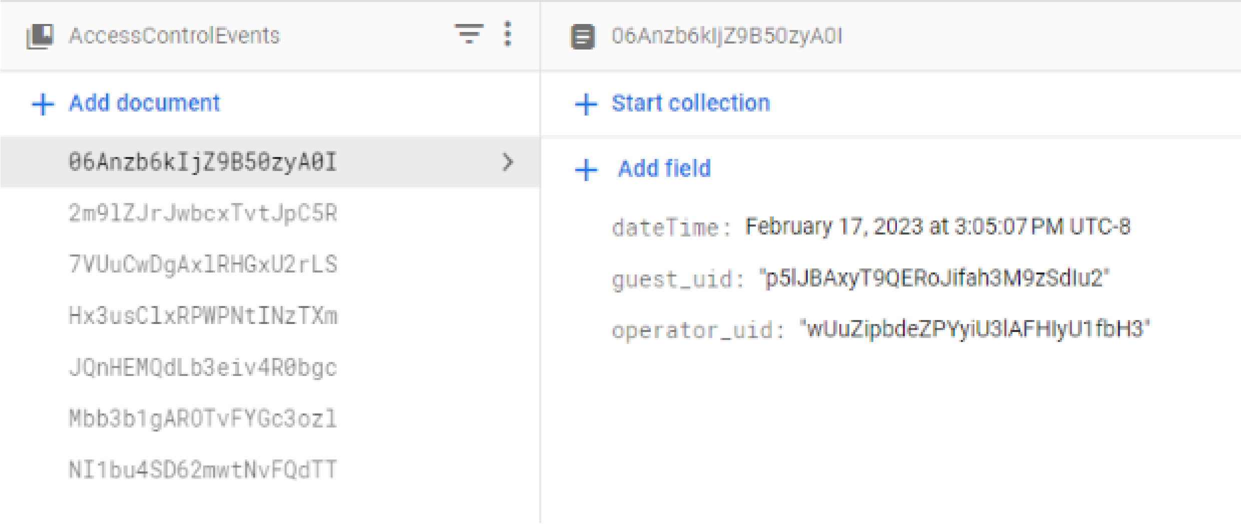
wUuZipbdeZPYyiU3lAFHlyU1fbH3
p5lJBAxyT9QERoJifah3M9zSdlu2
2/28 5:11 PM

Hi, Op_Ralph
OPERATOR TERMINAL

Access Control

Log Out

Operator Application Implemented in FlutterFlow. With guest ticket on left, and operator dashboard and admit function on the right.



AccessControlEvents

+ Add document

06Anzb6kIjZ9B50zyA0I
2m91ZJrJwbcxTvtJpC5R
7VUuCwDgAx1RHGxU2rLS
Hx3usC1xRPWPNtINzTXm
JQnHEMQdLb3eiv4R0bgc
Mbb3b1gAROTvFYGc3ozl
NI1bu4SD62mwtNvFQdTT

06Anzb6kIjZ9B50zyA0I

+ Start collection

+ Add field

dateTime: February 17, 2023 at 3:05:07 PM UTC-8

guest_uid: "p5lJBAxyT9QERoJifah3M9zSdlu2"

operator_uid: "wUuZipbdeZPYyiU3lAFHlyU1fbH3"

Event data captured in Firestore backend after guest admittance.

5.5 Relevant Conclusions

In the case of the project Aquaventure having a backend that is read and writable by multiple mobile clients is advantageous not only for simplicity and ease of use but scalability in the product. Allowing registration to occur at a single point for staff and guests as well as purchasing redeeming and using tickets via the same back end allows for ease of prototyping debugging as well as being able to implement other features on top such as purchasing add-ons by an operator or buy the guest application by using Google's firestore we are able to prototype features such as this easily and quickly which play into our main objective of developing a process to rapidly prototype and test application features.

6 Part C: Developing a deployment model to distribute application changes downstream with continuous delivery.

6.1 Breakdown of Goal

In this section we want to develop a deployment model for the Aquaventure product that distributes application changes downstream with continuous delivery in order to create a high quality product for the customer.

