Fall 12-7-2022

OADR (Optimized ATM Dispatch Route)

Brendon Hongpong
Sheridan College, hongpong@sheridancollege.ca

Neil Sano
Sheridan College, sanone@sheridancollege.ca

Mandeep Jaura
Sheridan College, jaurama@sheridancollege.ca

OADR (OPTIMIZED ATM DISPATCH ROUTE)

ATM MANAGEMENT SYSTEM | CAPSTONE PROJECT

STUDENT TEAM

Brendon Hongpong,
4th Year Student
E: hongpong@sheridancollege.ca

Neil Sano, 4th Year Student
E: sanone@sheridancollege.ca

Mandeep Jaura, 4th Year Student
E: jaurama@sheridancollege.ca

SUPERVISOR

Prof. Gurdeep Gill
E: gurdeep.gill@sheridancollege.ca
T: <office phone # and 2675>
Sheridan College

DOMAIN EXPERT

Pao Wong,
E: Paowong@wongatm.com
T: 909-438-3055
Community / Industry Partner

ABSTRACT

Third party ATM Companies that do not belong to any central banks often run into issues with scaling as they buy terminals from multiple different companies which they then need to manage themselves. Whether through manual calculations and understanding of a location's traffic and clientele, to providing an optimized route to ensure they save as much time and money on gas expenses and overtime. There is also a need to verify that the correct amount of money was loaded into the ATMs, if an employee has already visited that terminal, and a way for an employee to understand ATM error codes. However, each of these ATM manufacturers have their own sites that the owner has to login to which can create a lot of wasted time and confusion as they navigate through the many different websites and applications. OADR plans to consolidate the features of applications that are currently in use and provide a unified coherent experience in a web app that these companies can use. With a mobile application for field workers that allows them to verify their reloads, and repairs, and gives them access to a database to access the many error codes that an ATM Terminal can have.

ABOUT CAPSTONE PROJECTS

TIMELINES • PROGRAM • SCHOOL

- **September 2021 – December 2021:** Capstone Project Inception, 4-credit course (18 hours / week)
- **September 2022 – December 2022**: Capstone Project, 4-credit course (18 hours / week)

PROGRAM • SCHOOL

- Hons. Bachelor of Appl. Computer Science (Mobile Computing)
- Applied Computing, Faculty of Applied Science and Technology

Sheridan | Get Creative

Table of Contents

INTRODUCTION

In many countries, customers prefer cash as their choice in payments due to convenience and no service charges from the large financial service companies such as VISA or Mastercard. With the increasing advancement of technologies in the financial sector, customers are seeking faster, more reliable, secure, and convenient ways of accessing cash

The purpose of this document is to describe our project, our plans, our solution and how it will impact its industry.

Revisions

- 1.0 – Describes our Project, the Problem and our Solution including how it will impact the industry and its feasibility as a project.
- 1.1 – Editing work has been done on the PID to fix grammar issues and clarity.
- 1.2 – Creating the Table for the distribution of work and Project Management and filling out the project plan section
- 1.3 – Added information about the system context and use cases
- 1.4 – Added Testing and Validation sections to the PID
- 1.5 - Updated Software Model screenshots based on feedback from Defense
- 1.6 - Updated Project Plan to use Asana as the project Management
- 1.7 - Updated document to reflect the state of the Alpha Release
- 1.8 - Updated document to reflect the state of the Beta Release
- 1.9 - Updated document to reflect the state after the defence
- 2.0 - Updated minor changes suggested by the committee
- 2.1 - Updated document to reflect the state of the Final Release

PROJECT OVERVIEW

The project we are working on is OADR which is software that will be designed to reduce the effort a company needs to manage each ATM terminal they have purchased. We are a team of three 4th year Sheridan College students in the Honors Bachelor Mobile Computing Degree program and are experienced in developing applications throughout our 4 years at the college along with an experienced faculty supervisor who has overseen many capstone projects to assist us in completing our goal. We have two industry partners that will be providing insight into their industry and their workflow. The domain and industry that our project will be taking part in are Financial Transactions Processing as Wong ATM keeps track of all the transactions that occur on their terminals and receive a fee from each transaction that occur on the machines. The CEO of Wong ATM, Pao Wong has described his problem and the inconvenience of moving between 4-6 different applications every day to complete his work and that there are very few software solutions to assist the ATM Management Industry.

DOMAIN AND INDUSTRY OVERVIEW

OADR wants to address a problem in the Finance and Insurance (52) \ Credit intermediation and related activities (522) \ Activities related to credit intermediation (5223) \ Financial transactions processing, reserve and clearing

house activities (52232). This industry handles financial transaction processing, reserve, and liquidity services and/or checks other financial instrument clearinghouse services. The Finance and Insurance industry is very large having 195,919 establishments with 40,276 as employers and a GDP increase of 3% as of 2019. In the case of Canada, the Self-service technology market is not as known but for the entire market of North America, the size was valued at USD 28.3 billion in 2019 with an expected growth of 6.7% from 2020 to 2027. ATM's hold the largest revenue share with 51.6% in the self-service technology market in 2019. This means that there is potential for a lot of profit. The main issue however is the lack of software to manage ATMs from different companies. Our industry partners at Wong ATM have described a large problem they have encountered.

PROBLEM DESCRIPTION

Wong ATM is a small company based in Los Angeles that manages around 300+ ATMs and are our partners for this project. The problem that our partners have encountered is that they often open between 5-8 different ATM accounts depending on the company they have ordered the ATM terminals from to manage them. Making it difficult to determine the data they are receiving from their ATMs. An example of this data is which locations require a refill of the cash they contain. Is one of their ATMs not working or inactive and how many transactions were performed yesterday from that terminal? Our Industry partner Pao Wong at the time of the creation of this document is the only one that has knowledge of the traffic and cash flow of the locations, and he approves all the routes to reload the ATMs to his employees. This limits his ability to scale and expand his business to have more employees and ATMs because he finds that he is spending most of his time handling all these tasks with the current ATMs he owns.
Their workflow is also quite hectic as they often need to open 4-5 different applications to create and resolve tickets, receive phone calls from customers, dispatching their field workers, verifying if their field workers arrived at the location, to verifying if the field workers dropped the correct amount of money into their assigned terminals. Changing between 4-6 applications constantly can increase the chance of errors that can occur between the applications and cause the employees and our partner to lose track of what they were initially doing in the first place. Being able to consolidate all these applications into a singular application will ease the number of tasks that fall on our partner allowing more time to focus on the expansion of his business.
Their Field workers also use an app that provides them with the optimized routes they are assigned. However, if they reach a terminal that displays an error code, they often do not know what it means. This requires them to then call Wong ATM's office for them to retrieve that information, wasting a lot of time for both the field workers and the dispatchers.

SOLUTION DESCRIPTION

We will be using machine learning and data analytics, which will help us forecast and predict the most optimized route for the ATM re-filling process. Given the limited amount of money that is available, we need to find a way to divide the money to generate the most profit for our client. There are many factors to be considered; for example, the ATM in location B might be busy on Thursday which will give our client a profit of $2000 but it might generate only $1000 profit. If the weather on that day is rainy or snowy the profit of this location might be cut in half. The main question to this problem is "Would it be better for the available worker to re-fill location A rather than location B or C?" because other locations have different circumstances as well. Assuming that location B is the best choice to

be re-filled; The surrounding ATM of location B might also be a desirable choice to be re-filled as well considering the transportation cost and the salary that we must pay to the worker if they work overtime. "What if the ATM still has cash inside and does not need to be refilled today?" Would it be beneficial for the client to fill that ATM today? What if we need to come back and fill it tomorrow? It is difficult to keep track of all the different circumstances manually to know which route would be the best for the field workers to follow and distribute the money for the ATMs.

Mobile Computing

We will develop an Android application for the field workers so that they can more efficiently manage their daily tasks assigned to them. They will then be able to report to the office more easily of what they have done for the day. The application will include additional functionality for them such as the ability to upload and verify their gas bills to the main office and the ability to see the error code instructions from an error code they are shown at an ATM machine when the machine is broken.

