

Sheridan College

SOURCE: Sheridan Institutional Repository

Student Capstones

Honours Bachelor of Computer Science (Mobile Computing)

Fall 12-9-2022

Arcana Capstone Report

Jae Won Park

Sheridan College, parjaewo@sheridancollege.ca

Suzi Ryu

Sheridan College, ryusu@sheridancollege.ca

Caitlin Rush

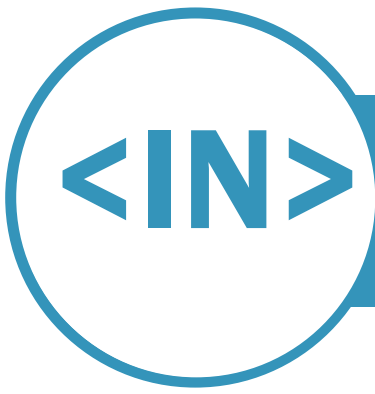
Sheridan College, rushca@sheridancollege.ca

Follow this and additional works at: https://source.sheridancollege.ca/fast_sw_mobile_computing_capstones

Recommended Citation

Park, Jae Won; Ryu, Suzi; and Rush, Caitlin, "Arcana Capstone Report" (2022). *Student Capstones*. 14. https://source.sheridancollege.ca/fast_sw_mobile_computing_capstones/14

This Capstone Open Access is brought to you for free and open access by the Honours Bachelor of Computer Science (Mobile Computing) at SOURCE: Sheridan Institutional Repository. It has been accepted for inclusion in Student Capstones by an authorized administrator of SOURCE: Sheridan Institutional Repository. For more information, please contact source@sheridancollege.ca.



ARCANA

VIDEO GAME, OUTDOOR ACTIVITIES | CAPSTONE PROJECT
HONOURS BACHELOR OF COMPUTER SCIENCE
(MOBILE COMPUTING)

STUDENT TEAM

Jae Won Park, 4th Year Student

E: parjaewo@sheridancollege.ca

Suzi Ryu, 4th Year Student

E: ryusu@sheridancollege.ca

Caitlin Rush 4th Year Student

E: rushca@sheridancollege.ca

SUPERVISOR

Prof. Ricardo Paharsingh

E: ricardo.paharsingh@sheridancollege.ca

Sheridan College

DOMAIN EXPERT

Prof. Will Barry

E: will.barry@sheridancollege.ca

Sheridan College

ABSTRACT

Our project, Arcana, is an augmented reality based mobile game. This application will promote outdoor activities for both children and adults and provide them with fun and engaging challenges. A user can grow their unique character and progress through puzzles and encounters. This solution will use many advanced areas of computer science, such as augmented reality and cognitive computing, and have many different components like cloud-computing and multiplayer functionality.

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](#)

Table of Contents

STUDENT TEAM	1
SUPERVISOR	1
DOMAIN EXPERT	1
ABSTRACT	1
ABOUT CAPSTONE PROJECTS	1
Introduction.....	3
Project Overview	3
Domain and Industry Overview	4
Problem Description	4
Solution Description	4
Mobile Computing	4
Cloud Computing.....	5
Advanced Areas of Computer Science.....	5
Solution Impact	5
Solution Feasibility	5
Design and Construction.....	5
Deployment	6
Adoption.....	6
Project Management.....	6
Responsibility Matrix	6
Release Plan	7
Risks	8
Project Requirements	8
Project Architecture	10
Validation and Testing	11
Testing Strategy.....	17
Validation Results	18
Conclusion	19
Future Work.....	20
Bibliography.....	21
Appendix.....	22
Appendix A.....	22
Appendix B	23
Appendix C.....	38

INTRODUCTION

The purpose of this document is to give a general overview of our proposed solution. The first revision, 0.1, is completed through collaboration in September and October 2021 and includes the following sections that go into detail regarding the industry, the problem description, how we will develop the solution and the advanced computer science components that we wish to use, and lastly, the feasibility.

Revision 0.2 was made October 2021 following work on the Project Plan which is meant to divide and allocate the project work for our team members into iterations of two-week sprints. During this revision, a responsibility matrix is added to further explain the tasks assigned to each member.

Revision 0.3 was edited during December 2021 which includes information and diagrams of models made in Visual Paradigm to describe the architecture we chose for our solution. Also in this revision, validation is added, including testing strategies and results with links to the JIRA Test Plan. Our conclusion is added in this revision with remarks about further development for this project.

Revision 1.0 marks the beginning of the second half of the capstone project starting from September, 2022. This revision includes changes that were requested from feedback on our defense last year. As a group, and in our respective functional areas, we enhanced the requirements model, the software architecture diagrams, the interaction models, and deployment model. Further details are outlined in the following project architecture section below.

Revision 1.1 involves the updates to the work breakdown structure defined in our project management plan in JIRA. All tasks have been accounted for and allocated into the appropriate iterations in our release plan. At this time, we also reviewed the risks and the state of their relevancy.

Revision 1.2 accompanies the submission of our Alpha Release. This Capstone Report document, formerly called the Project Inception Document, includes more edits and revisions to our models, overview and appendices.

Revision 1.3 follows our Beta Release submission. Our group has continued to add to our Arcana solution, which has been noted in the following report. We added more to our models, testing strategies and functionality.

PROJECT OVERVIEW

Our project, Arcana, is an augmented reality based mobile game that will consist of advanced computer science features and technical areas that brings the user outdoors and challenges them while having fun. The project's inception takes place in Fall 2021 and the further creation will happen in Fall 2022. The group members for this project are Jae Won Park, Suzi Ryu, and Caitlin Rush. All group members are responsible for full stack development and at least one functional area and stakeholder. Other sole responsibilities are mentioned below. Jae, who is the original creator of Arcana's expression of interest, has the role of Project Owner. He is responsible for being the

requirements and business analyst, lead of requirements model, lead of domain model, lead of test model, and providing communication support. Suzi is Arcana's SCRUM Master and she is taking the responsibilities of software architect, design model lead, deployment model lead, and interaction model lead. Caitlin is taking the role of Risk Analyst and will have the following responsibilities: integration lead, quality assurance lead, providing tools and device support, and leading user experiencing design. Our domain expert is Professor William Barry, who was the original supervisor of this project, and his expertise in the gaming industry will allow us to better assess the solutions feasibility and provide us with insight and advice. This following sections in this document go into detail about the industry and domain related to this project and the description of the problem and how we intend to solve it with our solution. It goes into further detail on which specific advanced computer science areas we are making use of and how it will help our project stand out against competitors. The following sections explain the impact that our project will have on the industry and to the stakeholders. Finally, this revision of our project inception document will go over the overall feasibility of our solution.

DOMAIN AND INDUSTRY OVERVIEW

Software Publishers – 5112 – “This industry comprises establishments primarily engaged in publishing computer software, usually for multiple clients and generally referred to as packaged software. Establishments in this industry carry out operations necessary for producing and distributing computer software, such as designing, providing documentation, assisting in installation and providing support services to software purchasers.” [1] There are 3,187 establishments in this industry, with an average revenue in 2019 of \$663.9 thousand and 76.1% profitable.

PROBLEM DESCRIPTION

The problem we are trying to solve with the creation of our project, Arcana, is to get people outdoors. Over these past couple of years, we've all been bounced between lockdowns and restrictions and have seen many changes to our daily activities and social lives. Our application will aim to promote outdoor activities and provide fun and challenging methods for children and adults to go outside, have fun, and engage in both physical and mental challenges.

SOLUTION DESCRIPTION

To solve the problem listed above, Arcana will promote outdoor activities by having many outdoor events that attracts players to socially gather at specific locations. Using Google Map's key location features, key features will turn into game locations where the players will have to physically walk near the vicinity and the game will recognize the players location through Google map services. Using this data, the application will start a game with the players nearby, allowing the players to form a community that goes outdoors.

Mobile Computing

In terms of mobile computing, this application will be deployed on iOS devices, such as iPhone and iPad. The game will make use of the device camera functionality and location services. using the camera functionality on mobile

devices, the in-app application will use the camera to show the player the life feed on camera with simulated enemies. Also, the built-in pedometer is a feature that we will be heavily using in our solution.

Cloud Computing

This solution uses cloud-computing through offering IaaS service where users can login to their account and have the boss battle effects shown in AR. More specifically, Firebase is used a lot for account management. This cloud service allows us to authenticate users that are signing up and logging into the application. Also, Firestore is used to store more of our other user data such as the role they have chosen, their usernames, and the character inventory. More cloud computing features have been implemented using Google Cloud's DialogFlow ES chatbot platform. This cognitive computing functionality provides our application with a customer service type of chatbot that allows user to gain more insight and send us solutions or log bugs they encounter while playing the game.

Advanced Areas of Computer Science

The primary differentiator for our project, in terms of an advanced area of computer science, is through the use of augmented reality and cognitive computing. As users proceed walking around in the real world, monsters and creatures will generate in their devices through AR. The simulated fights will have the user physically dodging projectiles or using their character's unique abilities to defeat the boss in order to level up the character and progress in the game.

SOLUTION IMPACT

This solution will have a great impact on the industry. The mobile game industry has been one of the most profiting technology industries in the latest decade. Especially with the start of Covid-19, many of the game industry has been booming due to many students, adults and children being at home. In 2020 only, the mobile game industry has reached a record sale of 10.73 billion US dollars. As we've seen with Pokemon GO, these kinds of augmented reality applications are fun and popular. Our application will give both kids and adults another fun and challenging game to play outside. To put in perspective, it has been released that Pokemon GO players spent around \$1.3 billion in 2020, besting its record sales from 2019. In conclusion, using location services and AR services to render the world of adventure into real-life using camera, this solution will help in solving a real-world problem of getting children outside.

SOLUTION FEASIBILITY

The sections below further explain the solutions feasibility in regard to the design and construction of the project, deployment of the solution, and once it has been deployed, the feasibility of adoption.

Design and Construction

Overall, the design and construction feasibility is promising. There are a few risks that could interfere with the creation of our application which could slow down the project or cause other issues to arise. One of the risks being that some location services are inaccurate. Our application depends on tracking the users movement and steps they

have taken in order to simulate AR monsters. So, if the location is not entirely accurate, it may be frustrating to correctly develop this feature of the solution.

Deployment

In deployment, the services of the game will be hosted on cloud server. The character information and monster information will be on everyone’s devices. Each device will carry the basic game information such as monster statistics, patterns, mini-games and on. The cloud servicing will be done to connect the players through multiplayer component. For the capstone project, the project will first focus on individual component and displaying the game services through individual devices. Next, the multiplayer component where the actions of one player will reflect on another player will be implemented.

Adoption

The adoption of this project has a high feasibility for many reasons. Firstly, augmented reality mobile games are very popular. Pokemon GO, for example, was a huge success and it is still a globally used game. AR is exciting and intriguing to users and the fact we are using it to make a game, also makes it fun. The application won’t be costly to the user, and this also increases the adoption feasibility.

PROJECT MANAGEMENT

RESPONSIBILITY MATRIX

Project Responsibility	Jae Won Park	Suzi Ryu	Caitlin Rush
Project Management			
Project Owner	V		
SCRUM Master		V	
Risk Analyst			V
Requirements Engineering			
Requirements/Business Analyst	V		
Stakeholder Champion (by Stakeholder)			V
Functional Area Champion (by Functional Area)	V		
User Experience Design Lead		V	
Software Architecture			
Software Architect	V	V	V

Requirements Model Lead	V		
Domain Model Lead			V
Design Model Lead		V	
Deployment Model Lead	V		
Interaction Model Lead			V
Construction			
Full-stack Dev (UI, Code and unit-tests)	V	V	V
Integration /DevOps Lead	V	V	V
Testing			
QA Lead	V	V	V
Verifications & Validation Champion (by Functional Area)			V
Test Model Lead		V	
Support			
Tool and Devices Support	V	V	
Communication Support			V

RELEASE PLAN

The release plan of Arcana project will be divided into multiple releases. These releases will be aimed for specific due dates.

1. Inception Release
2. Architectural Release
3. Elaboration Release

With the following second phase of our capstone, we will be dividing our project to multiple releases again. These releases will be aimed for specific due dates.

1. Alpha Release
2. Beta Release
3. Final Release

All iterations have been arranged within the Project Management Plan within JIRA found [here](#), where all use cases are identified and the tasks have been accounted for and allocated into the appropriate iterations. Building off of our models, we can break the use cases into either user stories, dev stories, or system stories and assigned them accordingly. There are also many different types of use cases such as infrastructure, research, and support. Each of these are also further broken down to monitor workload and minimize interdependencies.

RISKS

Our Risk Management Plan found [here](#), outlines the risks associated with our project. We have different phases that the risks go through depending on their state. First, we identify the risk and place it in the Identification column. After this risk has been analyzed to determine the probability and impact, it gets placed into the Analysis column. From there the mitigation strategies are being determined and while this is happening, the risk is placed into the Planning column. The risk moves into the Monitoring column as we watch it, and we record updates in the comments. If a risk should arise, it gets placed into the Active column and subtasks will be assigned to mitigate the risk. Finally, once the risk is no longer active, it gets put into the Resolved column because it is no longer relevant. Each risk is also categorized into either Product Risk, Project Risk, or Business Risk to better organize development. Also, each risk is labeled according to its probability and the impact it would have on our solution.

We have identified many risks that are probable regarding our solution. One being technical issues. For example, if the user is experiencing technical issues on their device, it will inhibit them from playing our game and it will be a problem for us as developers. We need to create our project with updated software and make sure its suitable for as many devices as possible. Another related risk would be network failures. Like the technical risk, if the network fails, it means an even greater amount of people cannot access our application. This risk is a bit more out of our control, since it would be the network's responsibility to get service back up and running, but it is still a risk we would have to monitor as we continue to develop our solution. Another big risk that we have identified is not finishing our project on time. We face a higher probability of this risk compared to other groups because we have started a little later due to the change of supervisors. We continue to monitor this as it is a very serious risk. To prevent this risk from becoming active, we just need to stay organized and keep on track of submitting on time. While developing we may face the risk of certain APIs or location services not working properly. Either they expire, get discontinued, or are down for updates, this is out of our control but serious as it could delay our solution from being developed. This is monitored regularly and if subscriptions are required, they are renewed on time. Last but not least, a serious risk we have identified is issues amongst our group. Although it has a low probability because we have worked together before, it would be a serious issue to our solution if this risk was active. To mitigate and prevent this risk, we have open communication and regular meetings.

During our project, we continue to monitor and analyze the risks we have identified to assess their relevancy to the solution. After starting the second half of our capstone, all risks are out of the identification phase and have made their way across the board. If they are still relevant, we continue to monitor and analyze them. A few of the risks, such as the college striking or having another lockdown, are no longer relevant so they are completed off the board.

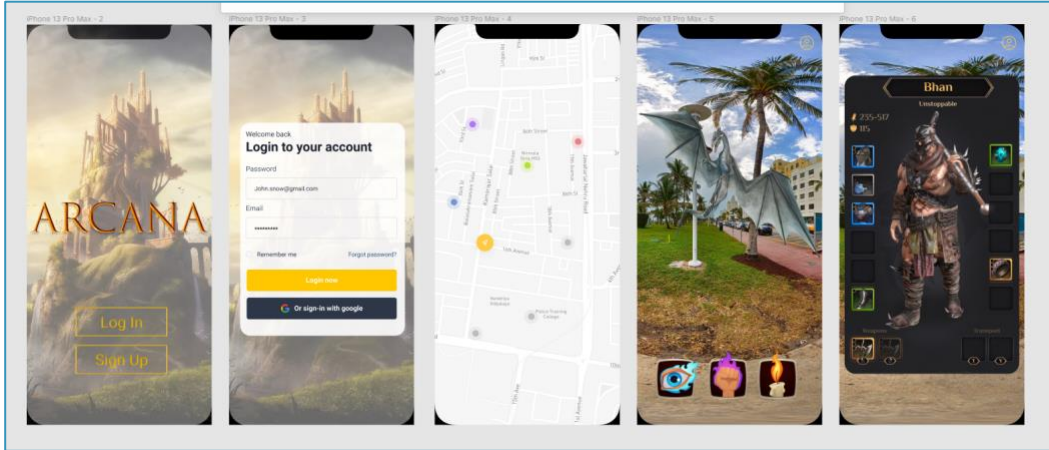
PROJECT REQUIREMENTS

Project Arcana has a simple system context diagram as well as simple functional areas.

The three major use-case in Arcana is

1. The user can interact with the characters, view their profiles, check their stats, and customize character.
2. The user's movement will be displayed on a map of their surroundings and the user will walk to point-of-interest which will create encounters.

- The user will battle with monsters in encounters for prizes and experience, which will further improve the user's character.



Figma Diagram 1: <https://www.figma.com/file/2UqvPNUP9orbnj8Y7v8vd8/Arcana?node-id=0%3A1>

PROJECT ARCHITECTURE

The project solution is structured in a MVC architecture.

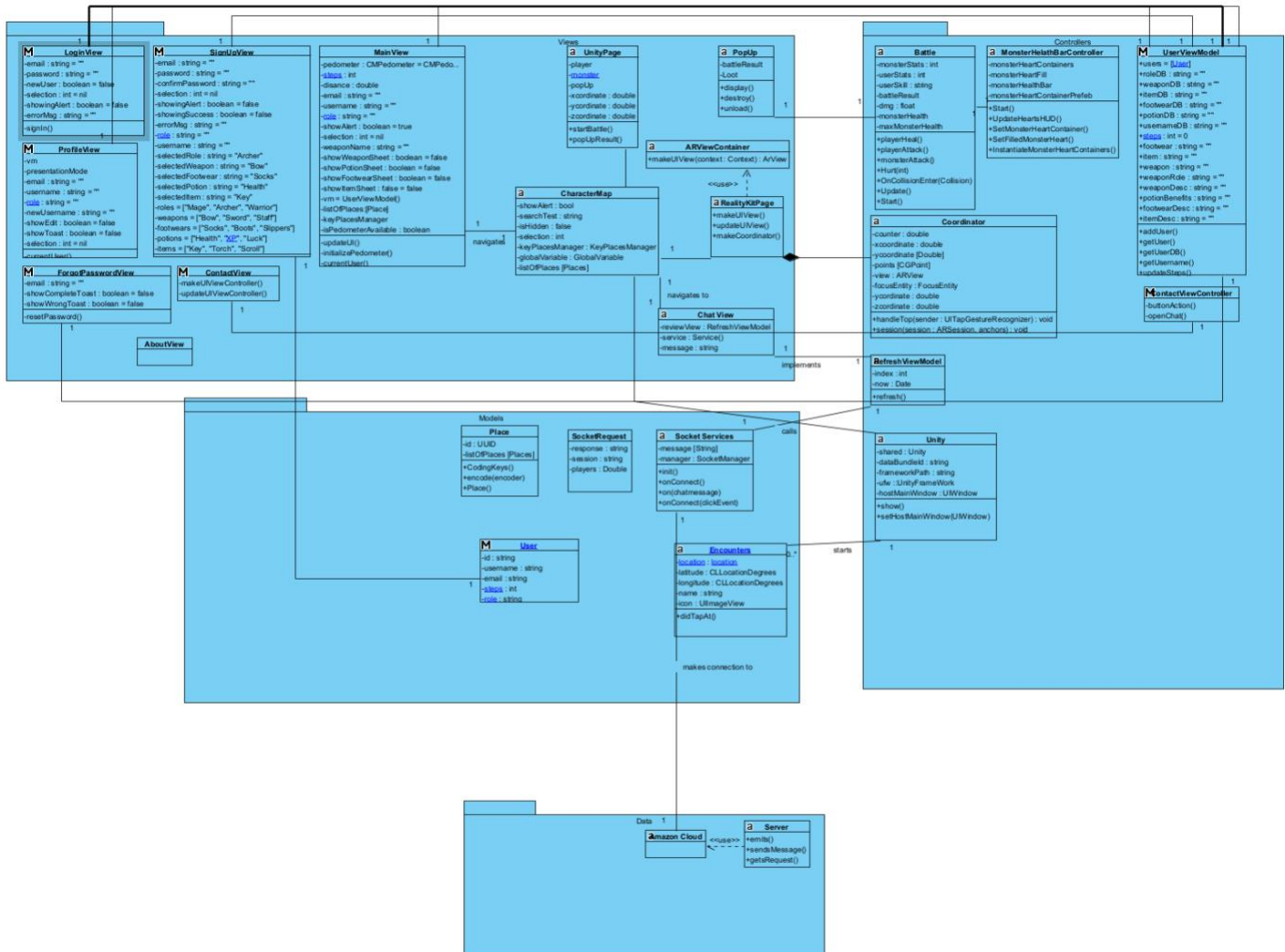


Figure 1: Architecture in VP

Our project is composed of View, Model and Controller architecture

In our view layer, it contains the user UI, in-game map, and the logic behind them. This is composed of main features of our application where the user will have their character displayed on map and interact with their characters by changing gears or looking at different character builds. Our view layer is also where the Augmented Reality game created through Unity is created.