6.1.1 Technical Vocabulary Defined

1. CICD - is a method to frequently deliver apps to customers by introducing automation into the stages of app development.
2. Agile - refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.
3. Github - Git is used to store the source code for a project and track the complete history of all changes to that code.

6.2 Research

Rapid prototyping is an essential process in software development that enables developers to create a working model of an application quickly. In this process, it's crucial to develop a deployment model to distribute application changes downstream with continuous delivery. That way the end product is continuously maintained and developed in the highest quality possible.

6.3 Solution Method

Given the research I have conducted, I am going to implement this into the Aquaventure project in order to create a CICD (Continuous Integration Continuous Delivery) deployment model using FlutterFlow and other tools. Below are some key points from my research about developing a CICD (Continuous Integration Continuous Delivery) pipeline and my implementation of these points in the Aquaventure project at Vantage:

6.4 Implementation

6.4.1 1) Start with a clear problem statement:

Before starting any software development process, you need to have a clear understanding of the problem you are trying to solve. It's essential to define the problem statement, goals, and objectives of the application upfront. This will help you to stay focused and ensure that the application meets the user's needs.

Problem Statement for Aquaventure Project:

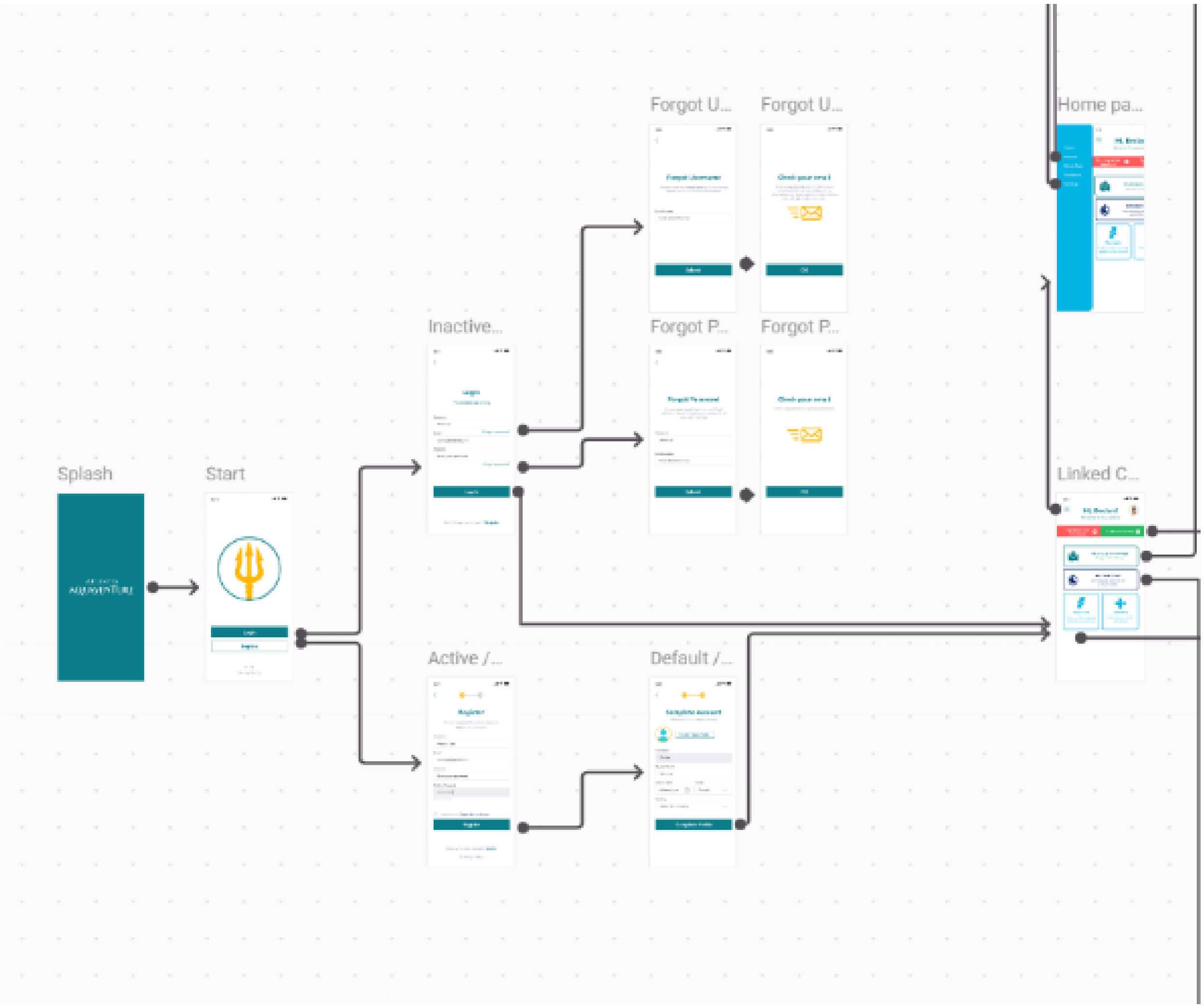
Develop a guest experience mobile application for Aquaventure that provides several functions such as;