Cloud Computing

We would use the API that is given by the ATM processor service and store transaction data directly in our database. The data will then be used to form suggestions through a machine learning process and then be used for Inventory management to determine the location of the ATM and the amount to be filled that would generate the most profit for the company (Potentially determine which ATM needs to be fixed, changed, or replaced from the data) and also be able to generate a financial report but we have to make sure that all the data is secured and encrypted in our database.

Advanced Areas of Computer Science

The advanced areas of computer science that we are going to implement in our system are machine learning and data analytics. There are multiple factors that we will have to consider to make the algorithm predict an optimized amount of money to be distributed among the ATM machines to generate the most profit. The example of the factors that we will have to consider are which days of the week, the amount that got withdrawn from the previous day, the locations that are close to the ATM that are being filled, is the area high-risk, the previous amount loaded and the cash that the company has for that day. The previous amount loaded is the key factor since the data is directly from the client which will help our algorithm to make predictions more accurate and more sensible to our client.

SOLUTION IMPACT

The development of this solution will have a significant impact on client productivity. Starting with saving time, the client is currently using more than three applications to manage their business. This solution will provide a central system to operate his business. This results in opportunities to focus on their business expansion, which requires more attention. Currently, the old system is giving incorrect suggestions to load their ATMs which do not make sense to the client. With the help of machine learning, the client will receive proper suggestions on the amount of cash to reload the ATM based on the traffic for the location. This will increase the efficiency of the client as the algorithm will suggest money to machines that are more heavily used. This should then reduce the time it takes for the company to

distribute the money for all the machines they own. This efficiency increase should then allow Wong ATM to invest more heavily in expanding its business.

The client is also manually notifying the dispatched field workers of ATMs that need a repair through a separate application so they can visit these locations and check the ATM. We are adding a ticketing system for client convenience. The client can generate the tickets for ATMs that are not working or having issues and this ticketing system will then notify the driver about the issue. It will also provide a ticket status so that client can see the ticket status whether it is resolved or in progress. Having a ticketing system that our customer's clients can easily access will reduce the need for our client to set up tickets themselves and save more time for the client.

Having an optimized route for ATM locations will save extra milage used by the driver to reach specific locations. It will not only give a route but also give extra information such as other ATM locations within the same route that also need to be loaded within the allotted time. Drivers will get notifications and a list of ATM locations which require them to visit within the mobile application that is connected to the cloud database to provide them with the required information they need.

Currently the field workers are using a mixture of Microsoft Teams and discord to confirm location status, gas bills and if they have reloaded the ATMs with the correct amount of money. This solution will benefit the driver by providing them with a way to claim their gas bills and confirm ATM status through the mobile application.

If there is any problem at an ATM, it will only display an Error code to field workers which makes no sense to them, so they call the company's head office to ask for a solution. With the mobile application, they can search the cloud database containing the error codes and find a video within the application to fix the issue.

SOLUTION FEASIBILITY

The feasibility of this project is doable but there are some risks that we need to consider. The main risk we believe will be security. As we have seen the data from our Industry partner the data must be protected as it contains the general location of the machines and how much money will be reloaded into the machine. If this data is revealed or breached it can put field workers at risk during reloading or recounting of the ATM terminals that they have deployed. This risk can be mitigated with 2-factor authentication, assigning roles to only allow specific people access to this information and encrypting the data to make it harder to get sensible data in case a data breach occurs. In terms of deployment, it is still a work in progress, but our client has told us that they are thinking of commercializing the software if we can successfully show them a viable product as of right now, they wish for us to create a program for them to use.

Design and Construction

Designing this solution is feasible but has clear security implications. As this solution will have all information from banks, all ATM data such as location, deposit amount, and money transactions. If the database gets hacked, it can lead to serious financial disaster for the client. We believe that having a proper crypto algorithm will be helpful to solve this problem but will need multifactor security implementation to protect the information.

Deployment

The solution will be implemented in two stages, as a web application and android mobile application. All the expenses will be covered by the client.

Adoption

As a developer we will be getting all our data from the client. He already has all permissions and authorizations for APIs. Therefore, it would not cost the client to share API with us. We will use Amazon cloud services to implement this solution. Servers will be extended based on requirements. While developing the solution, we could use this service at no cost. After completion of the project, amazon will start charging for services. As one client, it would not be costly for the user. However, we will keep track of the data, and will consult with the client about its requirements.

PROJECT REQUIREMENTS

This section contains the Requirements for OADR, the system context, the use cases of the project, and the prototypes we have produced. The system context will involve our stakeholders and our top-level use cases such as assigning routes to field workers and the field workers receiving said routes when dispatched. After this section, we will go into the architecture of the system to show the functionality of how the core classes will function and interact with each other.

Link to the VPository: https://online.visual-paradigm.com/w/ybkhppis/

SYSTEM CONTEXT

The system context revolves around the CEO, Accounting, and Fieldworkers of the business, they will be the primary users of the software. The Accounting will be the person that enters in the available cash for the day to ensure the CEO does not spend more cash than he is able to and keep track of the available cash flow through viewing the history of the input cash. The CEO can view all his terminals, and assign jobs to be completed on the terminals, this information will then be taken and passed through an algorithm that will assign routes to field workers that are available for the day. The Fieldworkers will then receive these routes and be dispatched out to the field, where they will be able to use the mobile application to login and view their assigned terminals and the jobs they must do for each terminal.

USE-CASES

WongATM Management System

The use-case shown below focuses on the CEO's side of the system which will primarily be in the web application. They will log in and be greeted with the ATM Management page allowing them to perform CRUD operations on the database for ATMs, Tickets, transactions, and then create a route that assigns tasks to the field workers.
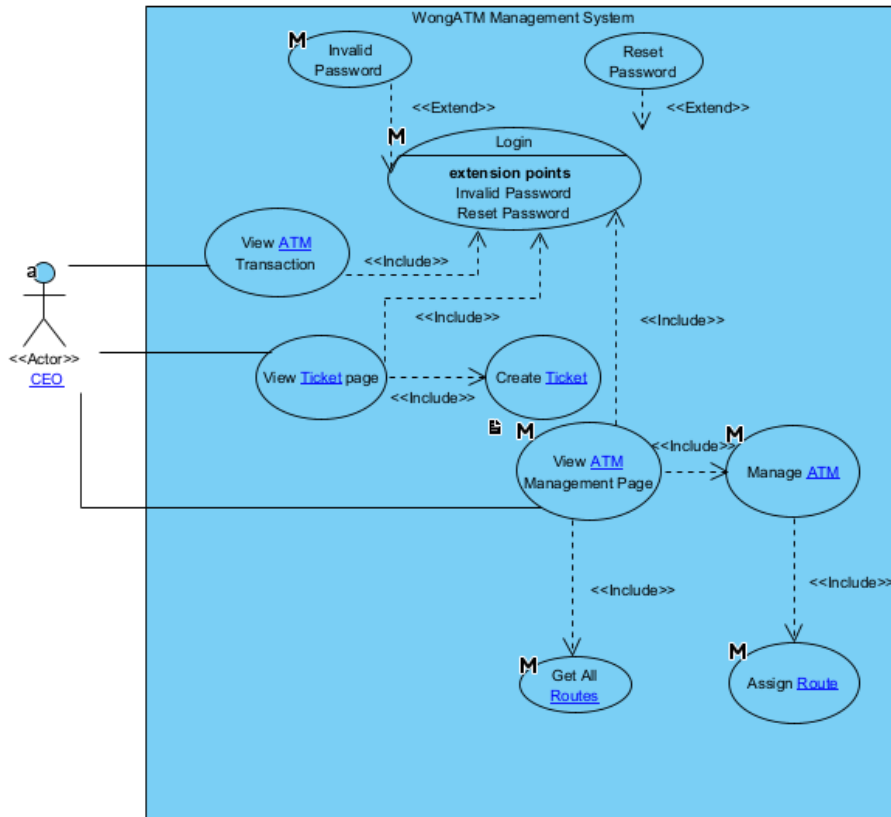

Fig0.1. CEO Use case diagram.