In our controller layer, we have allocated the different cloud services we need for the functionality of our program. The location services provide the necessary data for player location and event locations while the character

management and multiplayer service will be API services. The major use-case in this layer is the player physically walking around real-life locations to participate in battles and trigger monster events.

In our model, we contain all the necessary information about the models of our application such as users, game characters, monsters and on.

In our persistence layer, we have Unity resources needed for rendering and displaying AR components that are related with the character UI. The major use-case in this layer is the monster encounter AR use-case, where the player's character will be placed in an AR battle with our monster.

In our data section, we have a cloud component and our Firebase database that consists of user, character, and monster data. The main use cases for data layer are the user login and profile related use case.

We have decided on the layered architecture because of its flexibility and simplicity. We have outlined the use-cases needed and the different types of services that we'll need for this project. From this, we have determined that the layered architecture is best suited in dividing up the roles of each individual layers while some layers such as Service layer and presentation layer will have interactions to correctly display character movement on the mobile device.

For the second half of our capstone project, we have reviewed the defense feedback given to us and incorporated the changes into our models, either enhancing current models or adding new ones on top of what we already have. For example, it was suggested to us that we need to have more detailed specification regarding requirements on the use cases. Flow of events were added to the use cases in our requirements model to further visualize these requirements. Also, the interaction model section was revised to account for the use cases that have already been established in the solution with the addition of structural overviews, behavioural overviews, and collaboration overviews for each use case.

Our overall software architecture main class diagram was fine-tuned according to feedback as well, which can be seen in the figure above. We focused on polishing up different functional areas of our application while maintaining the architecture we designed originally.

VALIDATION AND TESTING

The following section will outline our strategies to test our implementation of different use cases according to the solution's requirements. Firstly, our team worked together in Visual Paradigm to create a validation model that is responsible for outlining our initial test plans, test suites, and test cases. Within packages according to the addressed functional areas, there are requirements diagrams to form these validation processes.

Figures 2 to 4 are screenshots of the test plans made according to each functional area that were performed in our first release. The follow figures numbered 5 to 11 show the further break down of the test cases found within the test plans.

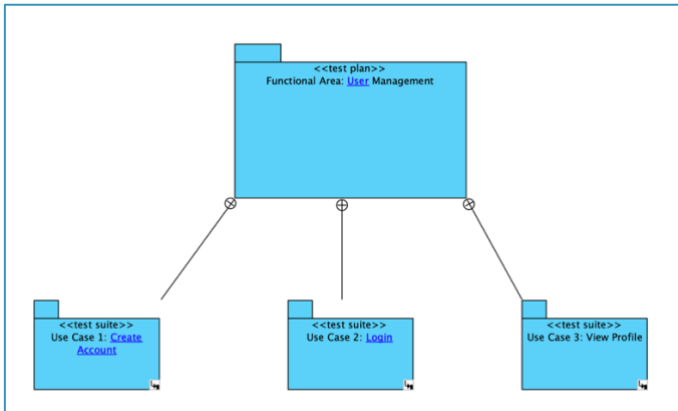


Figure 2: Functional Area 1 Test Plan

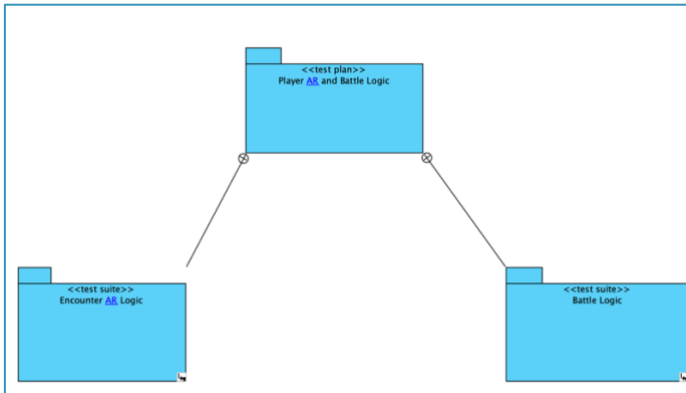


Figure 3: Functional Area 2 Test Plan

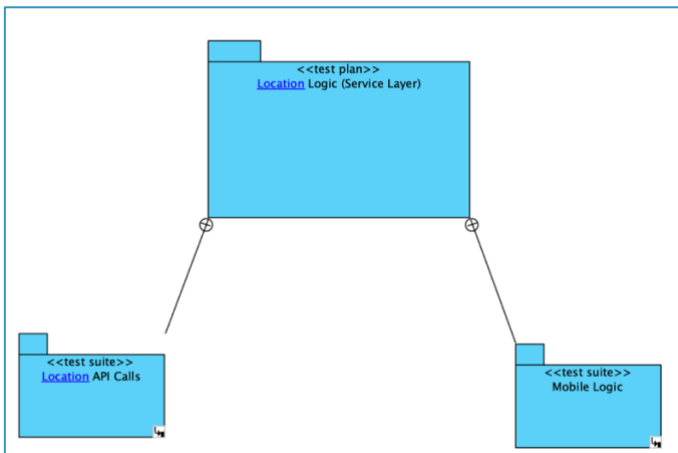


Figure 4: Functional Area 3 Test Plan

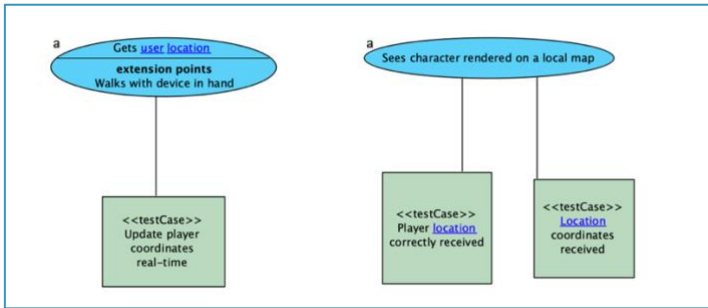


Figure 5: Location API Calls Use Case Test Plan

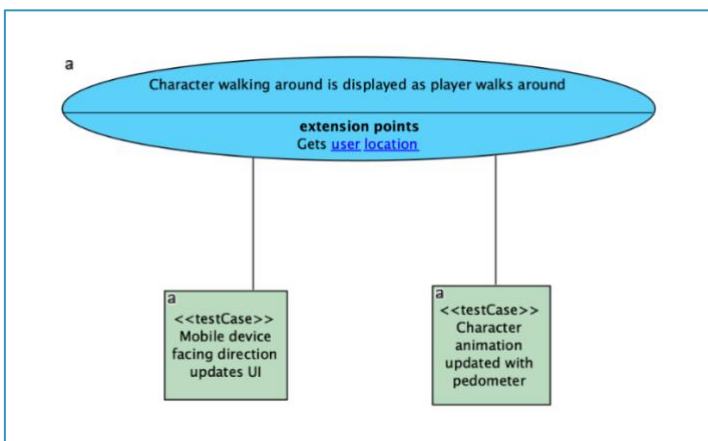


Figure 6: Mobile Logic Use Case Test Plan

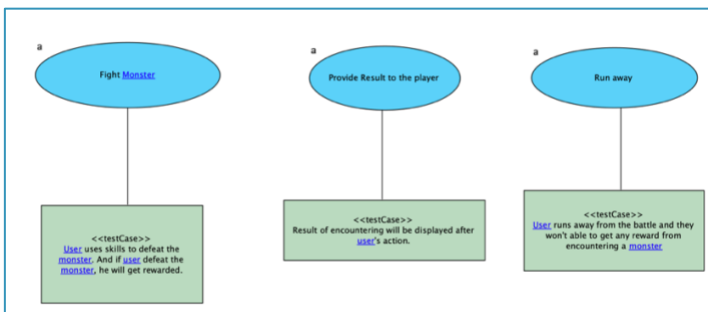


Figure 7: Battle Logic Requirement Use Case Test Plan

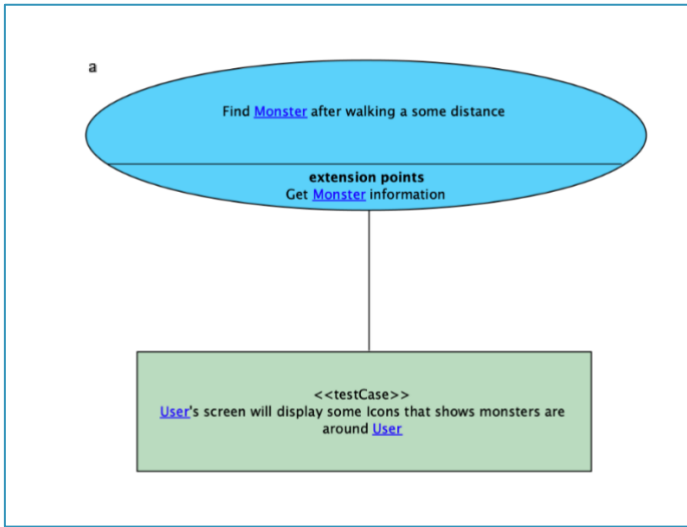


Figure 8: Encounter Monster Use Case Test Plan

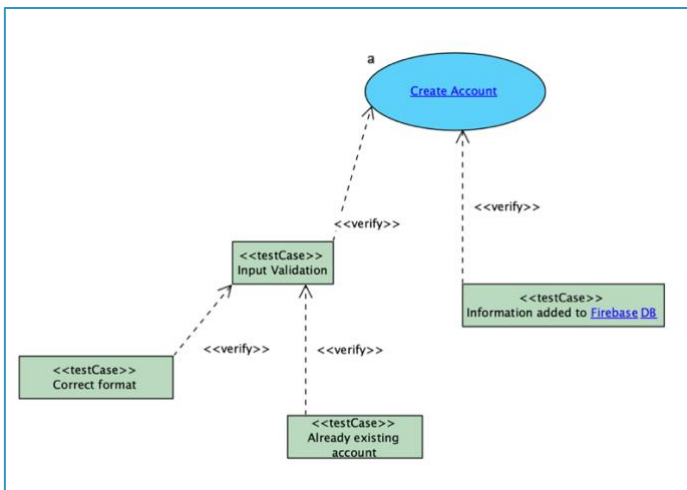


Figure 9: Create Account Use Case Test Plan

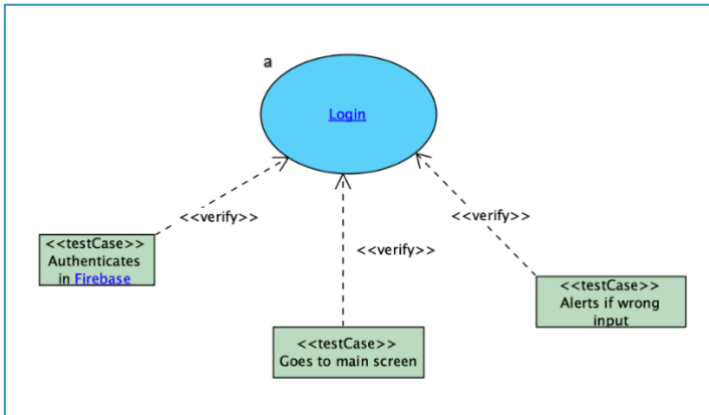


Figure 10: Login Use Case Test Plan

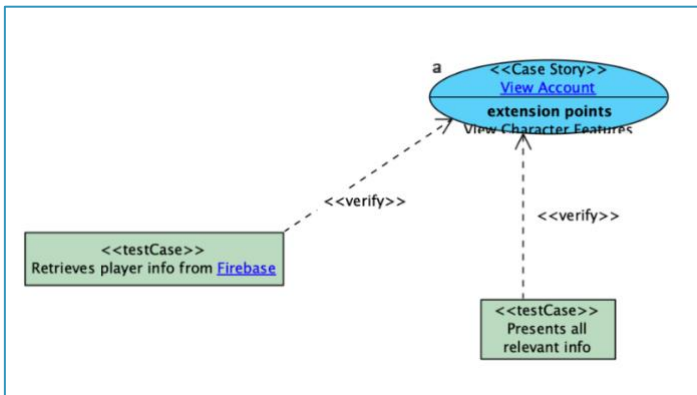


Figure 11: View Account Use Case Test Plan

For each of the above test cases, our team also outlined the steps needed to carry out this test, how to execute the said steps, and finally, the expected result of the test case steps. For example, in Figure 11, you can see the steps needed to test the authentication in Firebase when the user logs into the application. In our code, we execute this connection, and checking the database, and the expected result would be displaying a message, such as in terminal to confirm the successful connection and sign into the application. After this, other test cases would be verified, such as if the screen would then navigate to the main screen or not.

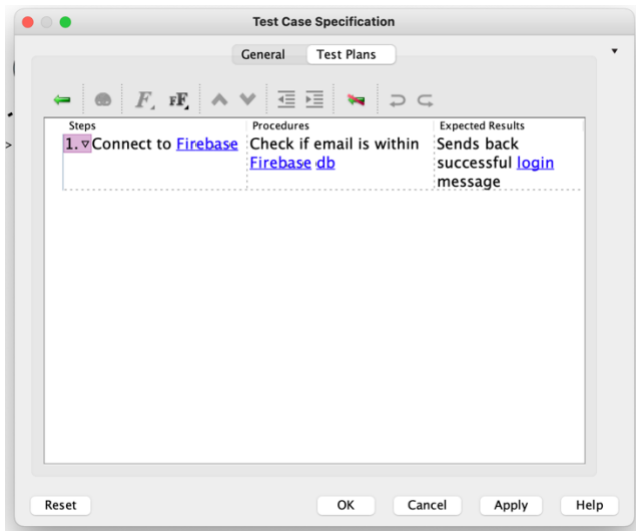


Figure 12: Test Case Steps for Login Authentication

In revision 1.3, for Beta Release, the validation models within Visual Paradigm were updated with a lot more detail. As you can see in Figure 13 below, and compare it to Figure 10 above, there have been many changes. The test cases are organized properly, and each test case has steps and expected outputs found under its details.

I have also included Figure 14 below, which is the test plan for the Reset Password Test Suite, to show a variety of the updates made. There is proper flow to the use cases, such as which test cases are used to verify and which have a derive dependency.

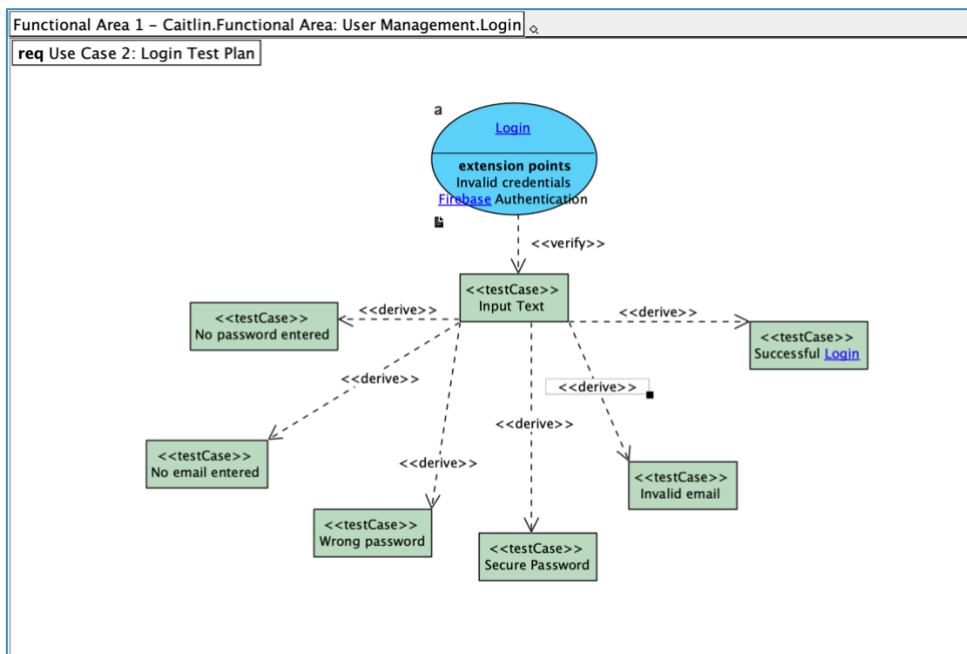


Figure 13: Updated Test Plan for Login

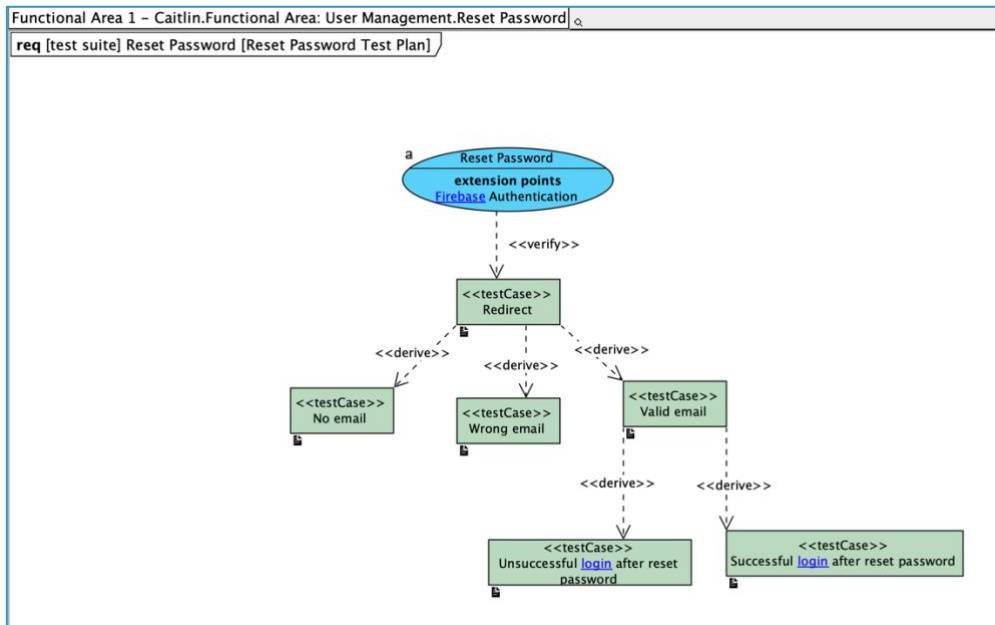


Figure 14: Updated Test Plan for Reset Password

TESTING STRATEGY

In addition to our testing diagrams in Visual Paradigm, we have outlined our test cases within a Test Plan in JIRA located [here](#) that follows the same kind of structure. An example of one of our tasks is shown below in Figure 15.