- User Registration/Login
- Payment Info Input
- Wristband Registration
- Park Map

- Friend/Party List
- Ride Wait Times

6.4.2 2) Use low-fidelity prototypes:

Low-fidelity prototypes are quick and easy to create, and they help you to explore different design options without spending too much time and effort. You can use tools like pen and paper, whiteboards, or digital tools like Zeplin to create low-fidelity prototypes.



Aquaventure Screen Board made with Zeplin

6.4.3 3) Develop the deployment model upfront:

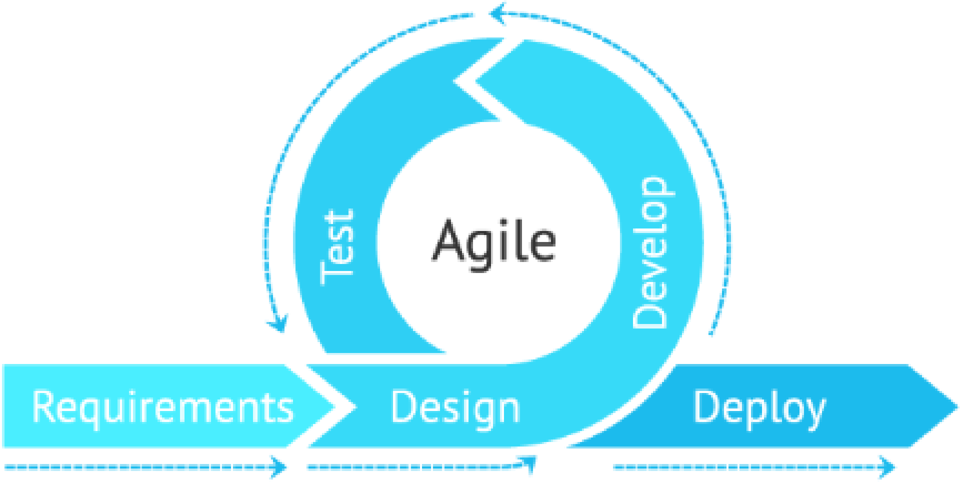
To distribute application changes downstream with continuous delivery, you need to develop the deployment model upfront. This involves setting up the infrastructure, defining the deployment pipeline, and automating the deployment process.

I would define the infrastructure of a mobile application in 4 major criteria;

- Frontend (Buttons, Widgets, interactables)
- Backend (Where the data is stored, communicates with the frontend to create dynamic content)
- Testing (TestFlight, virtual environments, ect)
- Production (IOS AppStore, Google Play, Web)

6.4.4 4) Use agile development methodologies:

Agile development methodologies like Scrum or Kanban are ideal for rapid prototyping as they emphasize iterative development and continuous delivery. This allows you to get feedback early on and make changes as needed.