This use case diagram primarily follows the accounting employees where they oversee updating the available cash the business can use to fill the ATMs every day. Like the CEO, they will also login and be limited to two specific functions uploading the available cash for the day and viewing the available cash history to keep track of the day-to-day cash flow. This allows the business to check that they are not deploying more cash than they can afford. This also allows the CEO to only spend the amount of cash that the accounting staff have set up to use.
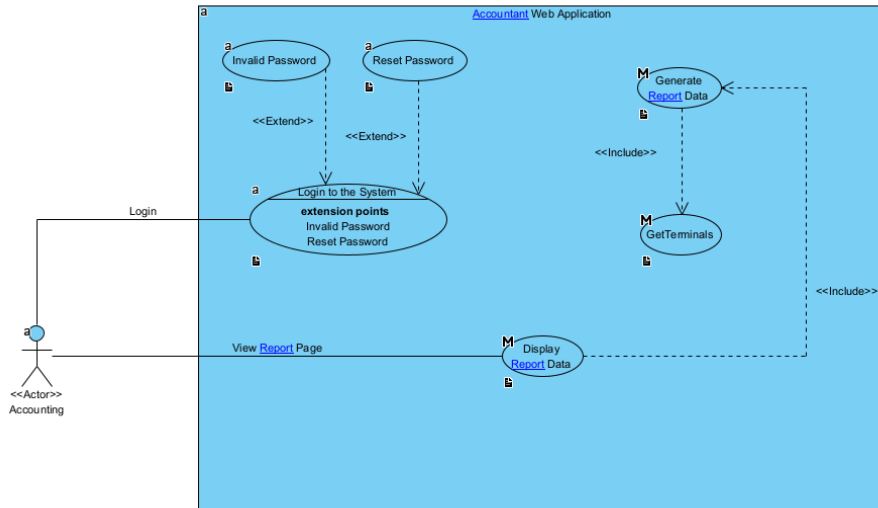
Fig 0.2 Accounting

This diagram shows the operation that the accountant can do which is a simple process of just viewing the Available cash history or uploading the available cash so the CEO can use it to deploy his fieldworkers with their specified load of cash to be delivered to the ATMs that require to be reloaded.
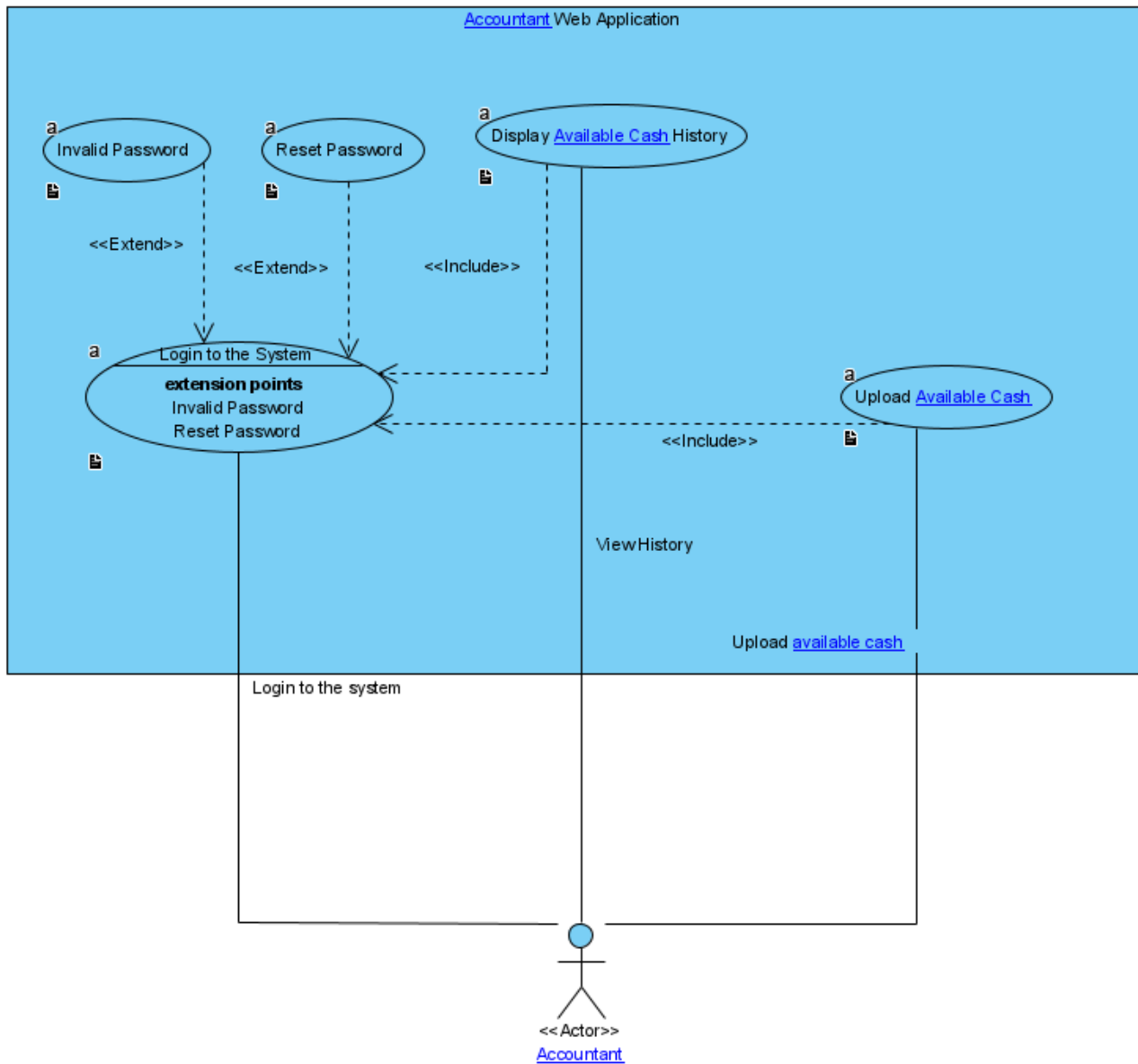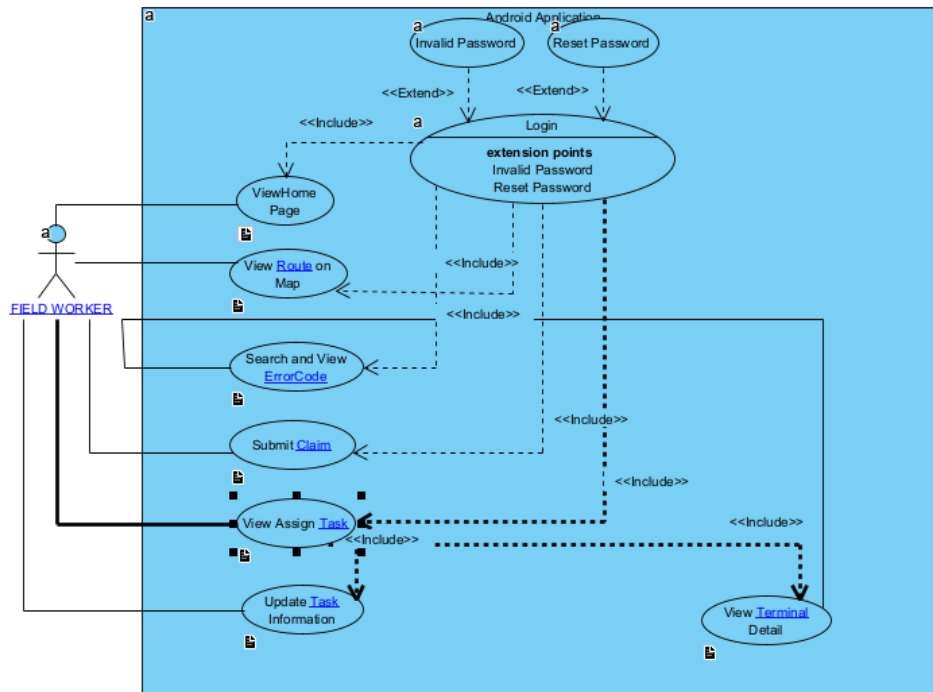
Fig 0.3 Accountant

Fieldworker user case Diagram
Fieldworker: Fieldworker will use android application to see their task, submit claim, and search for error code. The diagram below shows simple tasks that fieldworkers can do using android applications.