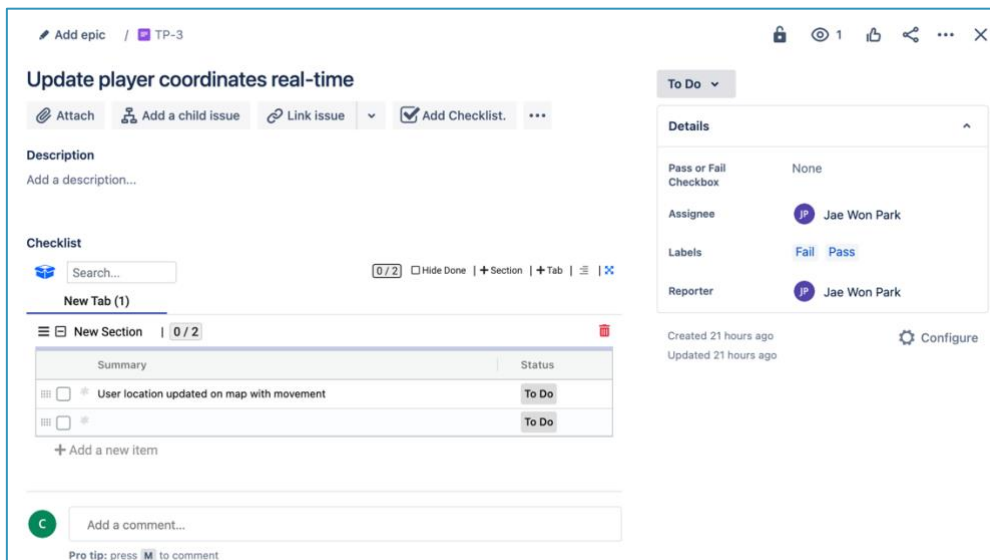


Figure 15: A Test Case Task in JIRA

As we develop our solution and implement more of the features and user requirements, we can add these tasks to represent test cases in our test plan. The test cases are allocated to everyone according to the functional area they are responsible for, and we have pass and fail tags to identify the result when executed. The detailed findings of our

tests will be explained in comments and if the test did not pass, we can add an issue in our BitBucket bug tracking system to get that fixed.

For revision 1.3 and Beta Release, the Testing Strategy has improved. There has been more organization implemented in our Test Plan within JIRA. For example, as seen in Figure 17, Test Cases are now arranged into proper Test Suites, called Epics within JIRA, to add another level of clarity and organization.

Also, as seen in Figure 16, a completed Test Case for a successful login, the test case itself has improved. Each test case has a clear description, expected result and a checklist of completed steps. We make sure to add a pass or fail tag and checkbox to keep track of our projects test cases.

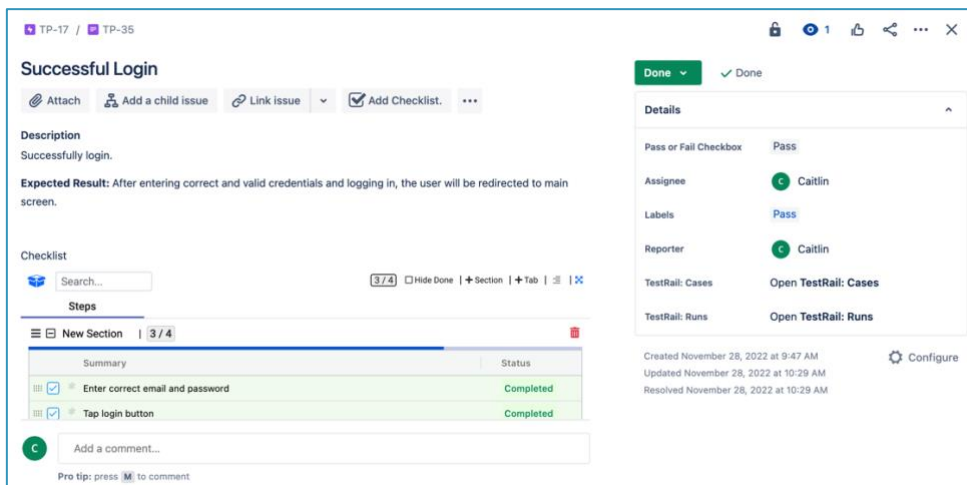


Figure 16: An Updated Test Case Task in JIRA

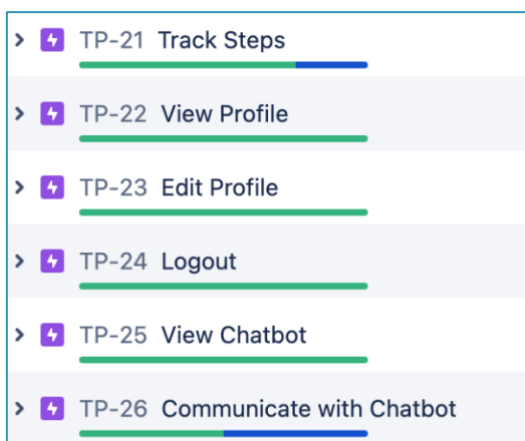


Figure 17: Test Suites / Epics in JIRA

VALIDATION RESULTS

As of now, out of 15 issues and test cases that were stated, 11 of them are complete and only 4 are left in progress. As for the user management functional area, most tests pass. The first run through of signing up and signing in checked off a lot of the steps and expected outcomes. From this inspection, Caitlin added the implementation of alerts to validate user inputs and it will notify depending on the error code returned from Firebase, such as incorrect email, weak password, and wrong credentials. Adding this step checked off more boxes for the test cases. One issue still ongoing is viewing and changing user profile information. Due to some Xcode issues with package dependencies, some issues were encountered with Firestore and it put a delay on some features. In terms of location related services, the user can correctly update their location to the database and the UI is reflected to user's location as well as the user's movement. Since this will be one of the core UI screens that the user will see, it is a big component in our project. As for the AR component, Suzi's focus has been focusing on the displaying AR, attaching it to the application and providing a good experience to the user. The test cases that have yet to be completed are more related with navigation test cases rather than the AR itself. In all, the validation results are well completed.

As of revision 1.3, which is related to our Beta Release, the test cases have more than doubled. Testing is an ongoing process and conducted alongside our development. Previous outstanding tests have passed, and new ones are in progress. As we near the end of our release, a final run through of regression testing is also done to ensure everything is in working order.

CONCLUSION

For earlier submissions, a video demonstration of our first release of the cloud computing aspect can be found [here](#). In this submission, users could sign up and sign into the app using an authentication service from Firebase. Another service from Firebase is using their cloud database, Firestore. When a user signs up for our app, more of their information is stored into our user's database, such as their username and role they want their character to be, such as warrior, mage, or archer.

With the Alpha Release, more features have been added. Users now can customize their characters. Now with sign up, inventory items can be selected for each character. A user has different options for weapons, potions, items, and even footwear. When inside the app, a more detailed description and information about each of the inventory objects that were selected can be seen on sheets that pop up on screen. A user can also view their profile information, edit their username, and request their password to be reset.

For the Beta Release, there were a lot of user interface adjustments and features added. A customer service type of chatbot was added as a cognitive computing component to our application. This bot answers general information about the application, such as how to play, or what to do in certain situations, and how to contact us further. It can also allow the user to log bugs they experience or submit suggestions they have for the app. This is a great way to further improve our application in the future. It builds reliability and trust between the users and us as the developers. Beta release also included a connection to multiplayer services. We also improved upon our AR component by adding extra assets and features to our Unity as External Library

FUTURE WORK

There is a lot more exciting features we can add to Arcana over the next year. We are planning to add a multiplayer component so a user can play alongside their friends and help each other to battle monsters, specifically boss fights. In terms of character management, in future releases we hope to implement logic that will combine the inventory items with certain advantages or disadvantages when a character is leveling up. For example, an archer's best weapon is a bow. So, if a user selects their role as archer and weapon as bow during character customization when signing up, it should give them a boost or level their character up straight away. We also plan to add loot and rewards for characters in certain key spots which will increase user and business owner satisfaction. Another interesting feature we want to add is a customer service chat bot, or storyteller NPC chat bot using a cloud platform which will increase the user's immersion into the game.

We aim to implement business users so they can create hotspots, key zones for our application.

We also hope to add editor functions, so it is easier to create monsters and add more challenges to the users.

We are also considering adding the option of deploying this solution to Android devices as well, so more users can enjoy our application.

BIBLIOGRAPHY

- [1]“Government of Canada / gouvernement du Canada,” *Innovation, Science and Economic Development Canada*, 01-Mar-2019. [Online]. Available: <https://www.ic.gc.ca/app/scr/app/cis/search-recherche>. [Accessed: 27-Sep-2021].
- Patil, R., Nema, S., & Kadam, S. (2017). Radio frequency identification system for asset tracking and inventory management in hospitals . Noida: IEEE.
- Reinhardt, U. E. (2000). The economics of for-profit and not-for-profit hospitals. Chevy Chase: The People to People Health Foundation, Inc., Project HOPE.
- Silber, J. H., Rosenbaum, P. R., Ross, R. N., Ludwig, J. M., Wang, W., & Niknam, B. A. (2014). Template matching for auditing hospital cost and quality . Gale.
- USPRwire. (2013). Report: Hospital Asset Management Market - Pharmaceutical - Global Forecast to 2017 . Infotrac Newsstand.
- Weil, A. R. (2015). Hospital Costs And Quality. Chevy Chase: The People to People Health Foundation, Inc., Project HOPE.
- Mathijs de Haas, Roel Faber, Marije Hamersma, How COVID-19 and the Dutch ‘intelligent lockdown’ change activities, work and travel behaviour: Evidence from longitudinal data in the Netherlands, Transportation Research Interdisciplinary Perspectives
- Alexander W. Bartik, Marianne Bertrand, Zoe Cullen, Edward L. Glaeser, Michael Luca, Christopher Stanton Proceedings of the National Academy of Sciences Jul 2020, 117 (30) 17656-17666; DOI: 10.1073/pnas.2006991117
- Brooks, S., & Webster, R. (2020, February 26). The psychological impact of quarantine and how to reduce it: rapid review of the evidence. THE LANCET. Retrieved December 15, 2021, from [https://www.thelancet.com/article/S0140-6736\(20\)30460-8/fulltext](https://www.thelancet.com/article/S0140-6736(20)30460-8/fulltext)

APPENDIX

Appendix A

Link to Arcana Bitbucket Repo

<https://bitbucket.org/caitlinrush/capstone/src/main/>

Link to Unity Standalone Bitbucket Repo

<https://bitbucket.org/suziryu1991/ar-arcana/src/main/>

Appendix B

Software Requirements Specification

Capstone Project Arcana

Software Requirements Specification

INFO 45828
CS Capstone Project

Authors
Jae Won Park
Caitlin Rush
Suzi Ryu

Professor Richardo Paharsingh

Table of Contents

1 System Context Diagram.....4

2 System Overview Diagram.....4

3 Glossary Grid.....6

4 Business Requirements Business Rule Grid.....7

5 Domain Model Class Diagram.....11

6 Functional Area 1 Caitlin Use Case Diagram..... 12

7 Functional Area 2 Suzi Use Case Diagram..... 13 8 Functional Area
3 Jae Won Use Case Diagram.....14

1. Introduction

2. System Overview

 2.1. List of Actors and their description

 2.2. System Context Diagram and its description

 2.3. Functional Areas and their descriptions and an overall use-case diagram for each functional area

3. System Glossary

4. Business requirements

 4.1 System Business Requirements

 4.2 Business Requirements by functional area

5. User Requirements Summary

 5.1 Table of use-cases and their descriptions including author

6. Domain Overview

 6.1 Domain diagram and it's description

7. Use cases

Section 1: Introduction

The purpose of this document is to demonstrate Software Requirements Specification for our capstone project Arcana.

Project Arcana is game developed by Jae Won Park, Suzi Ryu and Caitlin Rush, overlooked by professor Ricardo Paharsingh to tackle a real world problem caused by Covid-19 social distancing. When the project was first brought to life, social distancing and limited outdoor activities caused problems where students, children and adults would not go outside for outdoor activities. This project aimed to create an application that offered fun and co-op playability to promote outdoor activities while having fun and meeting new people.

Section 2: System Overview

The primary actor for project Arcana is the players.

Players are part of this game system. Players play the game and the application. They are able to create a user profile and create their game character. Players take their physical mobile devices and walk around the neighborhood to play the game. Players grow their characters by leveling it up through adventures and battling monsters. Players are able to interact with other players.

The other primary actor for this project is the business owners.

Business owners - Business owners target the marketing portion of our application. Business owners will be able to designate key areas as key target locations to draw people in for events and encounters

Amazon Cloud Services - Amazon Cloud Services (Amazon EC2) provides a cloud computing platform for socket connections to enable multi-player networking component of Arcana

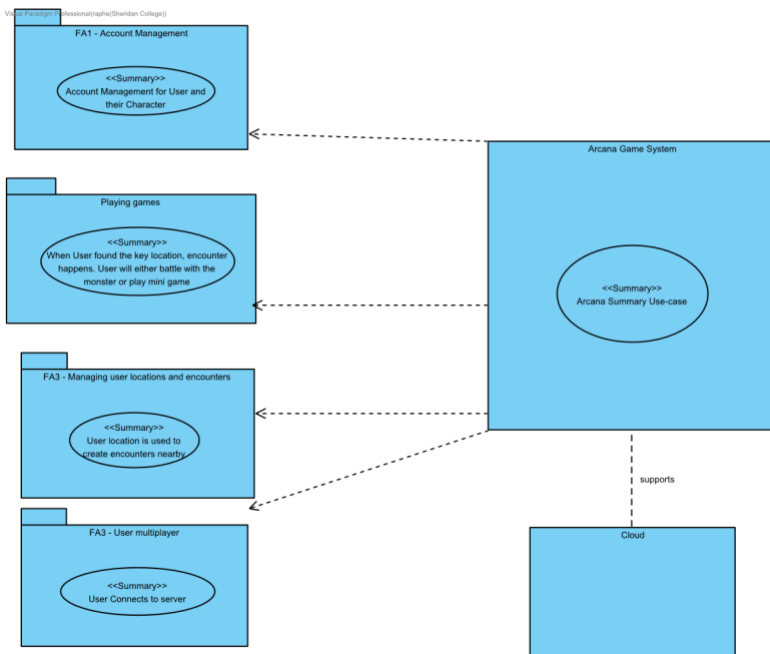
Google Database and Cloud services - Google Database hosts all the data of users and characters for Arcana. Google Cloud Services provide Locations and Places API that the application uses to provide location related services.

1. System Context Diagram



System Context diagram shows the overarching actors involved in Arcana game system and its main use cases. Below diagrams will display each key functional areas involving the actors and use-cases within them.

2. System Overview Diagram



The system overview diagram contains the summary use-cases that supports our main Arcana application.

The summary use-cases are

1. Account management for the user and user management
2. User plays Augmented Reality games
3. User location is used to create encounters nearby\
4. User connects to server

The following summary use-cases define our functional area for our application.

Section 3: System Glossary

3. Glossary Grid

Name	Aliases	Labels	Description
Monster			
Delete Account			
Update Account			
Create Account			
Logout			
View Account			
Login			
Account Management			
User			A person using the application
XP			Also known as experience points. This is a measure of character progress. A user collects these points to level up their characters.
Encounters			Encounters can be puzzles or fights with bosses or monsters. They are to be completed in order to collect experience points and progress in the game.
Firebase			Firebase is the cloud service we use to authenticate the user logging in and when a new user creates an account, it gets added in Firebase.
AR			Stands for augmented reality
DB			Stands for database
Steps			The number of steps the user has taken while out playing the game, accumulated using the device's pedometer.

Location The positioning of the user when playing the application. Determined through longitude and latitude

Role Type of classification of the user's character. Examples include: Healer, Archer, Magician, Warrior, etc.

Section 4: Business Requirements

4. Business Requirements Business Rule Grid

ID	Name	Type Rule
BR001	Business Rule 1	As a user, I want to see creatures in augmented reality so I can battle them
BR002	Business Rule 2	As a user, I want my progress to be tracked so my character can grow and advance
BR003	Business Rule 3	As a user, I want to be able to switch between screen views of the map, and character details
BR004	Business Rule 4	As a user, I want my character role to be selectable
BR005	Business Rule 5	As a business owner, I want my business to attract players by making my location a hot place
BR006	Business Rule 6	As a user, I want to be able to login securely in order to access the application.
BR007	Business Rule 7	As a user, I want to create account to be able to play the game.
BR008	Business Rule 8	As a user, I want to walk around and play the game by looking for encounters
BR009	Business Rule 9	As a user, I want to be able to play with another user

Section 5: User Requirements Summary

User Requirements - Table for Caitlin

Use Case	Author	Description
Create Account	Caitlin	This use case is for new users to sign up through Firebase so that they can then log in using this account information in order to access the app.
Delete Account	Caitlin	This will permanently get rid of the users account in Firebase and they wont be able to log back in with it.

Account Management Summary	Caitlin	This functional area is for account management, and includes use cases such as login, create account, which also include view profile, edit profile, and so on.
Firebase Authentication	Caitlin	Users are able to authenticate credentials to access firebase
Firestore DB	Caitlin	Users are able to get and put data from Firestore DB
Invalid Credentials	Caitlin	Users information is validated before being allowed to log in
Logout	Caitlin	Users are able to logout of user accounts
Login	Caitlin	Users are able to login of user accounts
Reset Password	Caitlin	Users are able to reset password
Update Account	Caitlin	Users are able to update information on account
View Account	Caitlin	Users are able to view details on account

User Requirements - Table for Jae Won

Use Case	Author	Description
Character walking around is displayed as player walks around	Jae	The user should walk the device in hand and the UI should reflect that. The user location should be provided by Google cloud services using personal device GPS and update their location to cloud services.
Manage Character	Jae	This is the summary use case for managing character. The functionality is to get user information, character profile and manage the character through equipments, raising levels and on.
Fighting the Encounter	Jae	This use case is the summary for fighting the encounter as a user. The user encounters a monster or an event. The user performs a series of tasks or moves to complete the encounter. The user is rewarded items and experience points.
Playing the game	Jae	This is the summary for playing the game use case. The functionality of this use case is the user will have encounters with their characters. The user will then have battles with monster or bosses in AR.
Sees the character rendered	Jae	Rendering function is done locally within device. The update is constantly called to reflect user location and presented on map. The logic required is in View while the data required to process this is sent from the controller to view.
Walks with device in hand	Jae	The user should walk the device in hand and the UI should reflect that. The user location should be provided by Google cloud services using personal device GPS and update their location to cloud services.