6.4.5 5) Test early and often:

Testing early and often can help you to identify potential issues early on and address them before they become bigger problems. This can also help you to ensure that the application meets the user's needs and works as expected.

- In the Aquaventure Project, using a tool like FlutterFlow integrates the principle of testing early and often into its design. When developing an application on FF, the editor informs you of issues and errors in the code in real time, and will not let you test the application until major errors are taken care of.
- Another feature of the Flutter framework is that it allows live compiling of the code allowing for a live testing feature in FlutterFlow where you can make changes to the application and instantly see these changes in its testing environment allowing for rapid change and testing.

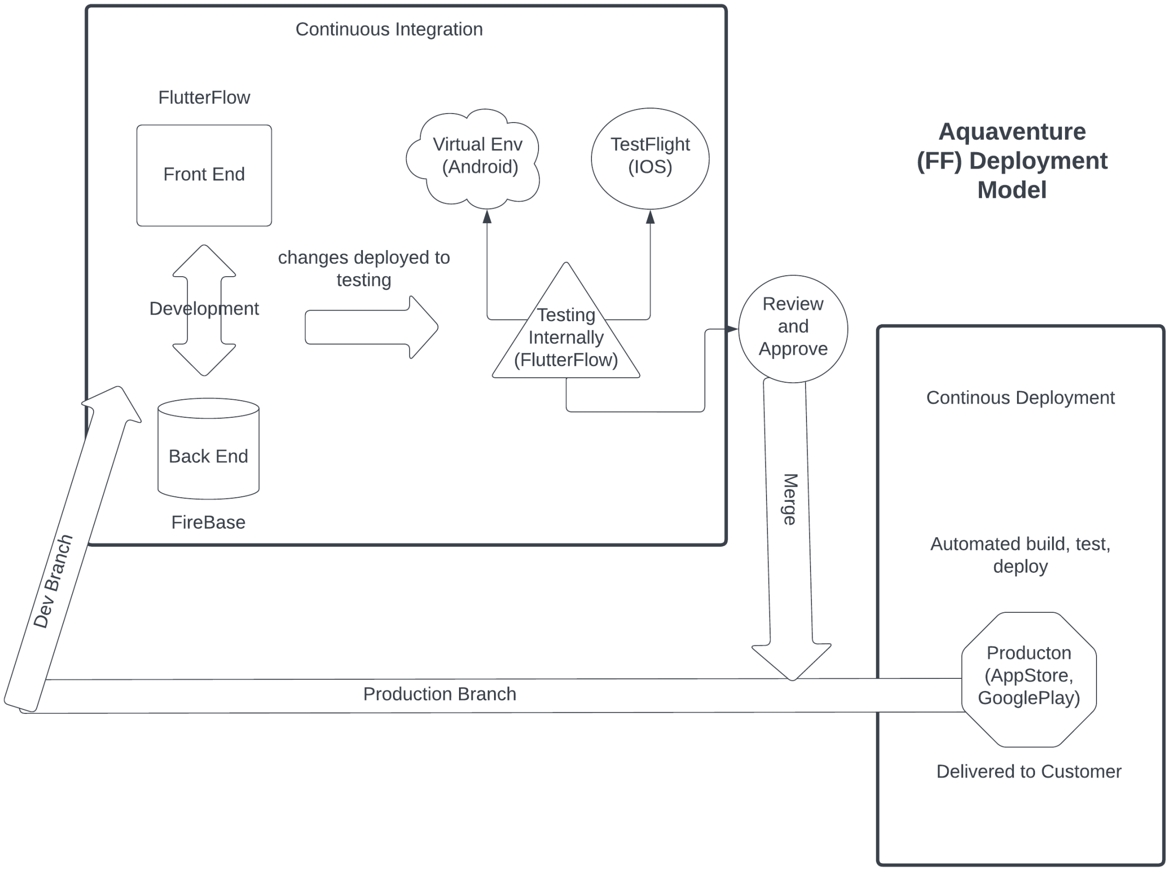
6.4.6 6) Use version control:

Using version control tools like Git can help you to manage changes to the application code and collaborate with other developers. This can also help you to roll back changes if needed and ensure that everyone is working on the same version of the code.