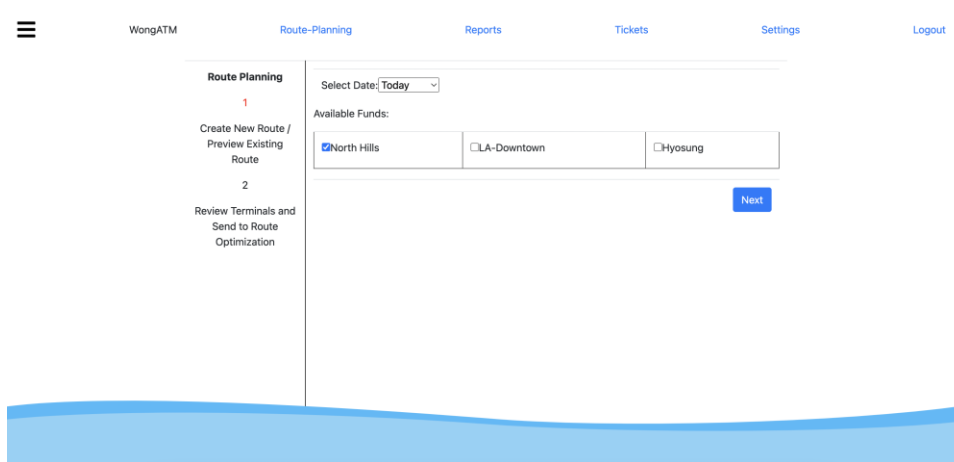# USER INTERFACE

Web Application UI
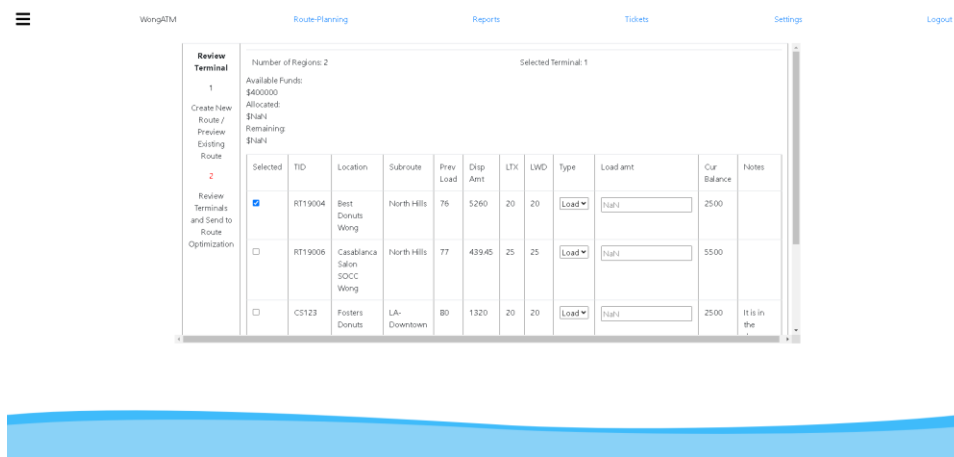

Fig1. Login Page.

Fig 2 Route Planning Page.
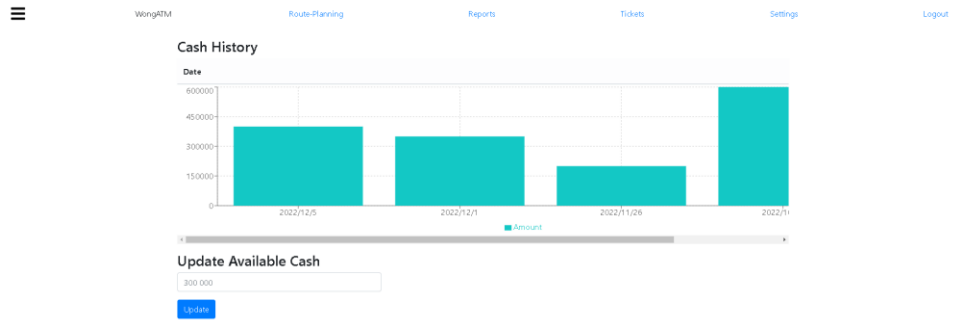

Fig 3. Route Optimization page.

Fig 4. Accounting Report.

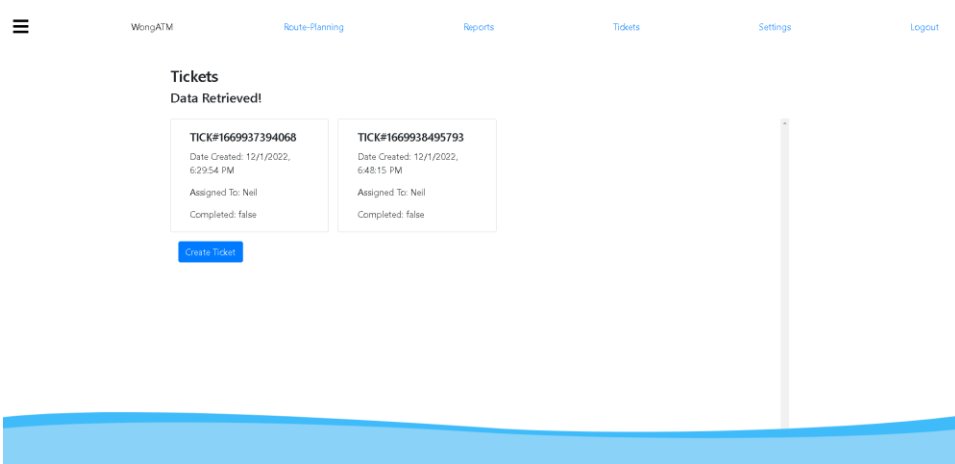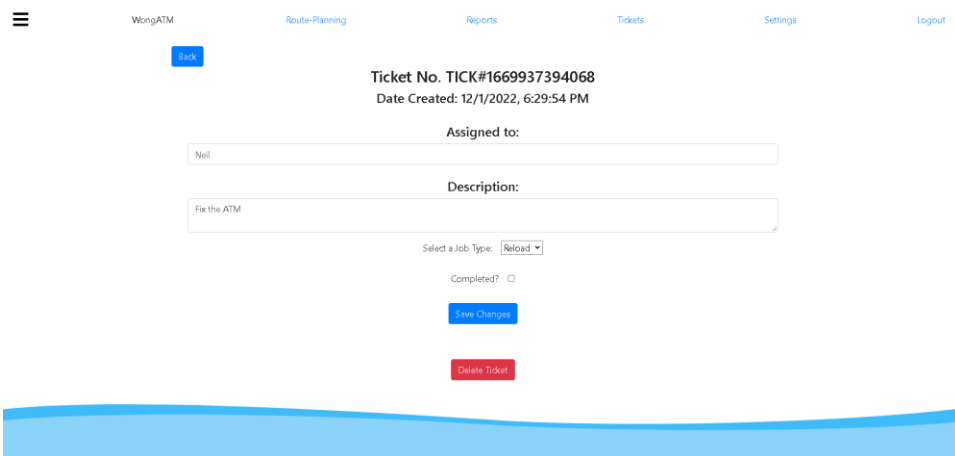

Fig 5. Tickets management page.

Fig 6. Ticket status page.