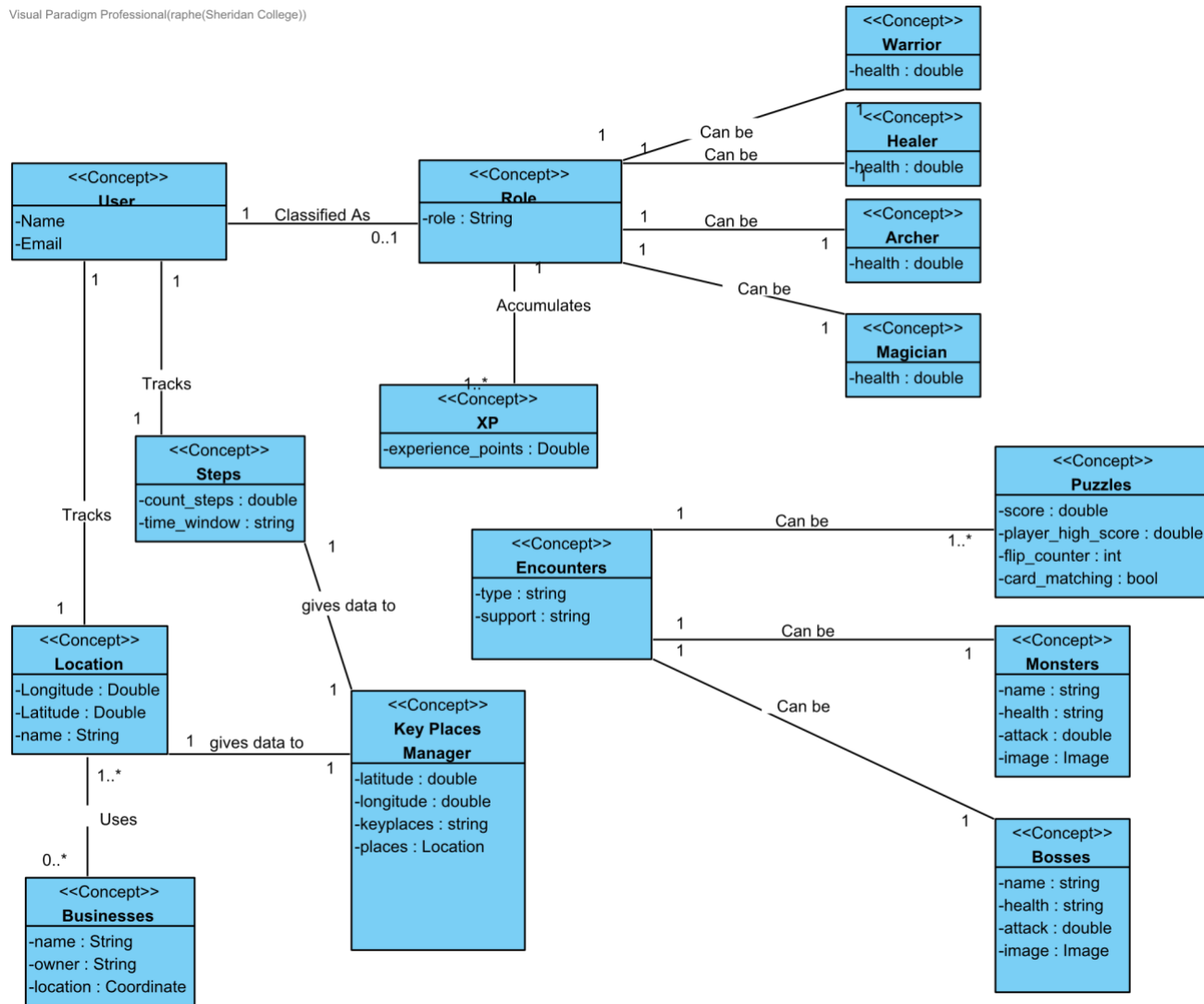
User Requirements - Table for Suzi

Use Case	Author	Description
Encounter Monster	Suzi	Monster encounter will be started when user is at the key area. It will contain monster's status, therefore, user can decided on his level, either he wants to battle with monster or Run away from monster.
Fight Monster	Suzi	Battle between Monster and the User will start, User and monster can attack eachother by using basic attack or skills. One of them lose all the health will lose the battle. It will send result to API Handler
Find the Key Spot	Suzi	User needs to find the key spot from location view to encounter with the monster.
View results of encounter	Suzi	Get result and all the items and xp that user earned. It displays the result.
Provide Result to the player	Suzi	API Handler will send the result data to data base. API Handler will get root items or xp data that player earned. Sending them to result display.
Run Away	Suzi	User decide to run away from the battle with the monster. User will not able to root any items or XP. Result will be send to API handler

Section 6: Domain Overview

5. Domain Model Class Diagram

Visual Paradigm Professional(raphe(Sheridan College))

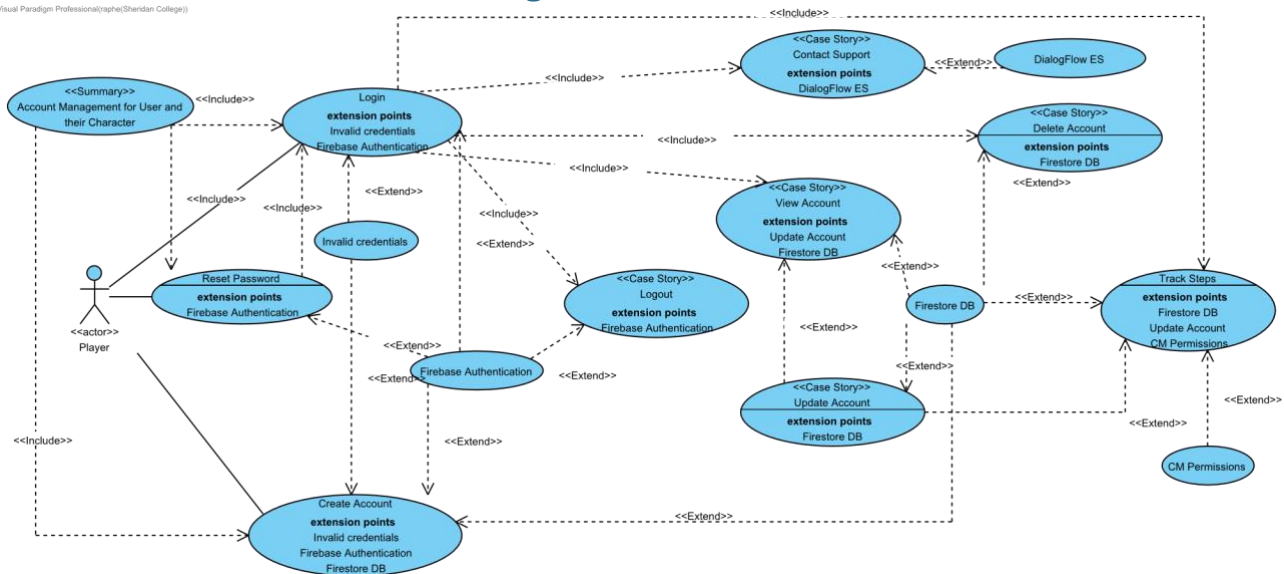


Domain overview diagram shows key classes involved with project Arcana. Key classes all branch from User (Concept). Users have name and email, and have role class involved with them, which represents the user's character. With it, core use cases such as the game involved or locations involved are classes linked with the user class.

Section 7: Use Cases

6. Functional Area 1 Caitlin Use Case Diagram

Visual Paradigm Professionals(raphie@Sheridan College)



This functional area is all about account management. The use cases implemented were login, logout, create account, reset password, view profile, edit profile, contact support via a chatbot, and tracking steps.

A new player creates account

A player logs in to Arcana application. From here a player can logout, switch authentication, and change password.

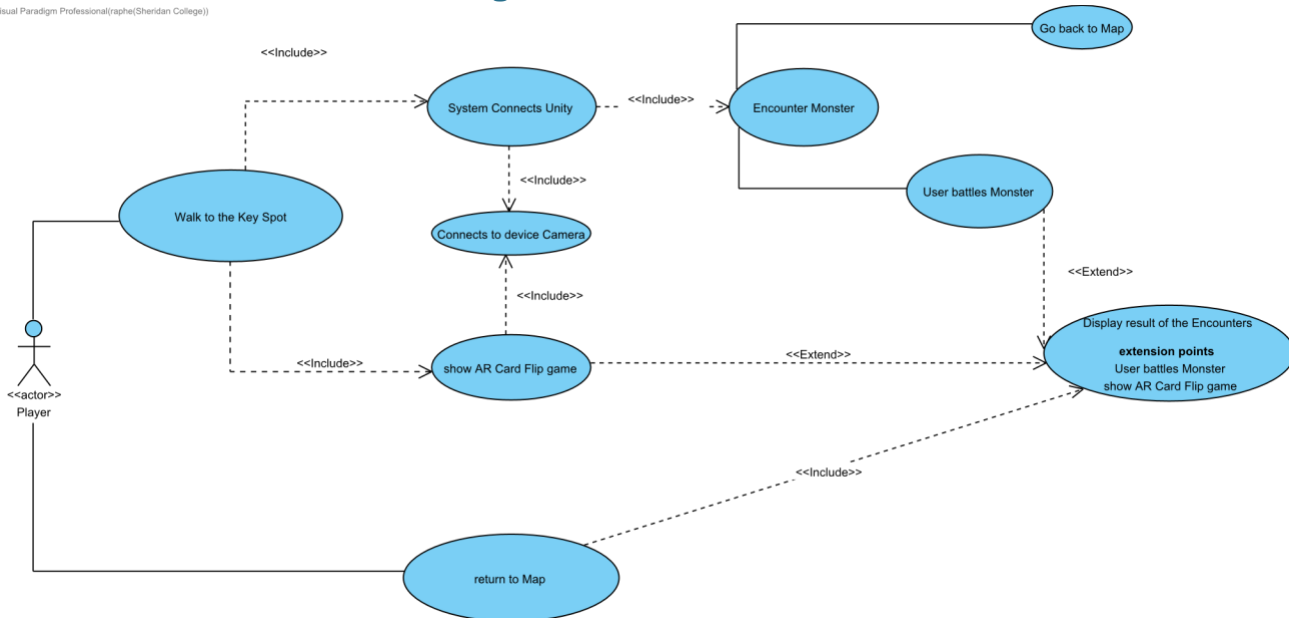
A player chooses role of a class for the players's character such as archer, healer, etc.

Then a player can choose equipments to wear for the character. The player walks around with device in hand, where the application will track how many steps the player has taken through pedometer features. Depending on numbers of steps taken or upon meeting encounter, the player sees mini games or boss fights displayed in augmented reality. The player is able to interact and enjoy the game with other players or play the game alone.

Upon clearing encounters, players are rewarded with items and experience to level up and improve their character.

7. Functional Area 2 Suzi Use Case Diagram

Visual Paradigm Professional(raphel@Sheridan College)



Our next functional area is about encounters for the game feature of our app which will happen when a user finds the key spot by walking. Use cases are playing the battle part of the game, and the mini game such as flipping cards to find matching pairs. After the battle, or when the card game is finished, the user will see the result of the encounter.

From the map view, the user will select one of the two buttons while being at an encounter.

The user selects AR battle button, which will make the system load our unity project.

The user sees the surroundings around the user through camera displayed on device. The user sees Augmented Reality character and monster, and sees interactable buttons to press.

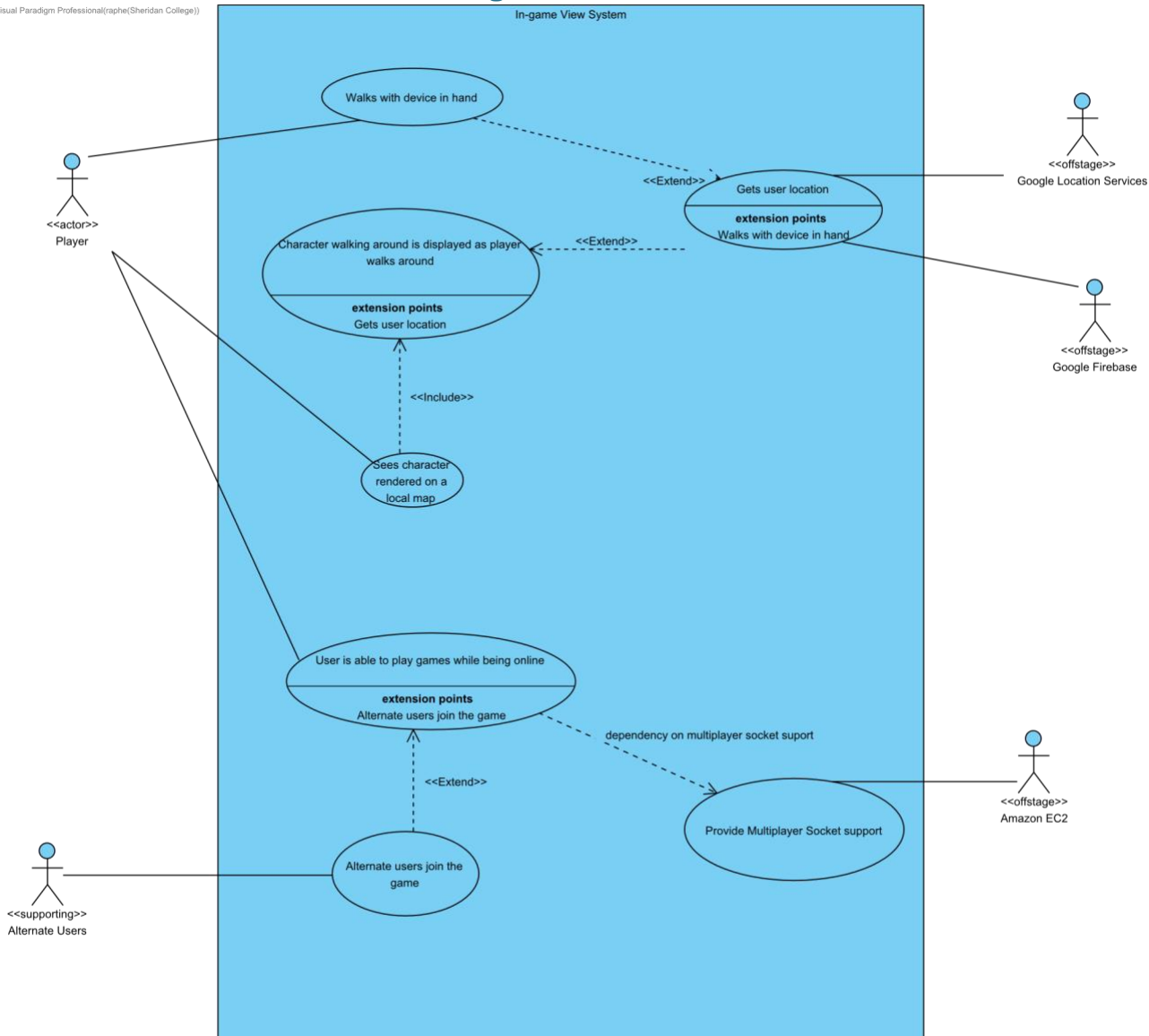
The user battles the monster.

The user sees result screen of the battle once the battle is completed.

In the other case, the user can press Augmented Reality Puzzle button. The user can

8. Functional Area 3 Jae Won Use Case Diagram

Visual Paradigm Professional(raphie(Sheridan College))



Our third functional area covers core functional area related with Arcana.

Our key use case with the user is that the user walks with device in hand.

The user walks with device in hand.

The user's location is updated live on screen.

The user's character is presented by a image sprite.

The user is able to head to encounters.

The user can start battles in Augmented Reality or puzzles in Augmented Reality at those encounters.

The user is also able to join multiplayer chat room

The system will create a connection bridge between user device and the mobile device.

The user is able to type messages to other users in the same event session

Other user's messages are retrieved and displayed to the user in the same event session.

Appendix C

Software Design Document

Capstone Project Arcana

Software Design Document

INFO 45828
CS Capstone Project

Authors
Jae Won Park
Caitlin Rush
Suzi Ryu

Professor Richardo Paharsingh

Table of Contents

1 System Architecture.....4

2 Structural Overview - Create Account Use Case.....8

3 Structural Overview - Login Use Case.....8

4 Structural Overview - Edit Profile Use Case.....9

5 Structural Overview - Reset Password Use Case.....9

6 Structural Overview - View Profile Use Case.....10

7 Structural Overview - Contact Support Use Case.....11

8 Structural Overview - user plays AR battle.....12

9 Structural Overview - User plays game.....12

10 Structural Overview - Create Account Use Case.....13

11 Behavioural Overview - Create Account Use Case..... 14

12 Structural Overview - Login Use Case.....14

13 Behavioural Overview - Login Use Case..... 15

14 Structural Overview - Edit Profile Use Case..... 15

15 Behavioural Overview - Edit Profile Use Case.....16

16 Structural Overview - Reset Password Use Case.....16

17 Behavioural Overview - Reset Password Use Case.....17

18 Structural Overview - View Profile Use Case.....18

19 Behavioural Overview - View Profile Use Case.....19

20 Structural Overview - Contact Support Use Case.....19

21 Behavioural Overview - Contact Support Use Case..... 20

22 Structural Overview - User meets encounters.....20

23 Structural Overview - user plays AR battle..... 21

24 Behavioural Overview - User meets encounters.....22

25 Collaboration Overview - Users meets encounters.....22

26 Behavioural Overview - AR encounters..... 23

27 Collaboration Overview - AR Encounters.....23

28 Structural Overview - AR Mini game.....24

29 User Plays Flip Card mini game Sequence Diagram.....24

30 Collaboration Overview - AR Encounters2.....25

31 Structural Overview - System interacts with Cloud..... 26

32 Behavioural Overview - System interacts with Cloud.....27 33

 Collaboration Overview - Cloud Connections.....27

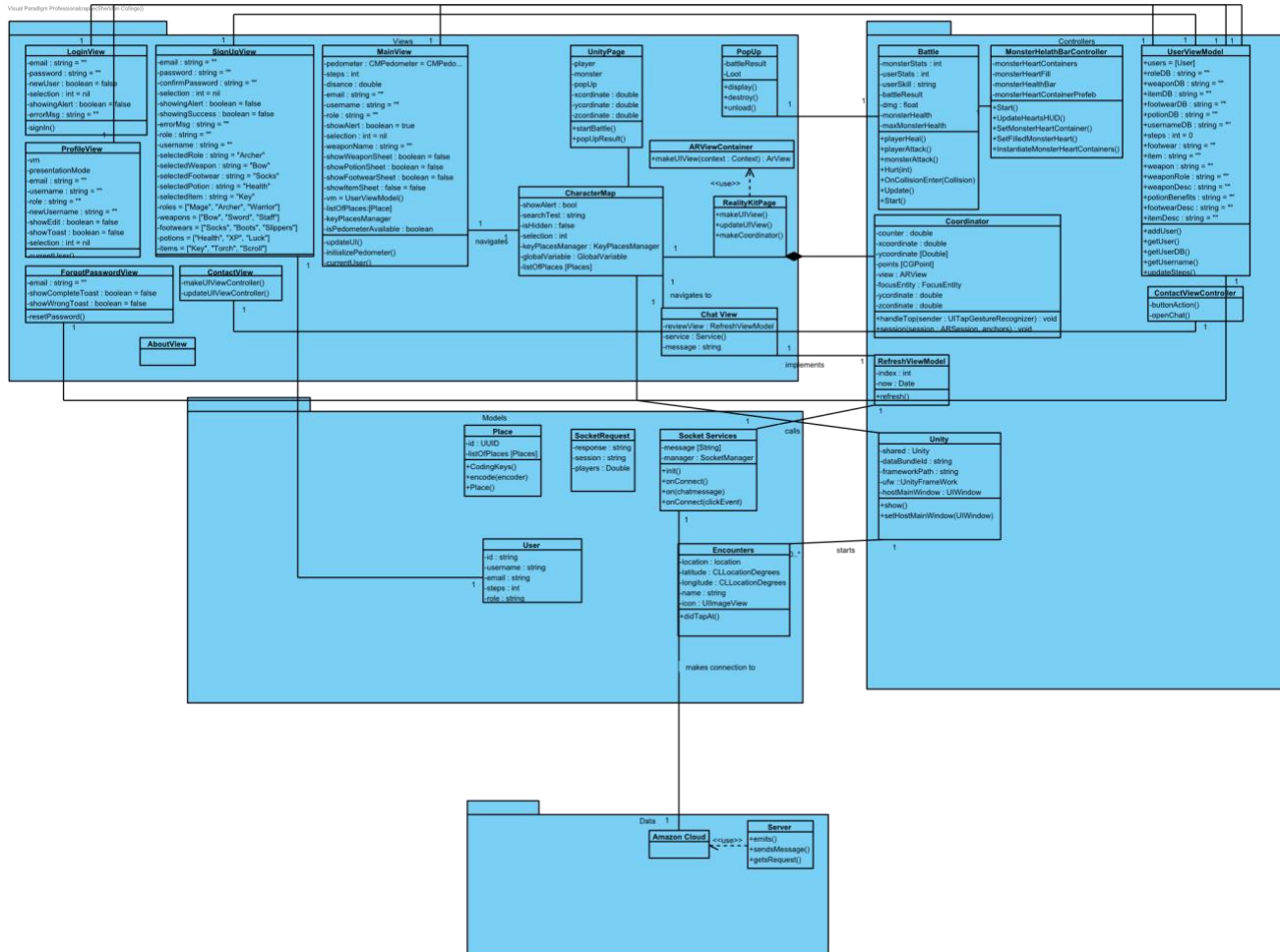
1. Introduction
2. Software Architecture
3. Detailed Design
4. Interaction Model

Section 1: Introduction

This software design document will demonstrate the software architecture of Arcana. It will display the overall software architecture by displaying various diagrams related with it. It will also dive into the different layers and subsystems of Arcana architecture.