- FlutterFlow has built in integration with Git as well as its own automatic version control features.

6.4.7 7) Continuously monitor and improve:

Continuously monitoring the application can help you to identify potential issues and make improvements as needed. This can also help you to ensure that the application is running smoothly and meeting the user's needs.



Given the 7 best practice criteria above, I have developed a visual CICD deployment model for the Aquaventure project.

6.5 Relevant Conclusions

In conclusion, rapid prototyping is a crucial process in software development, and it's essential to develop a deployment model to distribute application changes downstream with continuous delivery. By following these best practices, you can create a working prototype of the application quickly and ensure that it meets the customers needs.

7 Integration

7.1 How does this process create a high quality product at Vantage

Our goal in this study was to:

Develop and demonstrate the best practice for rapidly prototyping applications in a way that;

(A) Accelerates delivery, reduces complexity, and lowers cost.

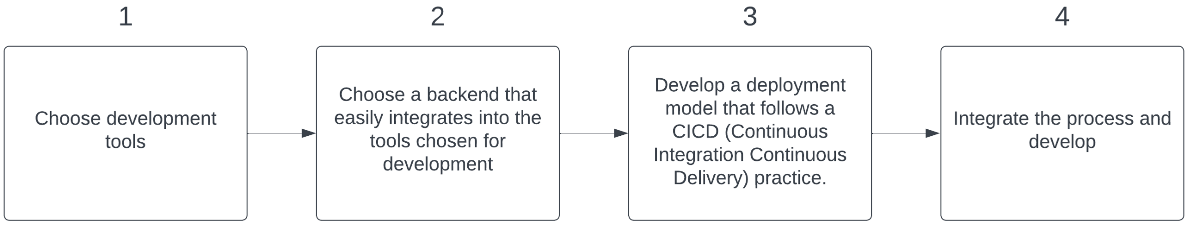
(B) Ties together multiple clients with the same backend, understands and fulfills customer requirements and architecture.

(C) Develops a deployment model to distribute application changes downstream with continuous delivery.

In order to create a high-quality Guest Experience product at Vantage for the customer Atlantis Dubai.

This process created is demonstrated by the diagram and steps listed below.

7.2 Diagram of Prototype Process


1
Choose development tools
2
Choose a backend that easily integrates into the tools chosen for development
3
Develop a deployment model that follows a CICD (Continuous Integration Continuous Delivery) practice.
4
Integrate the process and develop

1. Choose development tools that; accelerate delivery, reduce complexity, and lower cost. In our study we chose to a low-code tool, FlutterFlow.
2. Choose a backend that easily integrates into the tools chosen for development as well as meets customer requirement. In our study we used Google Firestore which seamlessly integrates into Flutteflow and also meets the customers requirements.
3. Develop a deployment model that follows a CICD (Continuous Integration Continuous Delivery) practice. In our study we developed a CICD deployment model using the 7 best practices in Part C.
4. Integrate the process and develop software.

7.3 Final Conclusions

Rapid prototyping is a crucial process in software development that enables teams to test and validate their ideas quickly, reduce time-to-market, and improve overall product quality. By leveraging software application tools, such as low-code tools like FlutterFlow, businesses can accelerate delivery, reduce complexity, and lower costs. Rapid prototyping also enables scalability and simplicity in the development process, allowing for the implementation of new features quickly and efficiently. Additionally, a deployment model for continuous delivery downstream ensures that the prototype meets the user's needs. Following best practices and adopting rapid prototyping tools as a key practice is imperative for software development teams to deliver high-quality software applications that meet customer needs and expectations in today's fast-paced and dynamic market.