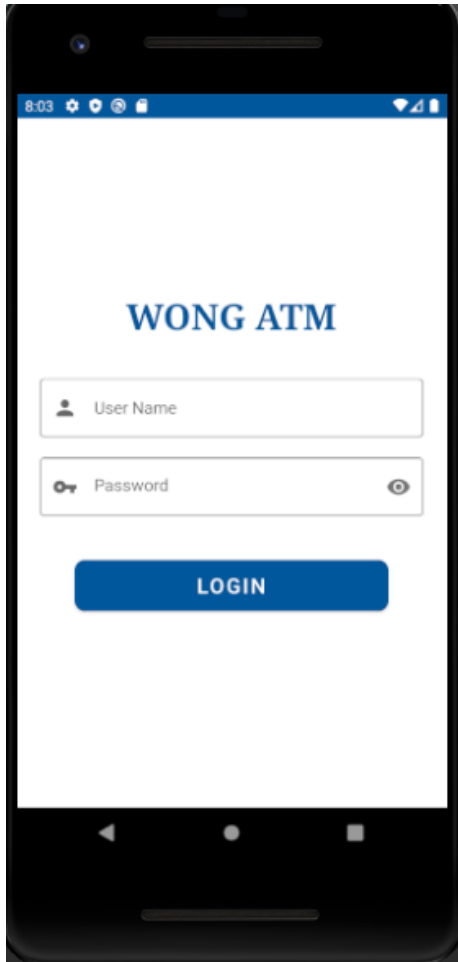Fig 7. Ticket Creating page.

Android Application UI
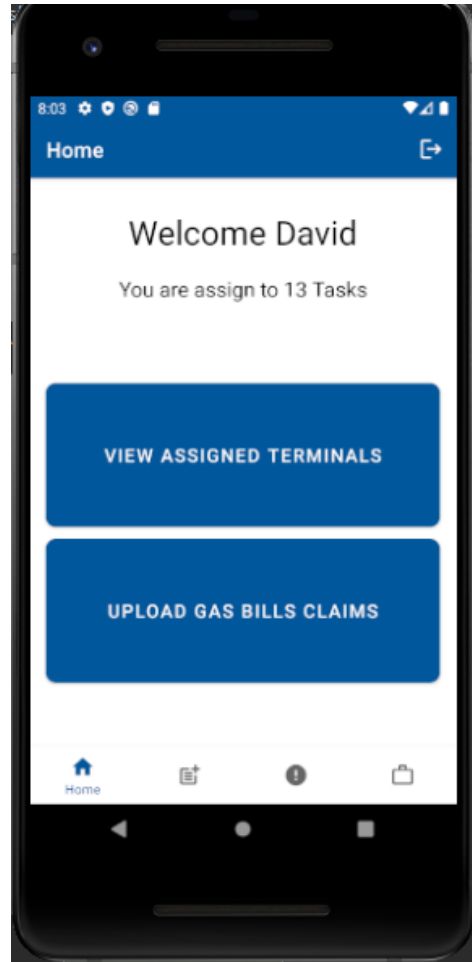
Fig 8. Login Page
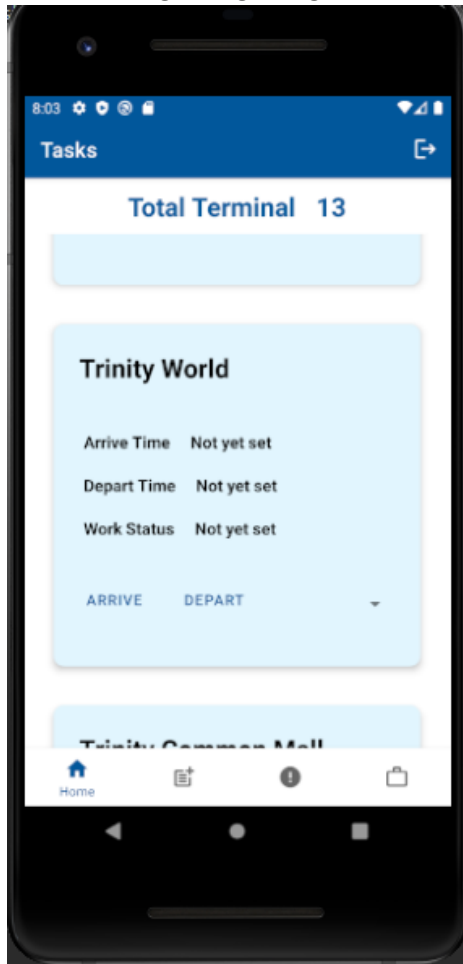
Fig 9. Home Page