Section 2: Software Architecture

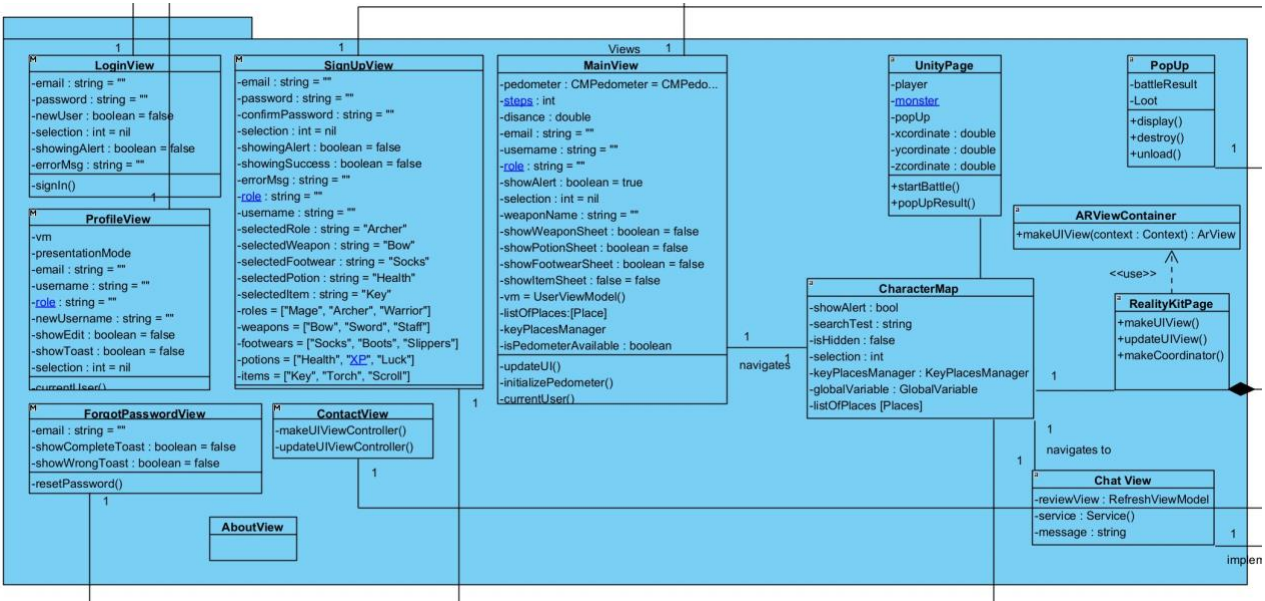
1. System Architecture



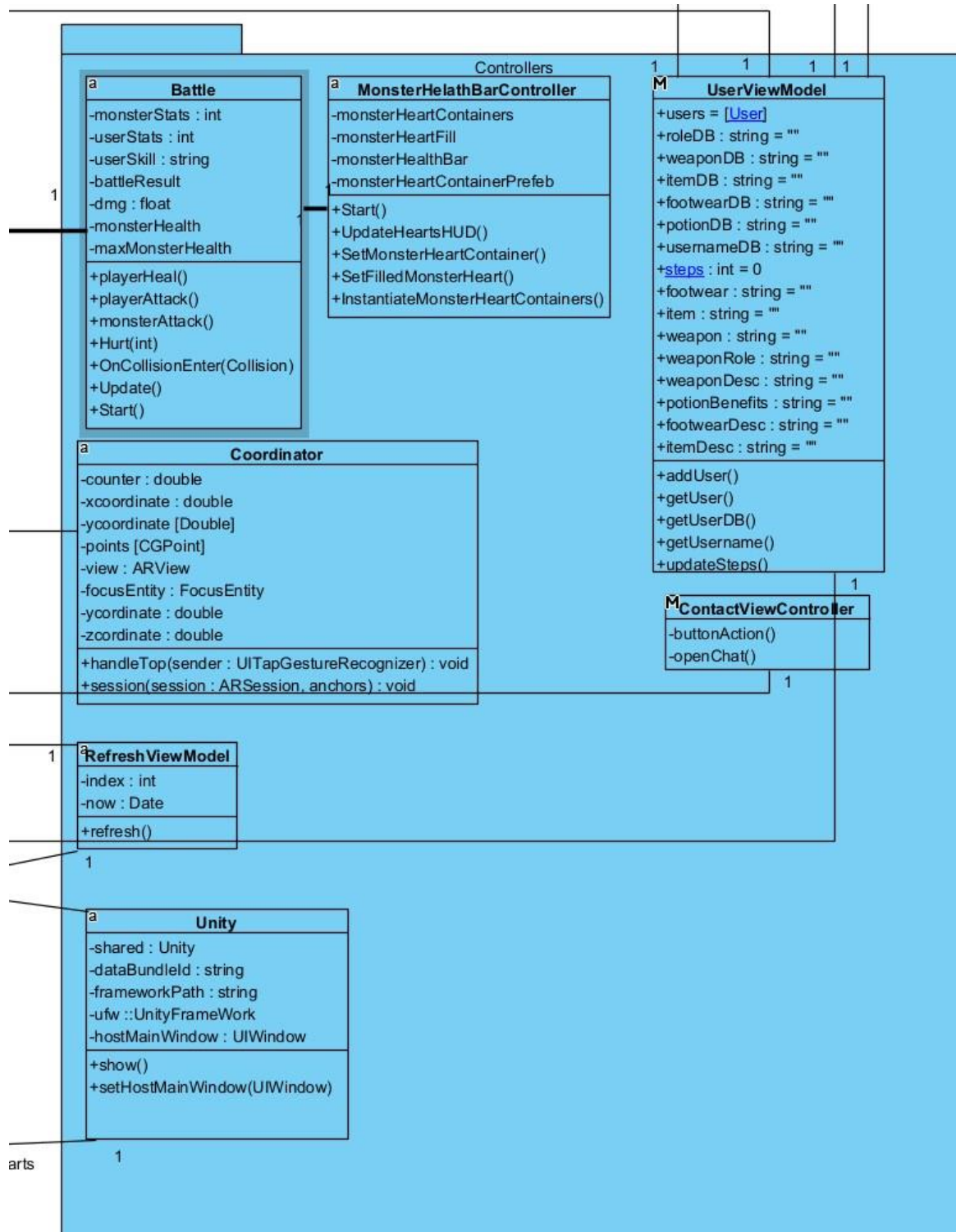
Arcana has a Model View Controller (MVC architecture). At our front and uppermost layer is the view layer. As Arcana project is an application built on SwiftUI on iOS, the view layer contains many of the core components. It is supported by application logic layer (Controller) where many of the services needed for View layer is handled here. Under application logic layer, many of the use cases involving the games, encounters and location services are handed here. Model layer is the third layer that contains all the core classes and model classes. Most of the user information and role are defined here.

We have our View layer in the top most layer, which contains most of our views for our application. As we decided on SwiftUI as to be our main programming language for our Native iOS application, we wanted to choose a software architecture that can separate our view from the rest of our logic and program. The View layer communicates with our Control layer. This communicates with our Model layer, which communicates with multiple supported libraries, APIs and databases, which sends the data back to our

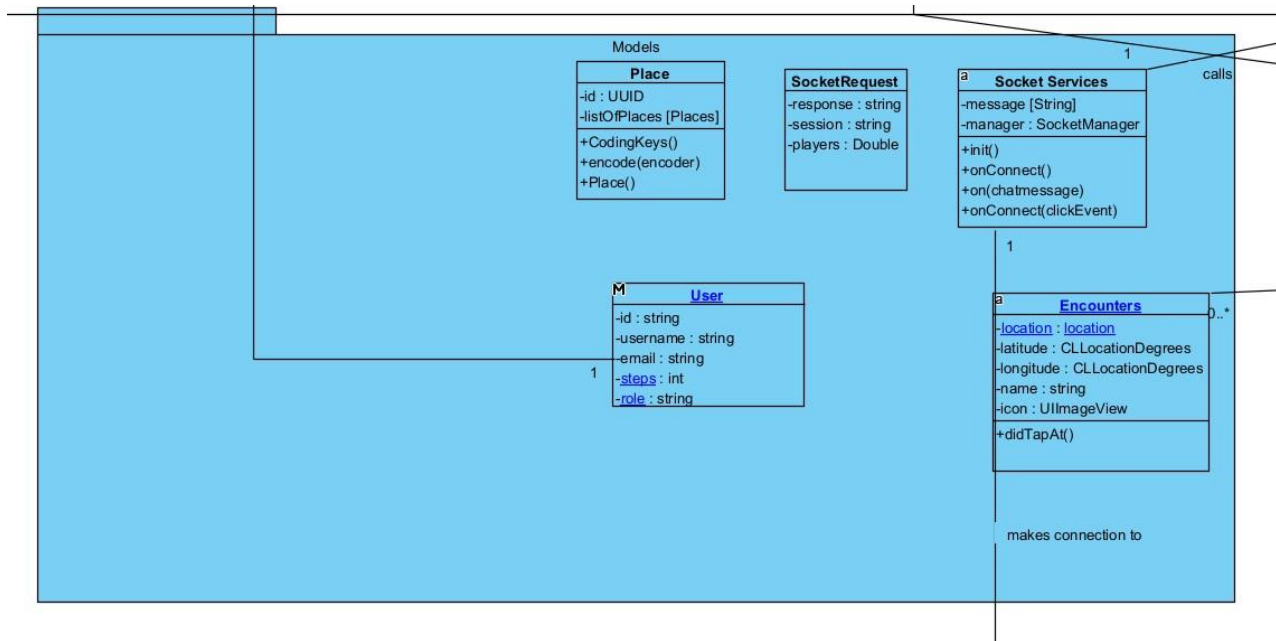
Controller layer to be sent to be used in the View layer. In the next slide, I will go more in-depth of how our application works through this MVC architecture.



Our view contains information on all the classes within the View. In SwiftUI, they are the view swift files. These classes contain view itself and view logic.



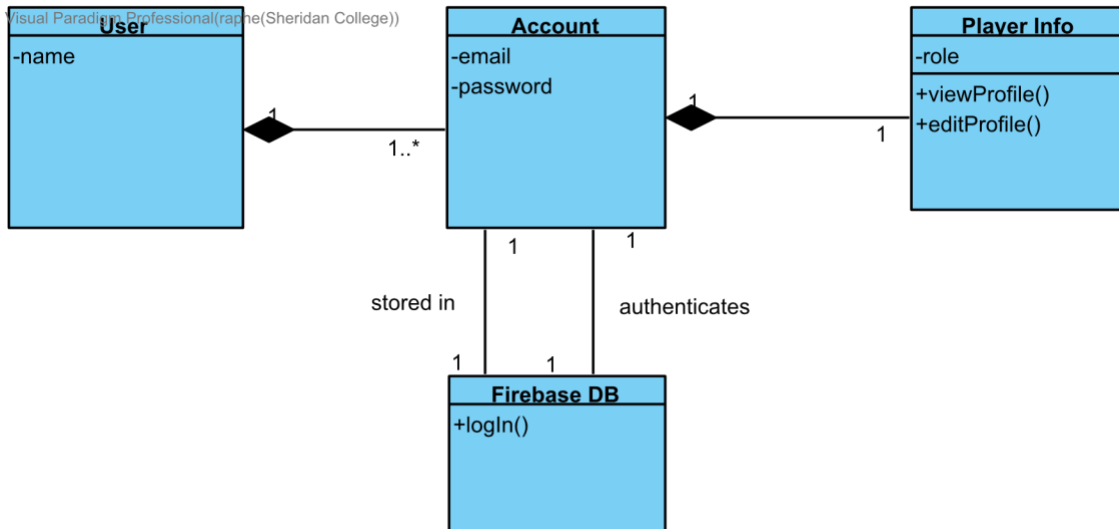
The Controller contains all the necessary component to pass data between model and the view. For example, UserView model handles all the data received from the User Model, which calls from the Google Cloud. UserView sends the necessary information to MainView to display all the correct information about user's account, user's character and on.



The model view architecture contains model data. It contains classes that define our main models that cover model usecases.

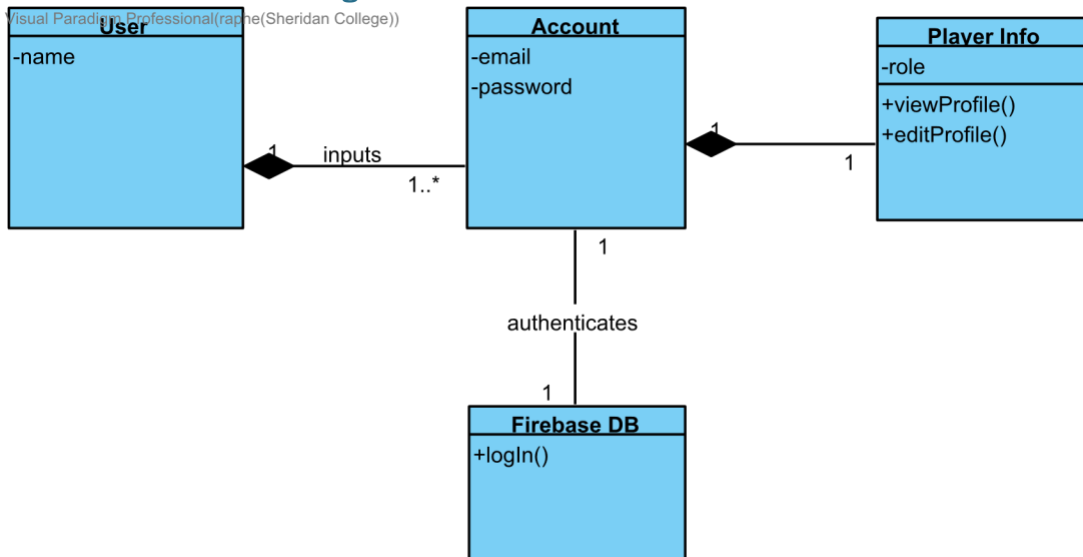
Section 3: Detailed Design

2. Structural Overview - Create Account Use Case



The design of creating account involves the application layer because when a user creates a new account, it sends an update to the Firestore database to add a new account with the inputted information as a new document in the user collection.

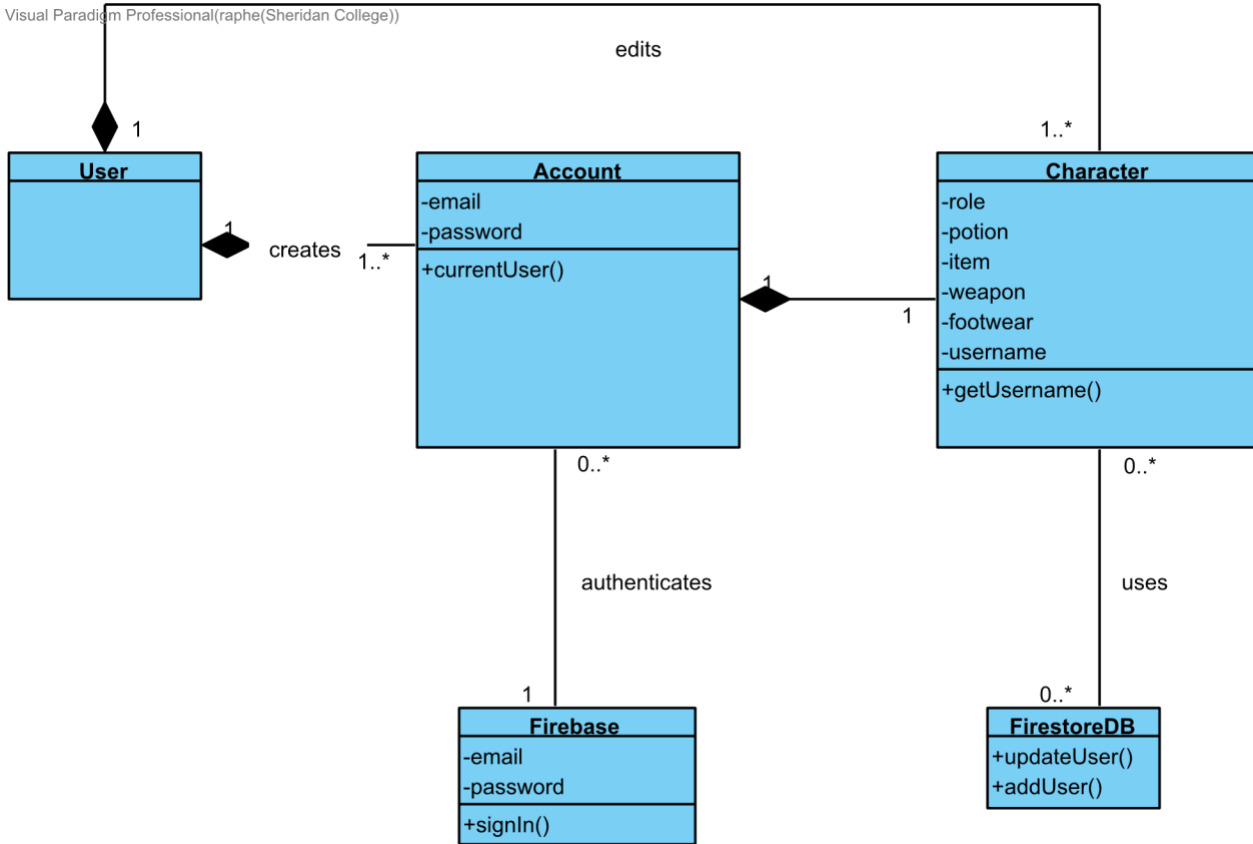
3. Structural Overview - Login Use Case



Logging in involves the application layer to authenticate the current user within Firebase.