Fig 10. Task Page unfilled Details
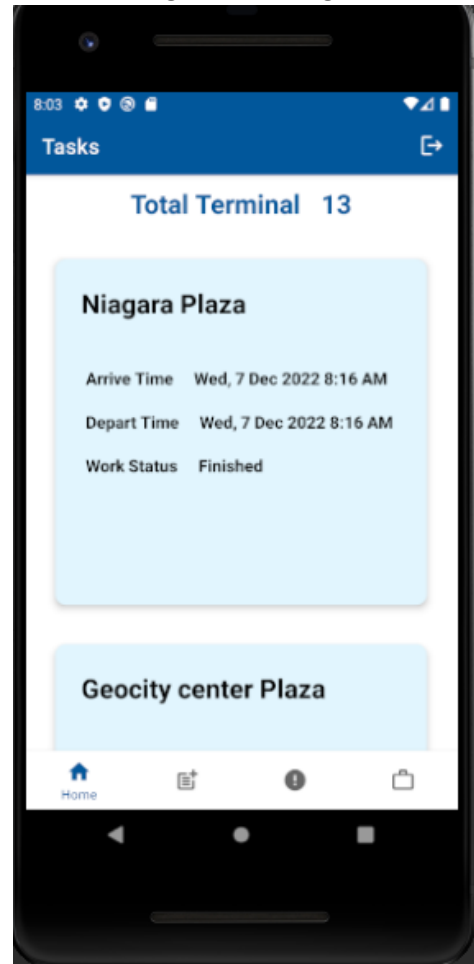
Fig 11. Task List Page filled with Details
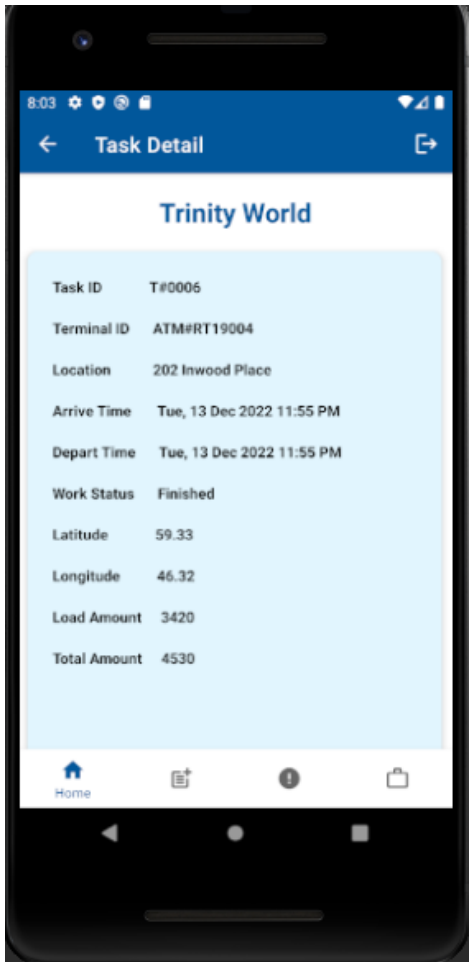
Fig 12. Task Detail
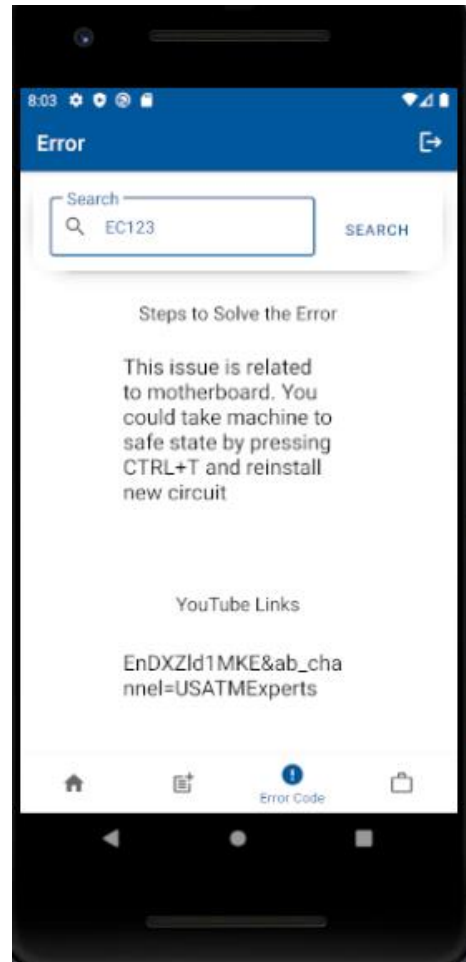
Fig 13. Error Page
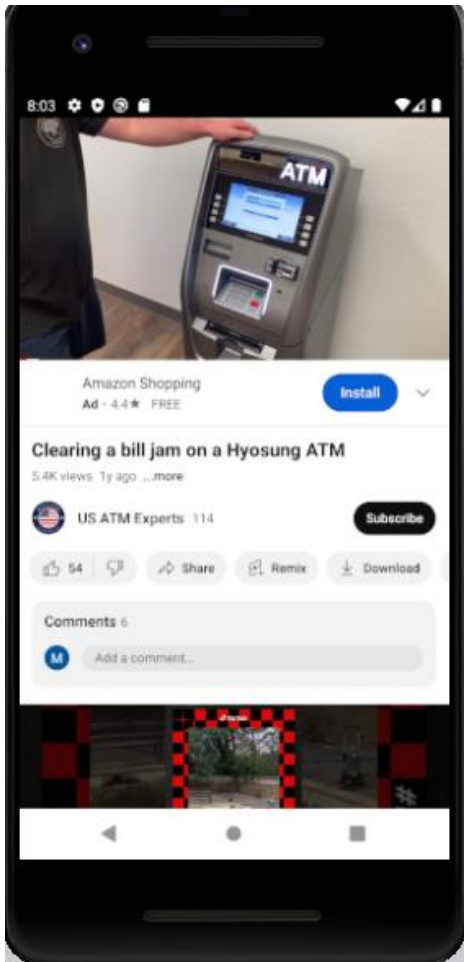
Fig 14. Youtube page from Error Page
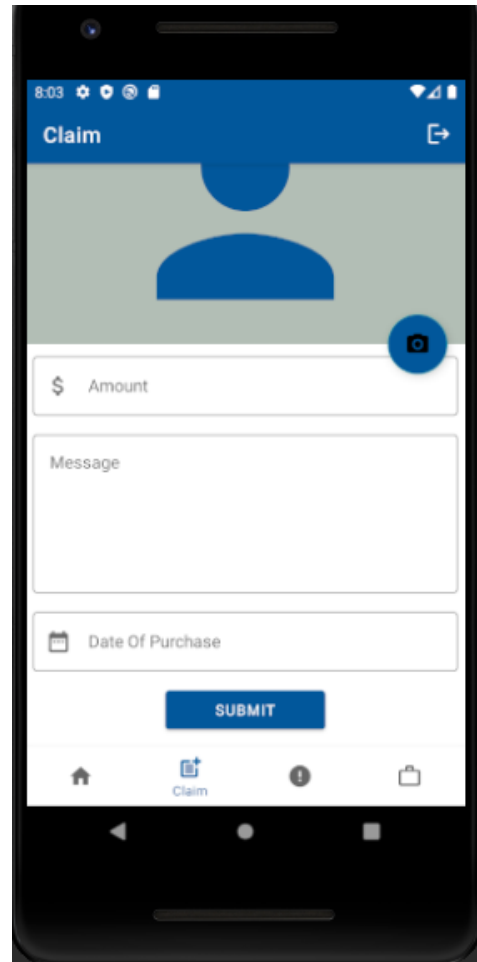
Fig 15. Gas Claim Page

PROJECT ARCHITECTURE

This is the project architecture section that contains OADR's architecture overview, the components of our system and a deployment model of how the project will be deployed. The architecture overview contains a brief description of our software system from the architecture that we use to the patterns that we will be using to guide our software development process. The component section will briefly describe the parts of our software system, what the parts will contain and how they interact with our system. Finally, we have the deployment model which shows an idea of how the project will be deployed, the libraries, frameworks, and functions that the client devices will depend on. Since we are using AWS, we will be also using Amazon Cognito for our user database to implement our logins functionality. Amazon Cognito also enables us to authenticate users through an external identity provider and provides temporary security credentials to access your app's back-end resources in AWS or any service behind Amazon API Gateway.

ARCHITECTURE OVERVIEW

Our back end will be hosted in Amazon's AWS cloud service and be using much of the built-in functionality to build our application. It will primarily be a serverless architecture, with the backend being composed of an API through AWS's API Gateway service, with the requests individually handled by lambda functions which will call our NoSQL DynamoDB to create, retrieve, update, and delete instances on the database depending on what the user decides to do through our front-end applications. Our front end will be created through an Android application for the field workers and a web app for the accountants and CEO that uses React.js. The web app will be communicating with the API a lot as it will be creating, reading, updating, and deleting information from the database as the CEO manages most of the business through a web application in his current workflow. For an architectural pattern, our project falls under a microservice architecture that has one main API that then communicates with an authentication service that will only allow its users to access specific parts of the application that they require to do their job.

The android application will also have a backend hosted on Amazon's AWS cloud and will consume API hosted on AWS's Gateway. The Android application will be used by Fieldwork, and It is developed using Kotlin in Android studio. Most of the time field workers will perform reading operations such as viewing terminals and task details, but they will also use the application to submit their gas bills.
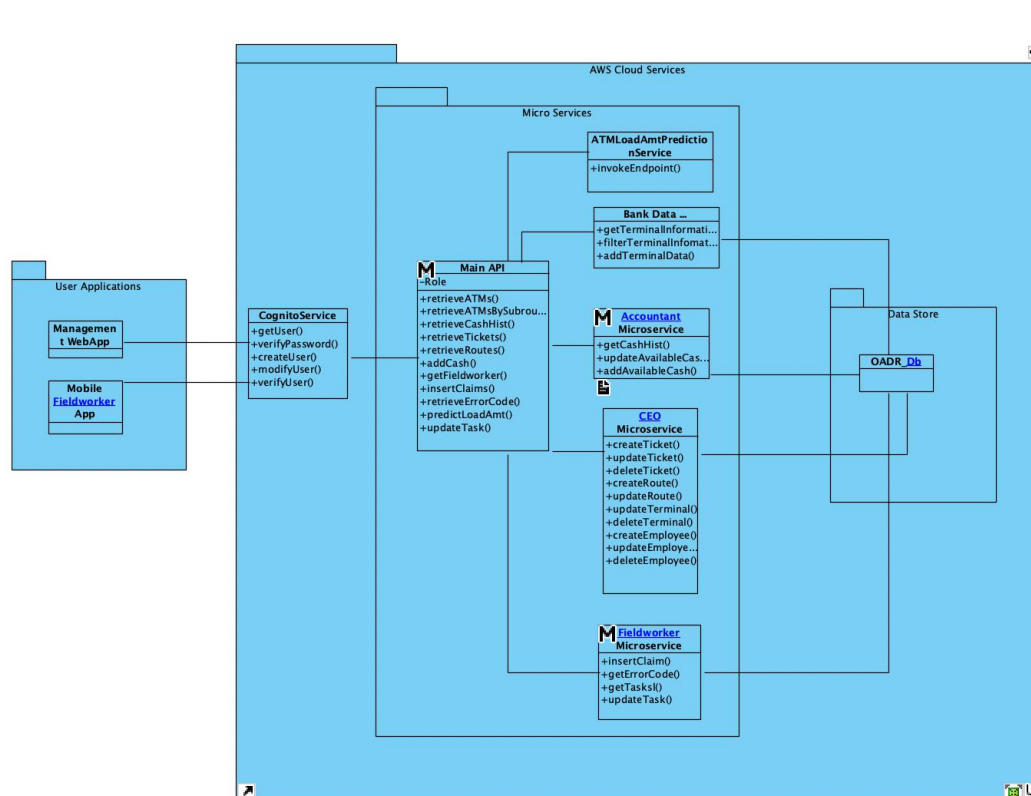
(Software Architecture)

SYSTEM COMPONENTS

The System is composed of the clients which will primarily be the CEO's computer, the accountant's computer, and the fieldworker's phone. The CEO and accountant will primarily interact with the system through the web app that we have created and send Get, Post, Put and Delete commands which will then invoke a specific lambda function in our API Gateway. These functions then send the appropriate requests and update our Dynamo NoSQL database. The Fieldworker application will have access to only requesting read, and update operations from the database for looking up errors, receiving tasks and updating their arrival times and departure times from a specified terminal. This system will manage their access to only allow the users to access the information their job requires.

DEPLOYMENT MODEL

The deployment model is primarily based around the accounting computers and the CEO's laptop which will use their web browsers that connect to the internet to communicate with the Main API that we are developing. The Main API depends on the API Gateway that contains all our Lambda functions that manipulate and interact with the DynamoDB. The web browsers will have to have JavaScript enabled to see interact and use our application. The React Library is what will handle rendering our front-end components into a usable User interface for them. The React Router library will give the user the ability to navigate between our Page Components and the Axios Library will be what allows us to communicate with our API Gateway easily and quickly.
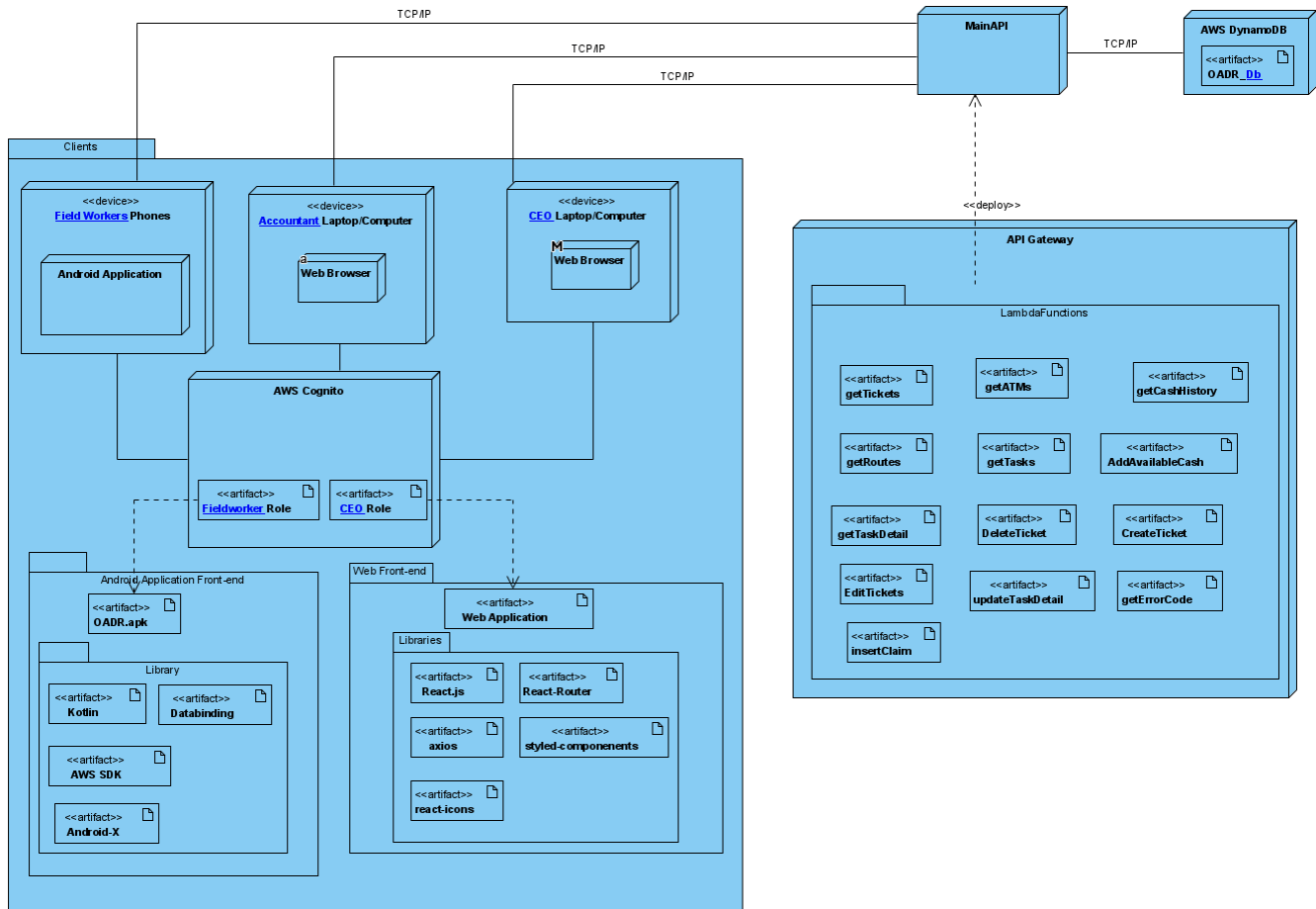
PROJECT PLAN

This section contains our plans for the project and the iterations that it will go through, along with the possible risks that we can encounter. Our project is managed through Asana with an iteration plan that includes 5 iterations from inception to the final release managed in, a risk management plan that will contain risks that we identify throughout our development process with strategies to mitigate them, and a test plan that will describe our testing process for the software to ensure the product is reliable and usable for our client. It also includes a table that will distribute our work to all 3 developers that will be working on this project. These assignments will be updated as we progress through development.

Asana Site PMP Link : https://app.asana.com/read-only/WongATM/8784741888003/296b575d04caebc6f1711c8255237e7b/list

ITERATION PLAN

Our iteration plan contains 3 iterations that are a work in progress, but we are planning. The Alpha release will contain the core functionality of the application on both mobile and web applications and will begin to involve testing

to ensure that no damning bugs are created, and that the product is functioning properly for our partners. In the Alpha stage, we will also begin researching Machine learning techniques and Data analytic tools that we can use to further streamline his workflow. The Beta release will then have the advanced Computer science areas implemented into the product and further analysis and validation testing will be required to ensure that the Machine learning algorithms and Data analytics are performing correctly and not providing inaccurate results. The Final Release will be where we will test the software and attempt to break it in multiple ways to ensure that the product will not encounter any debilitating bugs or issues in the field with our partners.

Asana Iteration Plan Link: https://app.asana.com/read-only/WongATM/8784741888003/296b575d04caebc6f1711c8255237e7b/list

RISK MANAGEMENT PLAN

Security is our main concern regarding risks because our client's business is involved with moving money from one location to another. The location of the ATM that needs to be filled is generated from our application and then sends the daily tasks to the field Worker through a mobile application, hence it is required that our service is always up so that the business is always able to function. This requires the server to be reliable, which is our main concern. To alleviate this issue, we plan to have a backup server as a redundancy because the risk of availability is important to our client business. Without this redundancy and if the server goes down the business stops slowly and loses revenue for the day. These are the top 5 risks that we have figured out so far that can affect our project and product.

Asana RMP Link: https://app.asana.com/read-only/RMP/8784741888003/8dafafe4d36cbff62ccc2161609e0fc1/list

Top 5 Risks

1. Server is down or become unavailable
Description: If the cloud database goes down, it can halt the entire product from running. This will completely halt the workflow of the business resulting in lost revenue.
Mitigation Strategy: Make sure we secure our data in database by Encryption the data and make sure our database is setup properly to prevent hacking.

2. Lack of Communication between project members
Description: Lack of communication between members can lead to work not being assigned to the right person, or a disconnect between what the members are developing for our client between sprints.
Mitigation Strategy: Mention each other in our Work Discord, Teams so we have multiple avenues of communication to ensure that any member if we need specific information can be contacted.