4. Structural Overview - Edit Profile Use Case

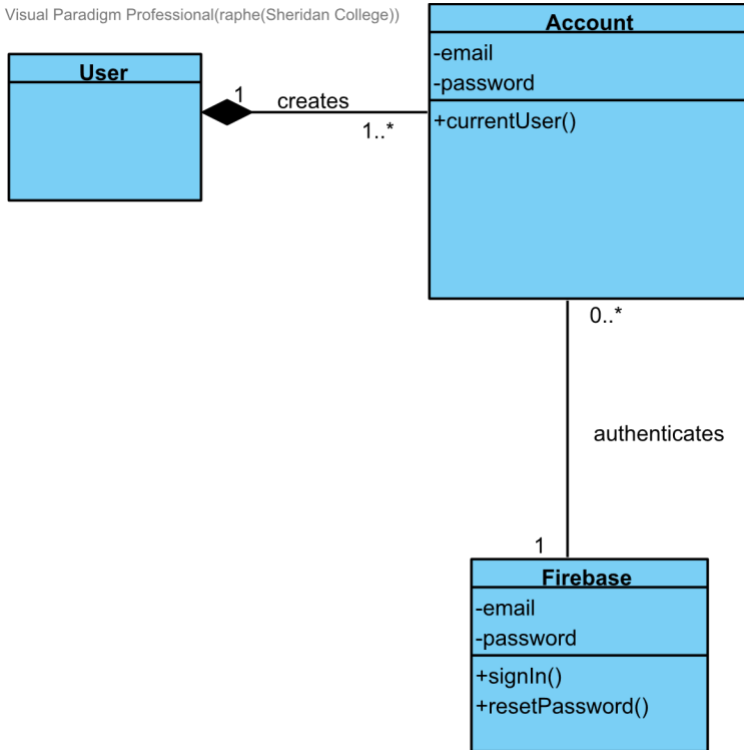
Visual Paradigm Professional(rapher(Sheridan College))



The design for Edit Profile uses application layer to first fetch the information from Firestore for the current user, then once the user has entered their new information, it sends another call to update that current document in the user collection in Firestore.

5. Structural Overview - Reset Password Use Case

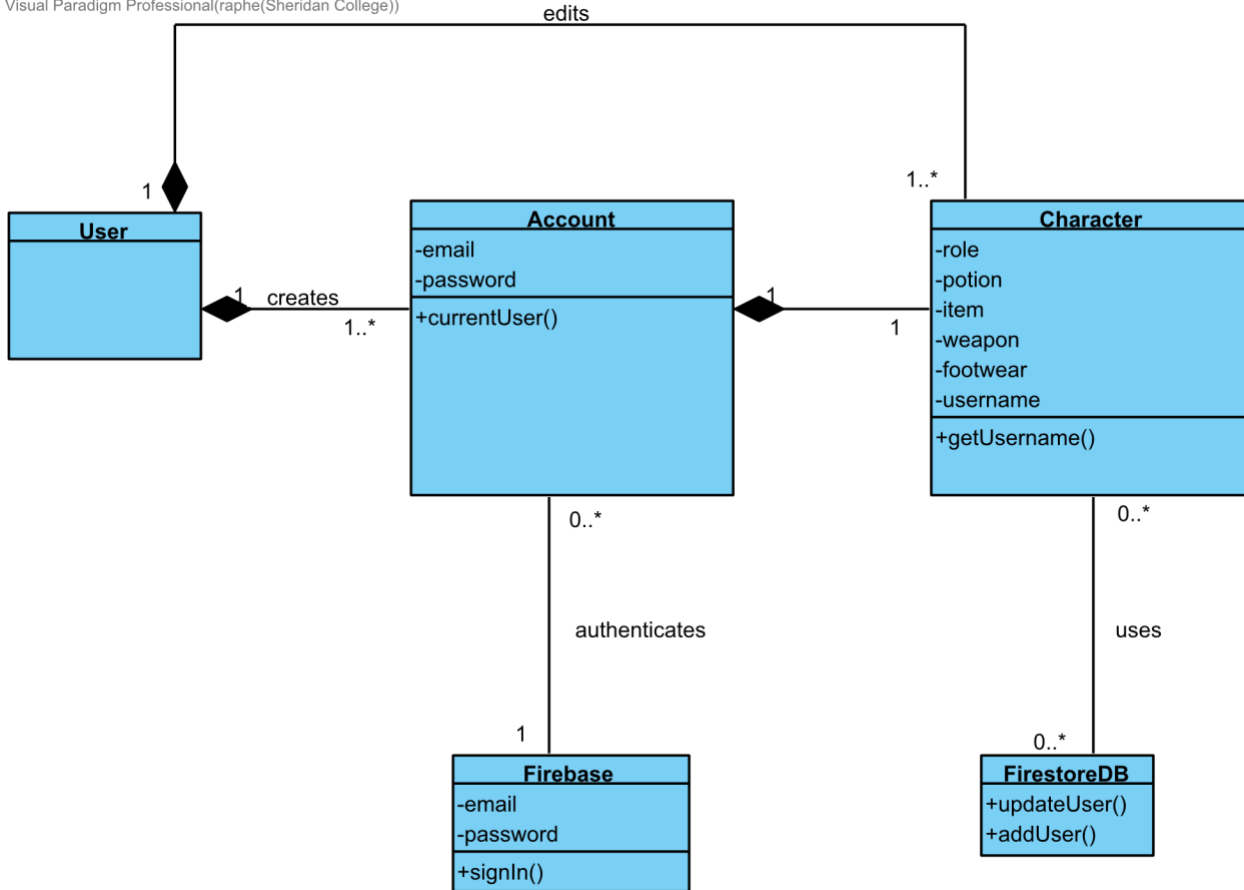
Visual Paradigm Professional(raphel(Sheridan College))



Reset password uses the application layer to connect to Firebase for the current user and sends them a reset password email for the matching email address.

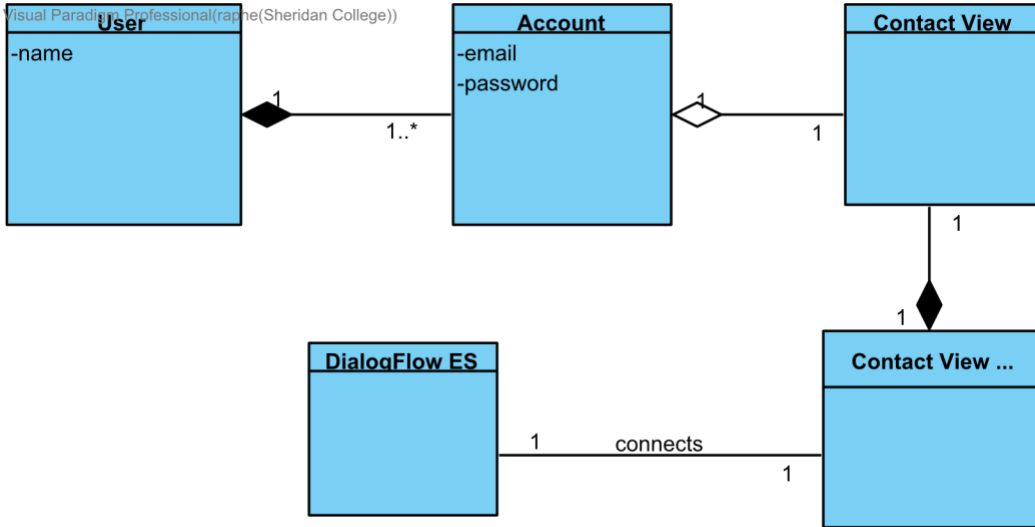
6. Structural Overview - View Profile Use Case

Visual Paradigm Professional(raphe(Sheridan College))



This design for View Profile involves the application layer. The profile information is fetched from the document matching the user within the database in Firestore and presented in the view.

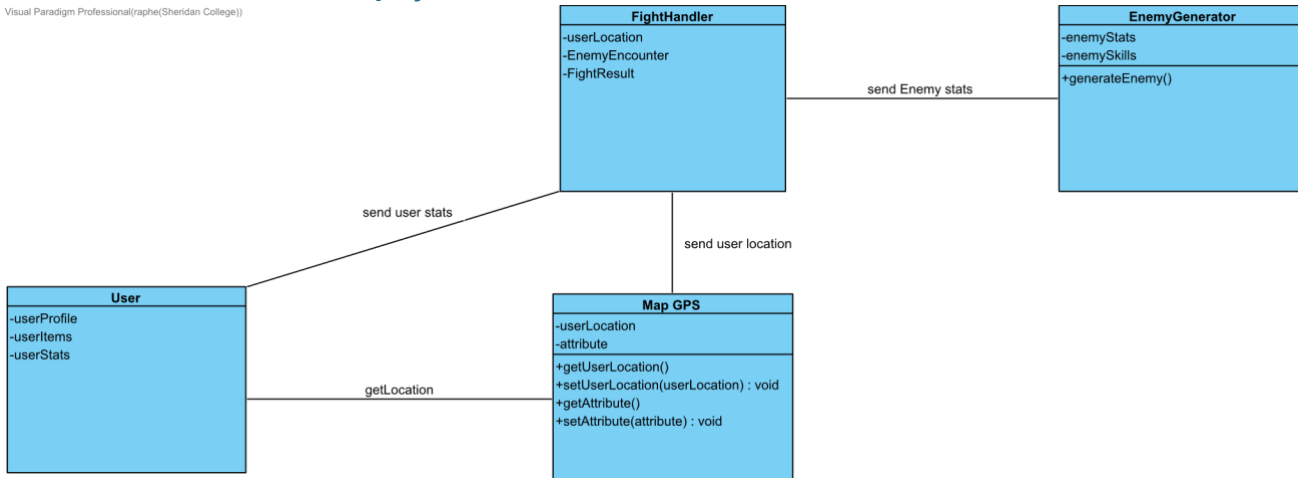
7. Structural Overview - Contact Support Use Case



This detailed design falls under application logic. All the cloud connections are done in application logic layer. The back-end calls are sent to the cloud connections, Google Cloud Services and Amazon Cloud Services

8. Structural Overview - user plays AR battle

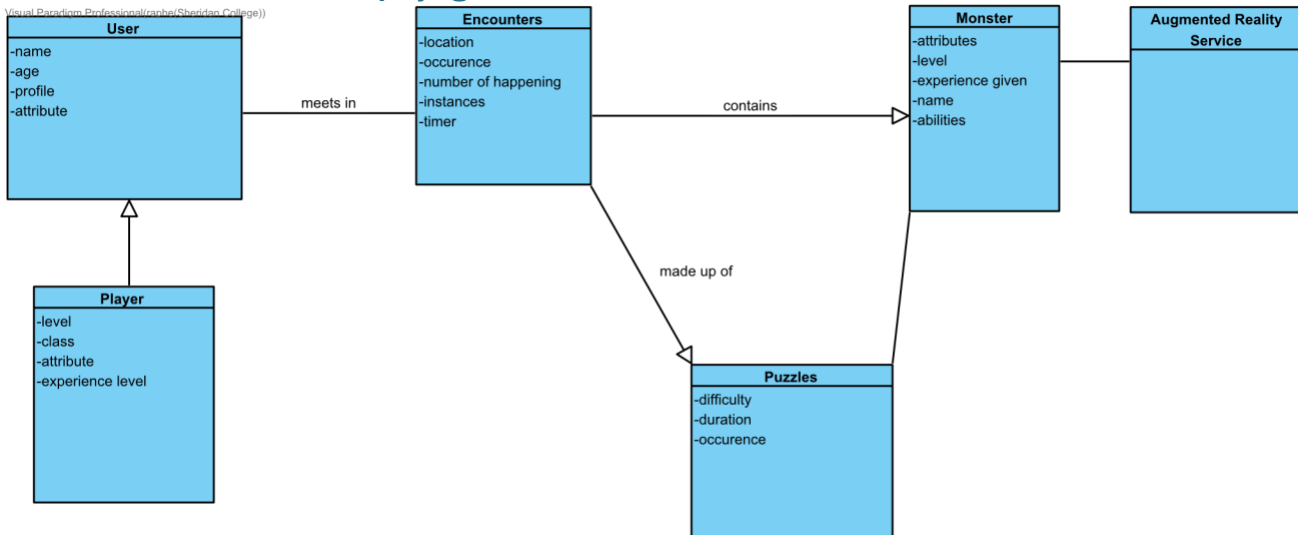
Visual Paradigm Professional(raphel@Sheridan College)



The following detailed design falls under Application Logic. AR Battle is part of view controller layer. User sends data to FightHandler, which interacts with EnemyGenerator class to render AR battle.

9. Structural Overview - User plays game

Visual Paradigm Professional(raphel@Sheridan College)

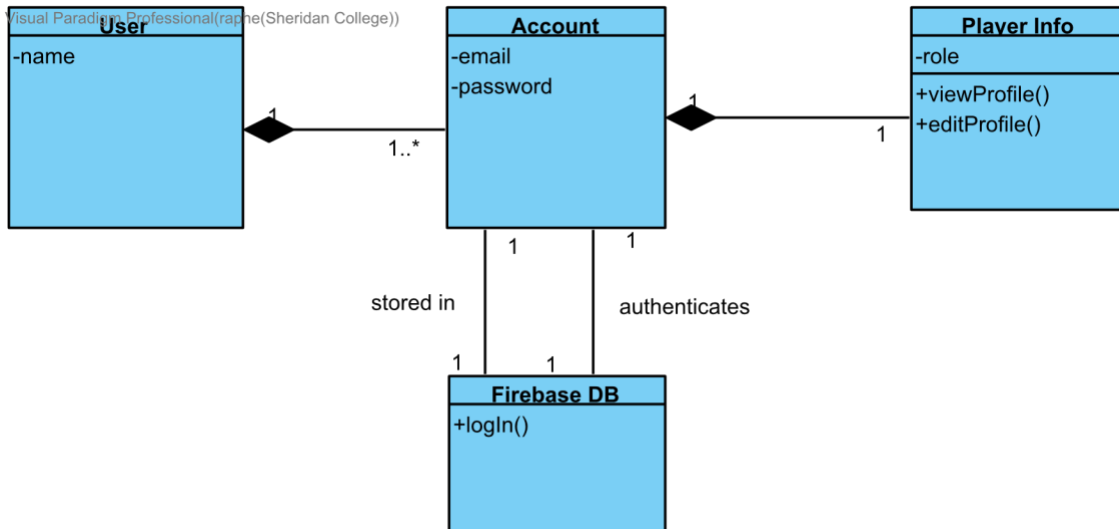


The following detailed design also falls under application logic. The user controlled by the player, interacts with the encounter to either battle a monster or solve a puzzle. Battling a monster is supported by Augmented Reality library, Unity, while the puzzle can be supported by AR service.

Section 4: Interaction Design

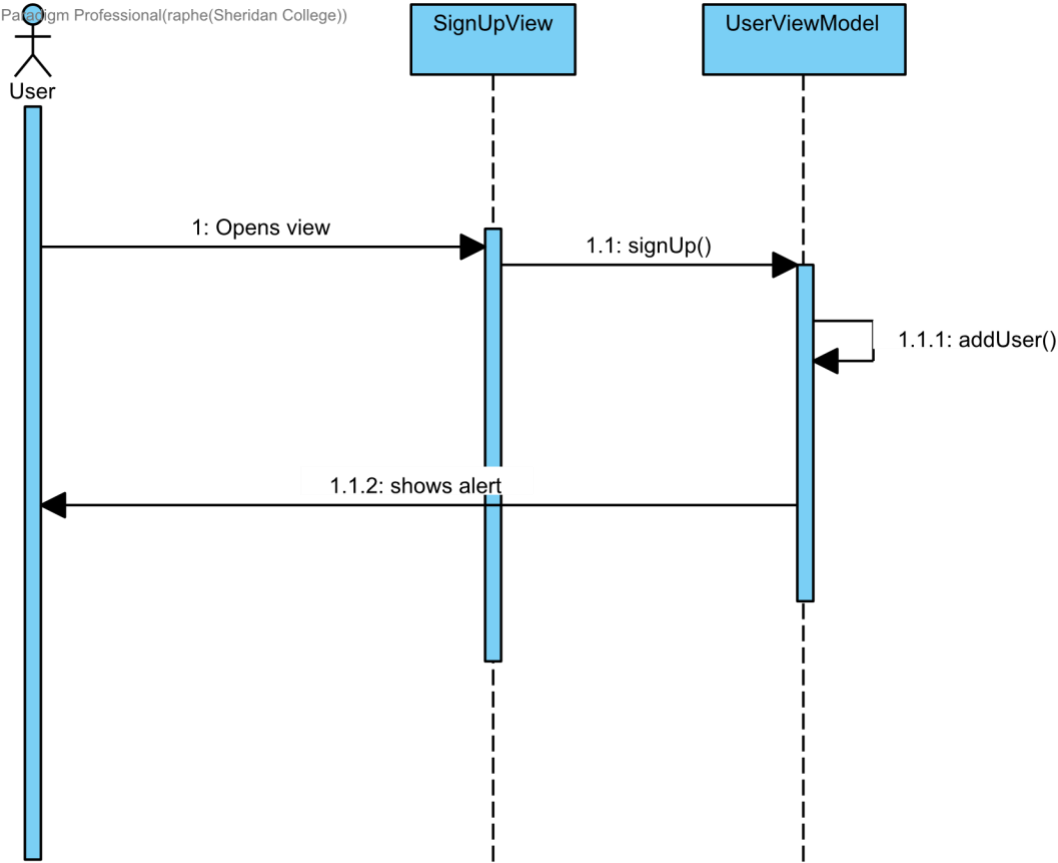
1. Create Account Use case
2. Login Use Case
3. Edit Profile Use Case
4. Reset Password Use Case
5. View Profile Use Case
6. Customer Support Use Case
7. System Interacts with Cloud
8. AR Encounters
9. AR Mini game Use Case
10. Cloud Connections

10. Structural Overview - Create Account Use Case



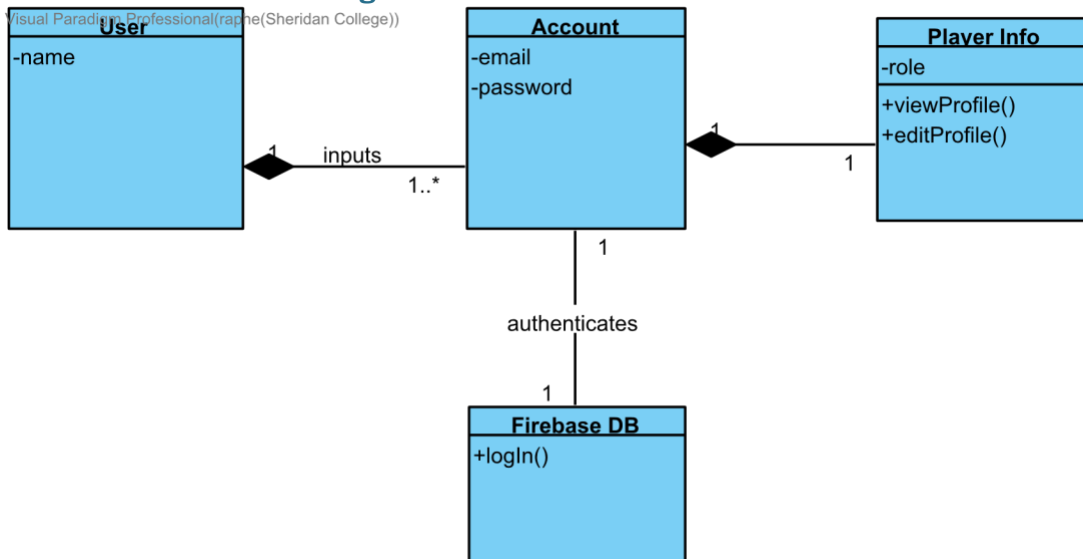
11. Behavioural Overview - Create Account Use Case

Visual Paradigm Professional (raphe(Sheridan College))



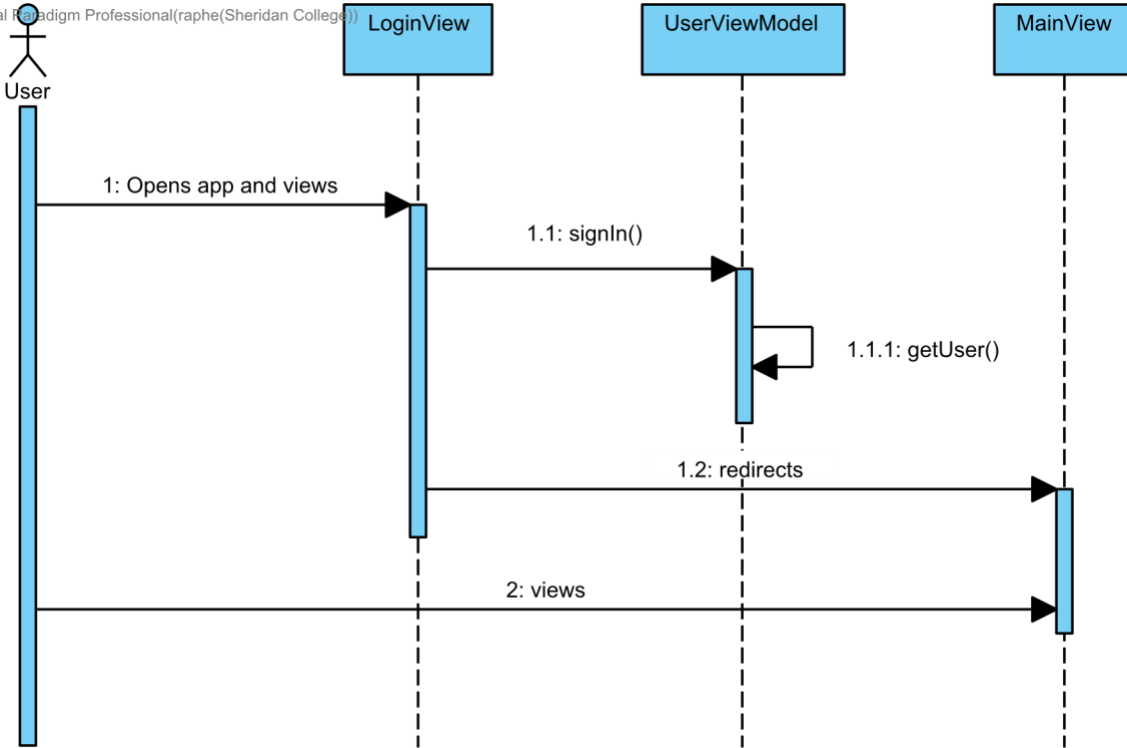
12. Structural Overview - Login Use Case

Visual Paradigm Professional (raphe(Sheridan College))



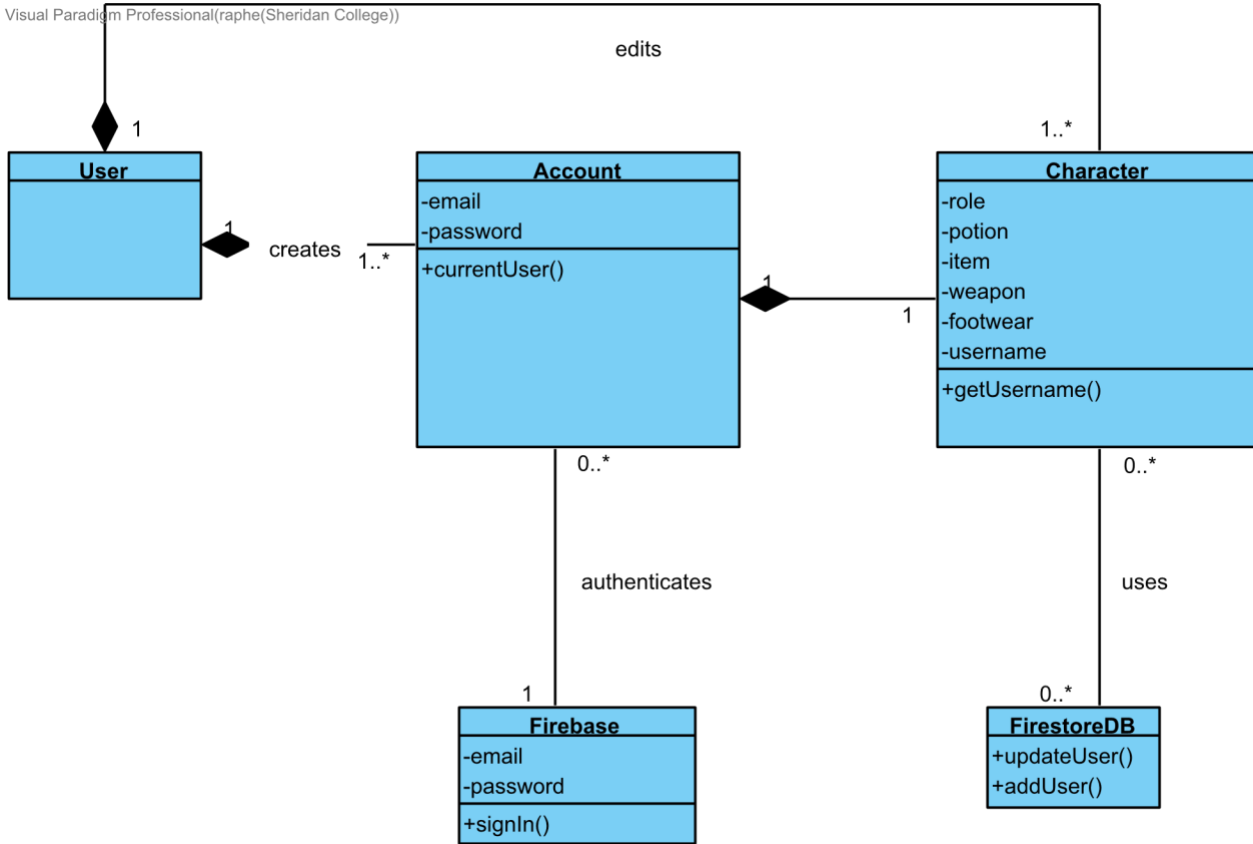
13. Behavioural Overview - Login Use Case

Visual Paradigm Professional(raphe(Sheridan College))



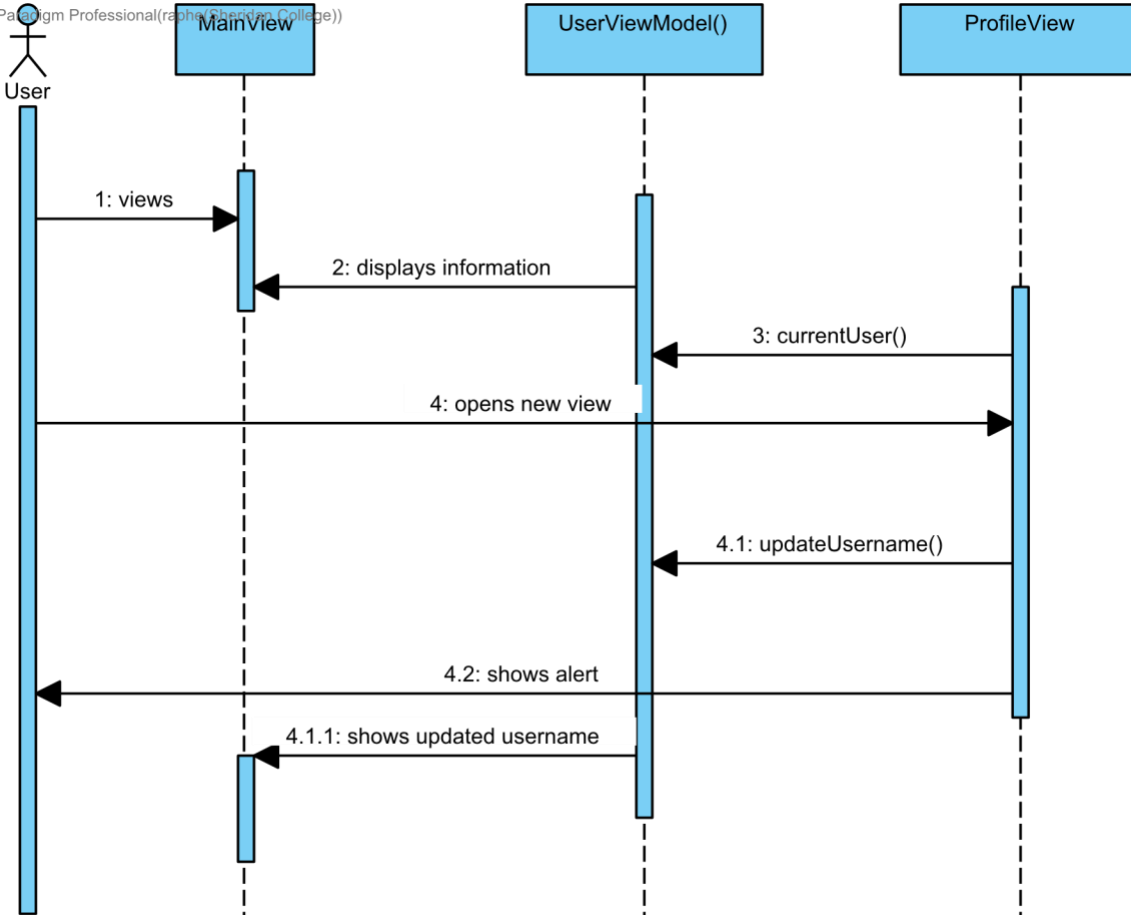
14. Structural Overview - Edit Profile Use Case

Visual Paradigm Professional(raphel(Sheridan College))



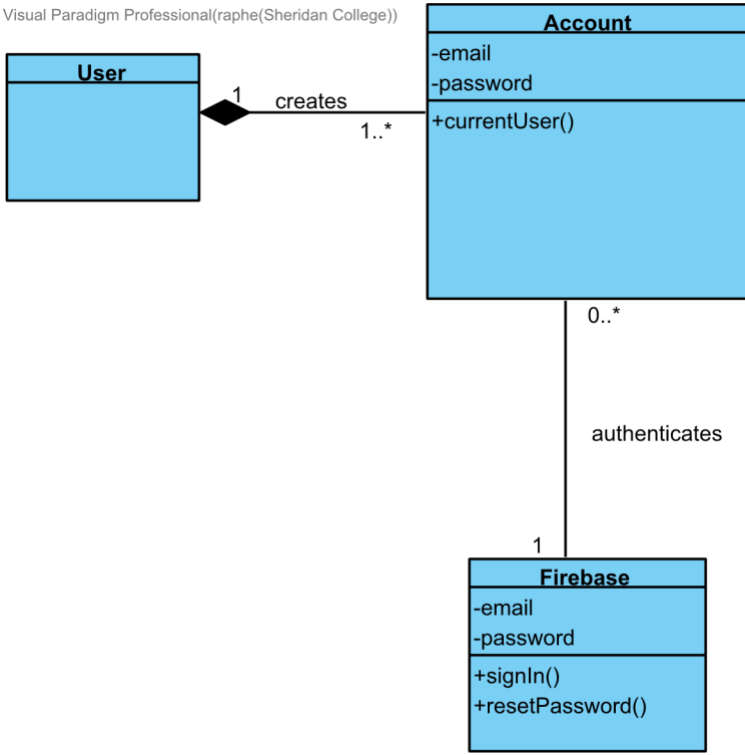
15. Behavioural Overview - Edit Profile Use Case

Visual Paradigm Professional (revised Sheridan College)



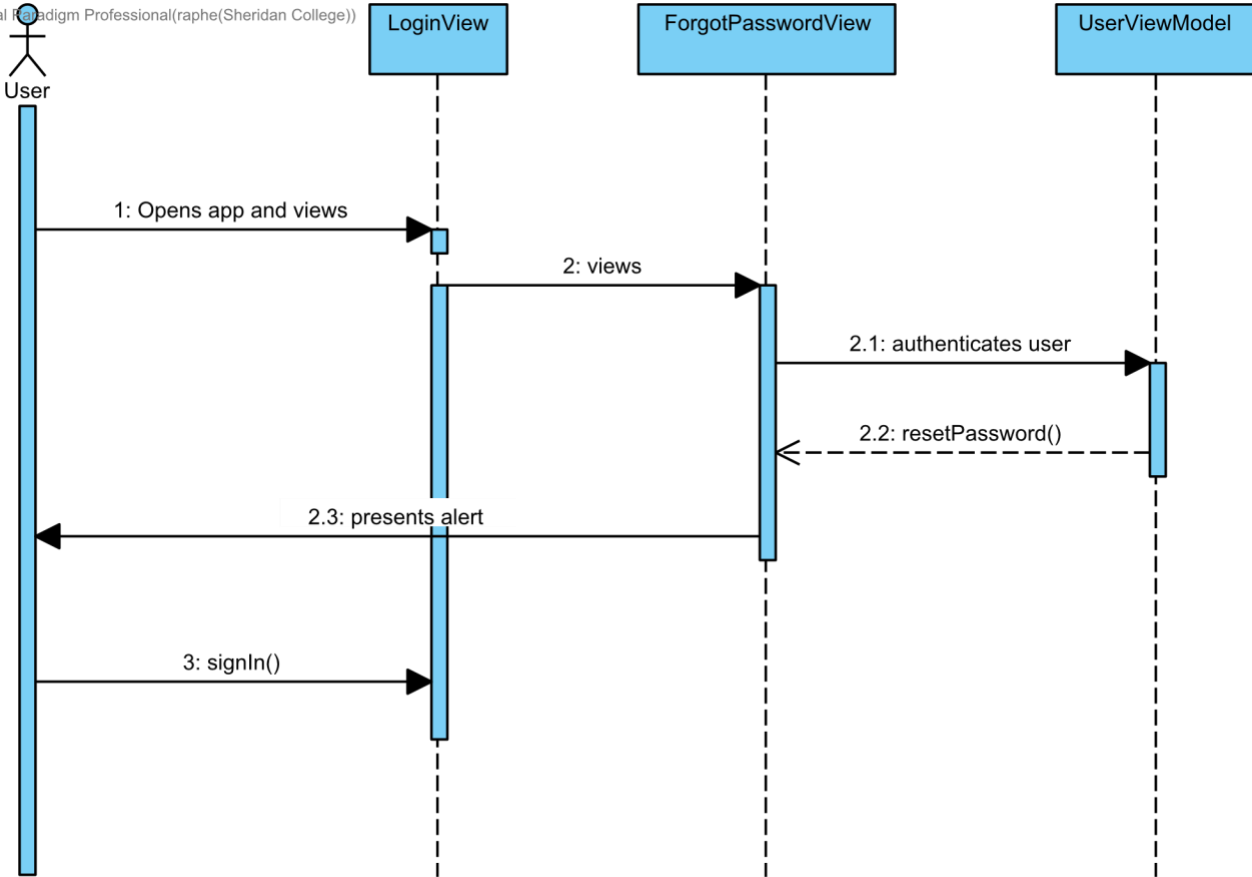
16. Structural Overview - Reset Password Use Case

Visual Paradigm Professional(raphel(Sheridan College))



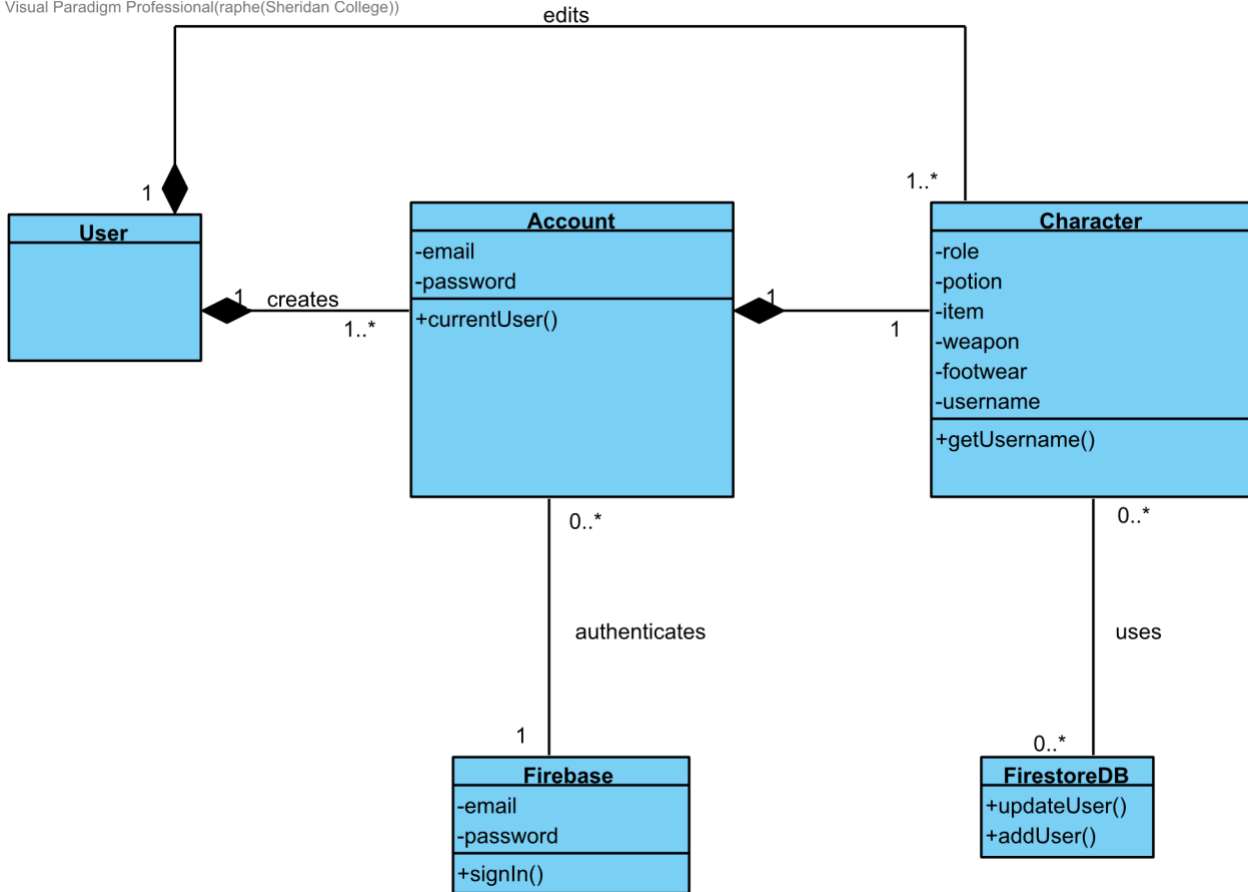
17. Behavioural Overview - Reset Password Use Case