3. Potential data breach.

Description: If a data breach occurs, it can be catastrophic to both the business and product. The product would be branded as unreliable and insecure if the data containing money loaded or ATM location is exposed. Field workers may also be targeted while loading a hi-traffic machine at a location.

Mitigation Strategy: Make sure we secure our data in database through Encryption and only provide the necessary access to parts of the database that our users need to perform their jobs.

4. Data in database becomes corrupted or lost through either drive failure, power loss or natural disaster

Description: If the power disconnects during a transfer of data or upload of data occurs, it can cause data corruption.

Mitigation Strategy: If we have multiple servers in different locations, we could reload the table with the data from another location to ensure the servers have redundancy or if we are given the option to run the server with RAID-5.

5. If the End product is hindering the business's workflow

Description: If the End product is hindering the business's workflow the client will not want to use our product which will result in failing to deliver the usable product to the client.

Mitigation Strategy: We will have to send the test software throughout development to get comparison tests and determine if we are creating unnecessary wait times in his workflow.

PROJECT MANAGEMENT

Project Responsibility	Brendon	Neil	Mandeep
Project Management			
Project Owner	Brendon Hongpong		
Scrum Master			Mandeep
Risk Analyst		Neil Sano	
Requirement Engineering			
Requirements / Business Analyst	Brendon Hongpong		
Stakeholder Champion (by Stakeholder)	Brendon Hongpong (CEO)	Neil Sano (Accounting)	Mandeep (Fieldworker)
Functional Area Champion (by functional area)	Brendon Hongpong (CEO)	Neil Sano (Accounting)	Mandeep (Fieldworker)
User Experience Design Lead	Brendon Hongpong (CEO)	Neil Sano (Accounting)	
Software Architecture			
Software Architect			Mandeep
Requirements Model Lead	Brendon Hongpong		
Domain Model Lead	Brendon Hongpong		
Design Model Lead			Mandeep
Deployment Model Lead			Mandeep
Interaction Model Lead			Mandeep

Construction			
Full-Stack Developer (UI, code and unit tests)	Brendon Hongpong	Neil Sano	Mandeep
Integration / DevOps Lead		Neil Sano	
Testing			
QA Lead		Neil Sano	
Verification & Validation Champion (by Functional Area)	Brendon Hongpong	Neil Sano	Mandeep
Test Model Lead	Brendon Hongpong		
Support			
Tool and Devices Support		Neil Sano	
Communication Support	Brendon Hongpong		

VALIDATION AND TESTING

This section of the document explains our validation and testing strategies we have done so far and what we plan to implement as we continue developing our project in the future.

TESTING STRATEGY

Our validation strategy, as of now, is a back-and-forth process between talking with the client to devise requirements and developing the features. We then display our prototype to them to get their feedback on the increment to further guide us to develop a product that suits their needs. Our testing strategy is still a work in progress but has primarily been a mix between functional testing and integration testing to ensure our many libraries are working together appropriately. In the next few months, we plan on moving to use unit tests and continue developing our project with TDD (Test-Driven-Development) to reduce potential bugs in our program and smooth out our development process.
Link to Asana Test Plan: https://app.asana.com/0/1203217838041614/list

VALIDATION RESULTS

The core functionality of the applications has been completed but unfortunately without the ability to deploy routes as of this time due to a restriction in the software that our client wants us to use in deploying them. In our future work, we discuss in more detail why this has happened. Primarily it comes down to price, current manpower to develop the project, and the need to reduce retraining the employees of the company.

CONCLUSION

The problem that we are solving is the logistics and efficiency of any small company that is managing ATM vending machines (300-500). According to our clients, there is no all-in-one solution that suits their needs, as of right now, they are using 6 different applications to handle them all. The CEO of the company is the only one that knows and understands all the data that the business requires. This ranges from the amount of money that should be deployed to a specific region of Los Angeles to handling the error codes that pop up on their ATM machines. As he is the only one that knows this information he is heavily relied upon and is hitting his limits in what he can handle. He requested that we create a system that integrates all the 6 applications features that he uses into one system and uses a Machine Learning algorithm to automatically distribute the money based on his previous inputs. As the project is still early in development, we are targeting to improve the ease of use of the software and improve his current workflow. In the future, we plan to solve all the key issues that our client has.

PROJECT SUITABILITY

Our solution is still in its preliminary stages of development as mentioned before but, since we are building this project as a microservice architecture it allows us to either add or remove features depending on our client's requests. Our client uses multiple web applications for distributing money and scheduling routes for his employees. The employees on the other hand use a mobile application that tracks them to ensure they are doing their assigned jobs. Our project involves both a web and mobile application as well. We plan to have similar features but, include more useful tools such as an error code lookup integrated into the mobile app. As for our web application, we are simplifying their current solution and removing what our client often finds unnecessary in their current workflow. Our client is also overseeing our progress and is constantly giving us feedback on our design and ideas. This reaffirms that we are on the right path as we have received no major complaints about how our project is progressing.

DOMAIN EXPERT EVALUATION

Overall, we have received positive feedback from our clients that the feel of our application suits their work but, requires some change in the UI for the reports and tickets to enable them to read the data more effectively. They also wanted a way to export the report data to an excel or csv file. For the ticket page, our card-based design was liked but they added suggestions in terms of unneeded fields such as "Assigned to" be replaced with terminal ID and location name. In terms of the Route-planning page, they suggested the idea of also displaying the tickets to the side of the Route planning table to remind them of the terminals that a client has requested of them to check out.

USER TESTIMONIALS

This project is being developed for a real company, so our feedback is from their employees to determine if our solution works for them. The feedback from the employees has also been primarily positive with some complaints

that are similar to our Domain Expert's evaluation of our current work. Mentioning similar complaints about the user interface for the tickets and reports having unnecessary fields or missing fields that would more easily identify the customer or location of the terminal. These are issues that are easy enough to adjust and fix. We plan on implementing these changes in the span of next week.

FUTURE WORK

We will continue working on this project in partnership with WongATM to build a solution that works for them. Once the application solves WongATM's issues they have expressed interest in generalizing the application to be used for other management and administrative purposes. Developing insights and displaying reports can be used and expanded upon in many other industries, particularly retail businesses where shipping costs and profit margins need to be monitored closely.

As of now the implementation of the Route planning for WongATM, we need the approval of our client allowing us to use WorkWave and its respective API as was mentioned in our defence. If we tried to develop a custom solution on our own the scope of the project would be too large for just the three of us alone. There is documentation for integrating WorkWave, but it states we need knowledge of WorkWave's web application, features and terminology. While we are familiar with the WorkWave application in terms of how it is used, we have not been given access to manipulate and use it ourselves because of how costly it is for a license that doubles in price for every car added. When given access we would receive a key for a query-string parameter or an HTTP header that would allow us to access, manipulate, place and deploy routes.

As an alternative, we were looking into Google maps as they were an easier option for us to deploy a route with but when asking our client about what service they wanted us to use they insisted we use WorkWave as they would not have to retrain their field workers. Resulting in our current situation with the project. The Google Maps method that we were thinking of using would use the Google Maps Platform such as the Dynamic Maps API and show their location on the map. Once the selected terminals were verified, we would then upload their location data (Longitude and Latitude) to the DynamoDB where the android application would then take over. Once the data is collected and a fieldworker logs into the android application. The application would then take the Longitude and latitude and create a route using Google maps.

BIBLIOGRAPHY

R. and Segment Forecasts, "Self-service technology market Size REPORT, 2020-2027," *Self-service Technology Market Size Report, 2020-2027*. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/self-service-technology-market. [Accessed: 17-Sep-2021].

G. of Canada, "Government of Canada / gouvernement du Canada," *Finance and insurance - 52 - Summary - Canadian Industry Statistics - Innovation, Science and Economic Development Canada*, 01-Mar-2019. [Online]. Available: https://www.ic.gc.ca/app/scr/app/cis/summary-sommaire/52. [Accessed: 17-Sep-2021].