Visual Paradigm Professional(raphel(Sheridan College))

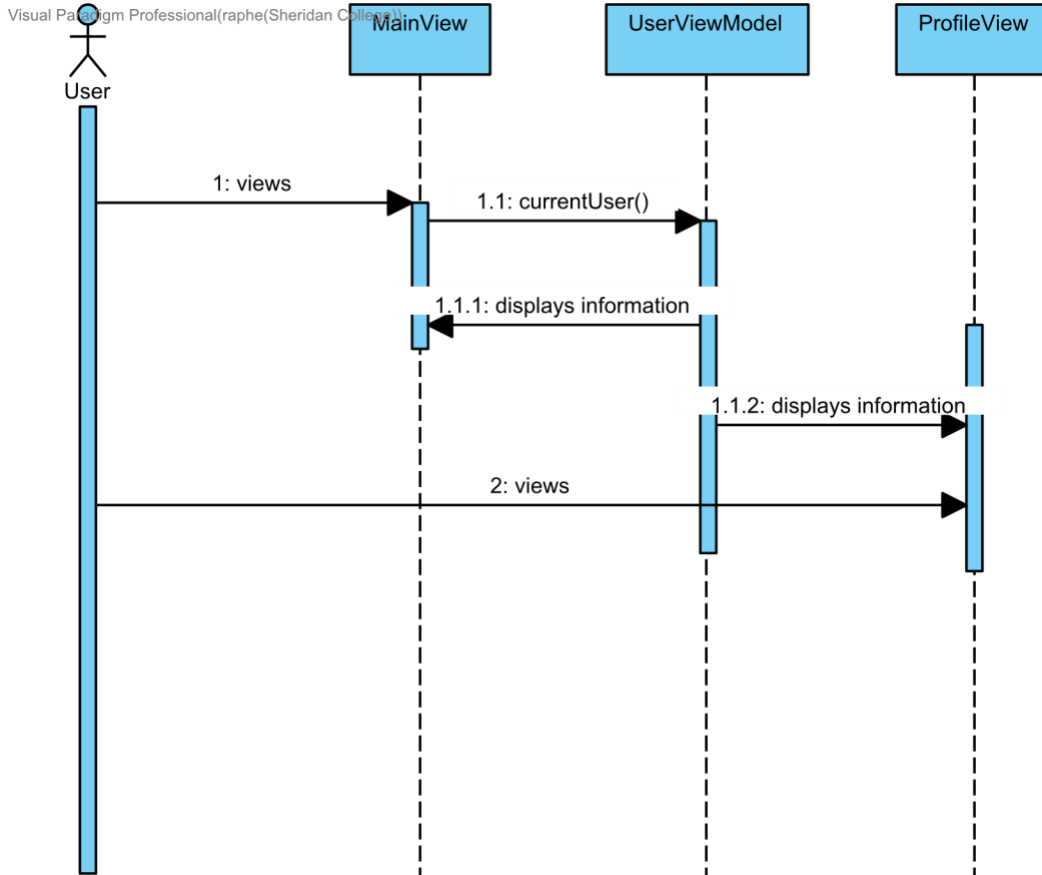


18. Structural Overview - View Profile Use Case

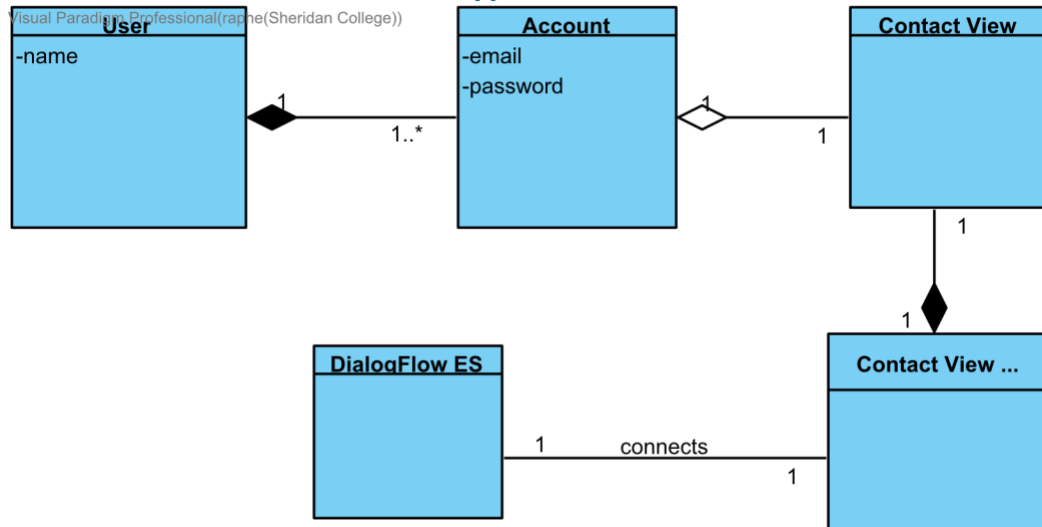
Visual Paradigm Professional(raphe(Sheridan College))



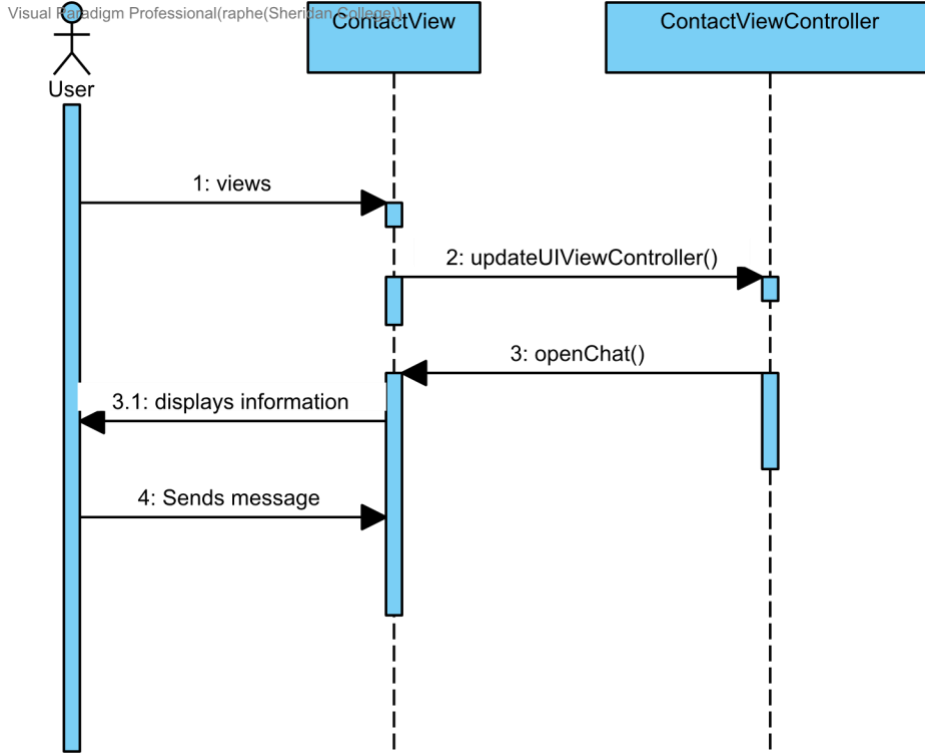
19. Behavioural Overview - View Profile Use Case



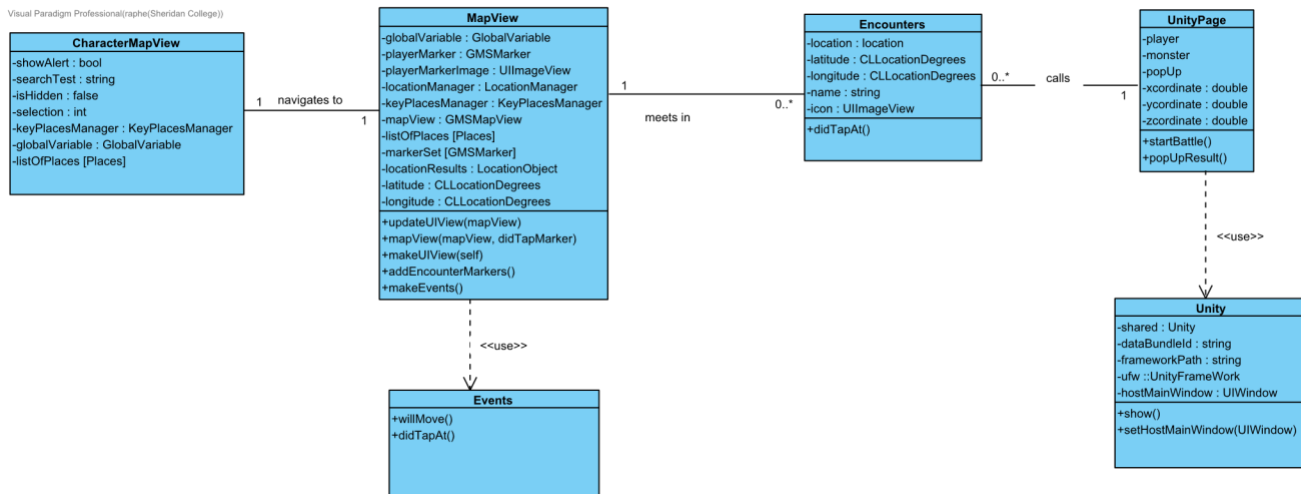
20. Structural Overview - Contact Support Use Case



21. Behavioural Overview - Contact Support Use Case

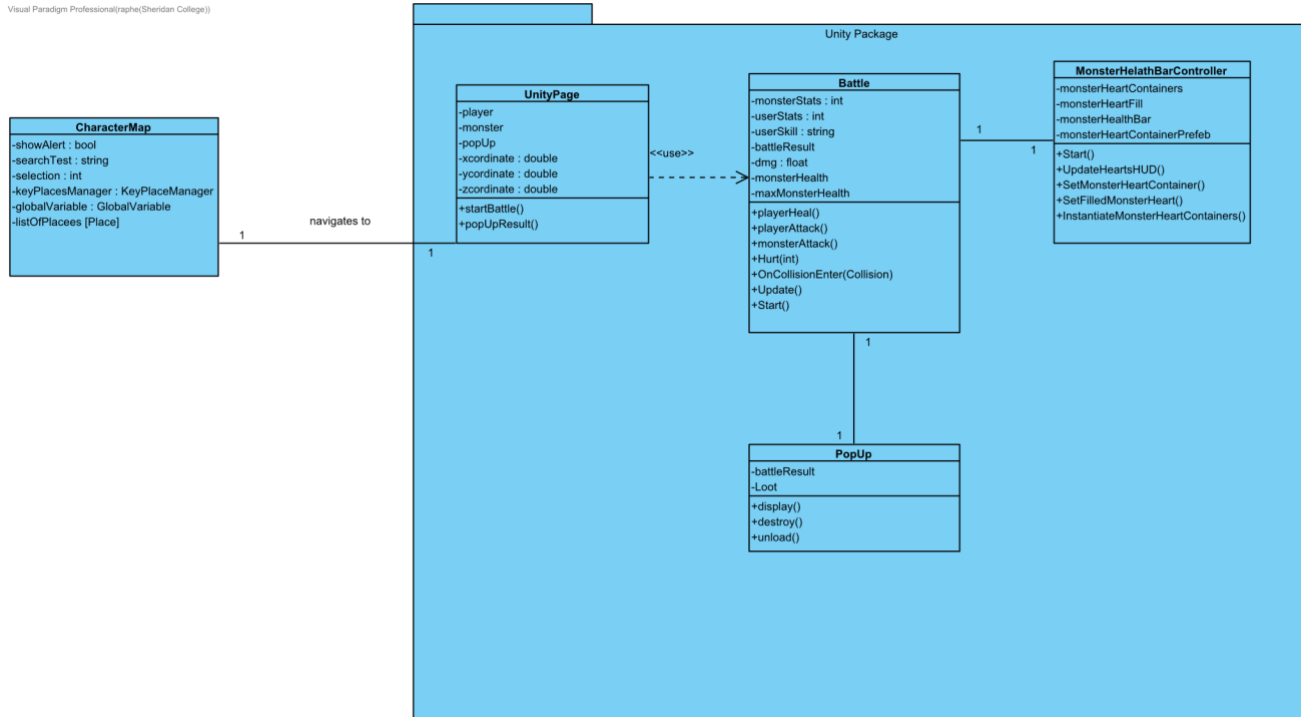


22. Structural Overview - User meets encounters

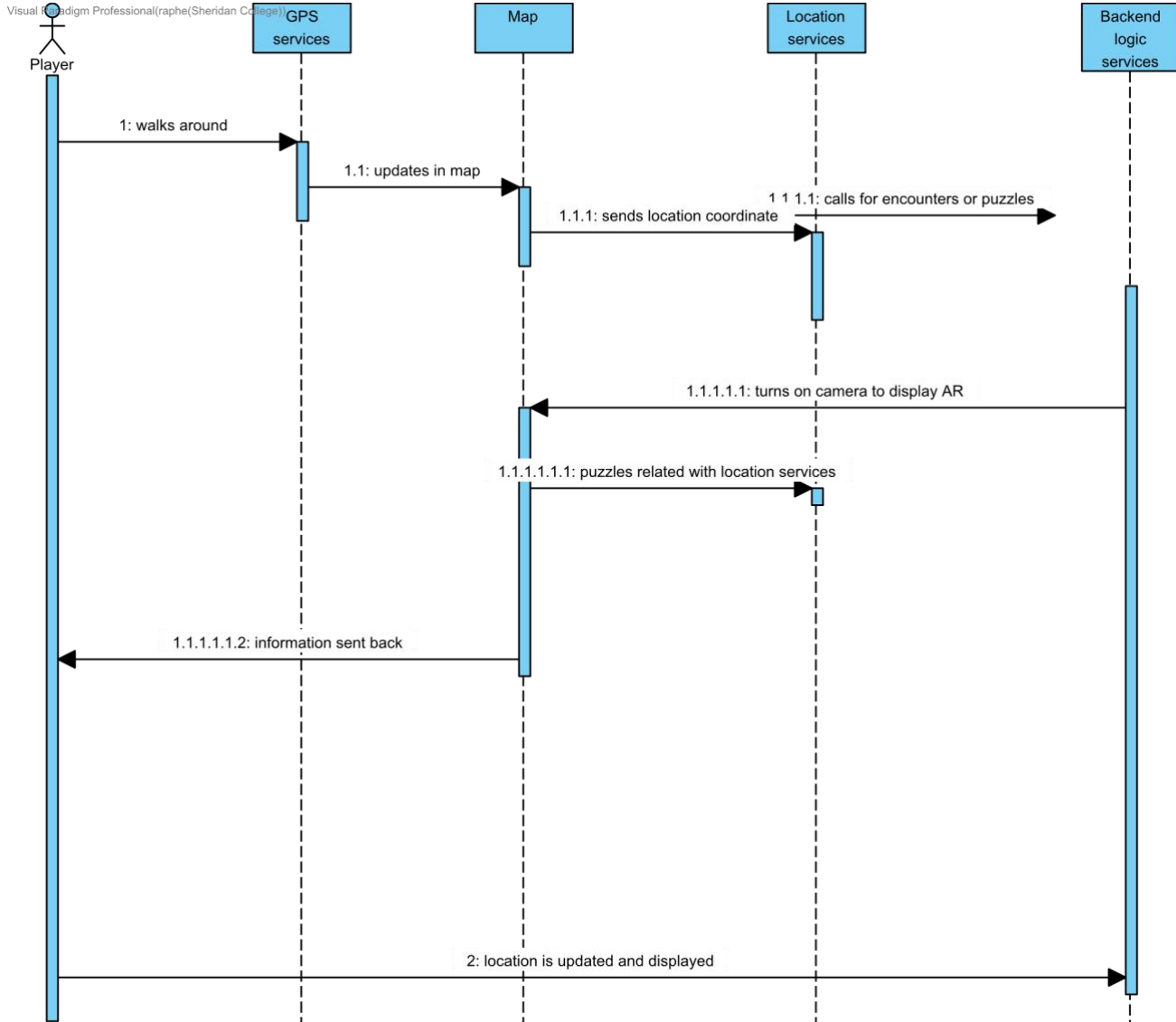


23. Structural Overview - user plays AR battle

Visual Paradigm Professional(raphie(Sheridan College))

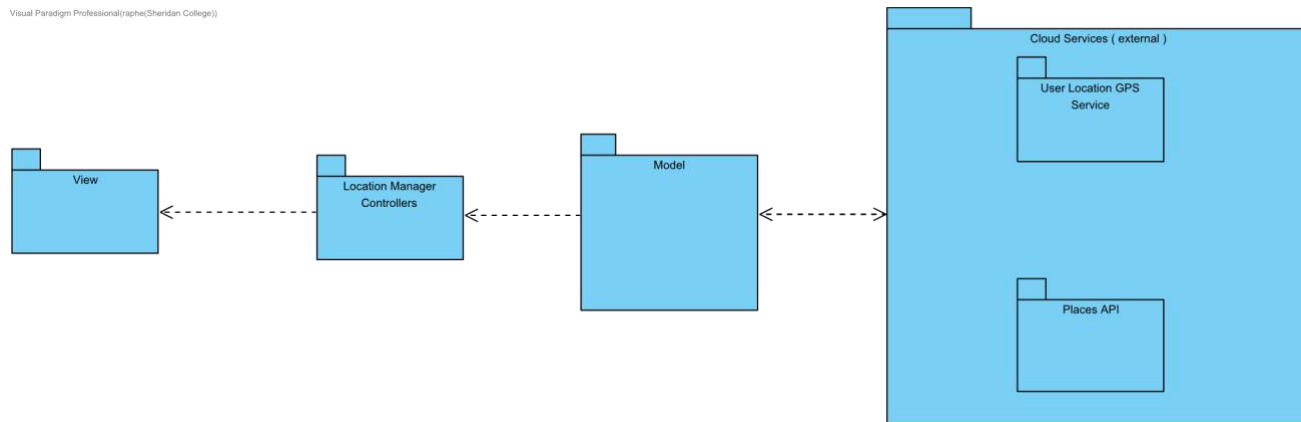


24. Behavioural Overview - User meets encounters



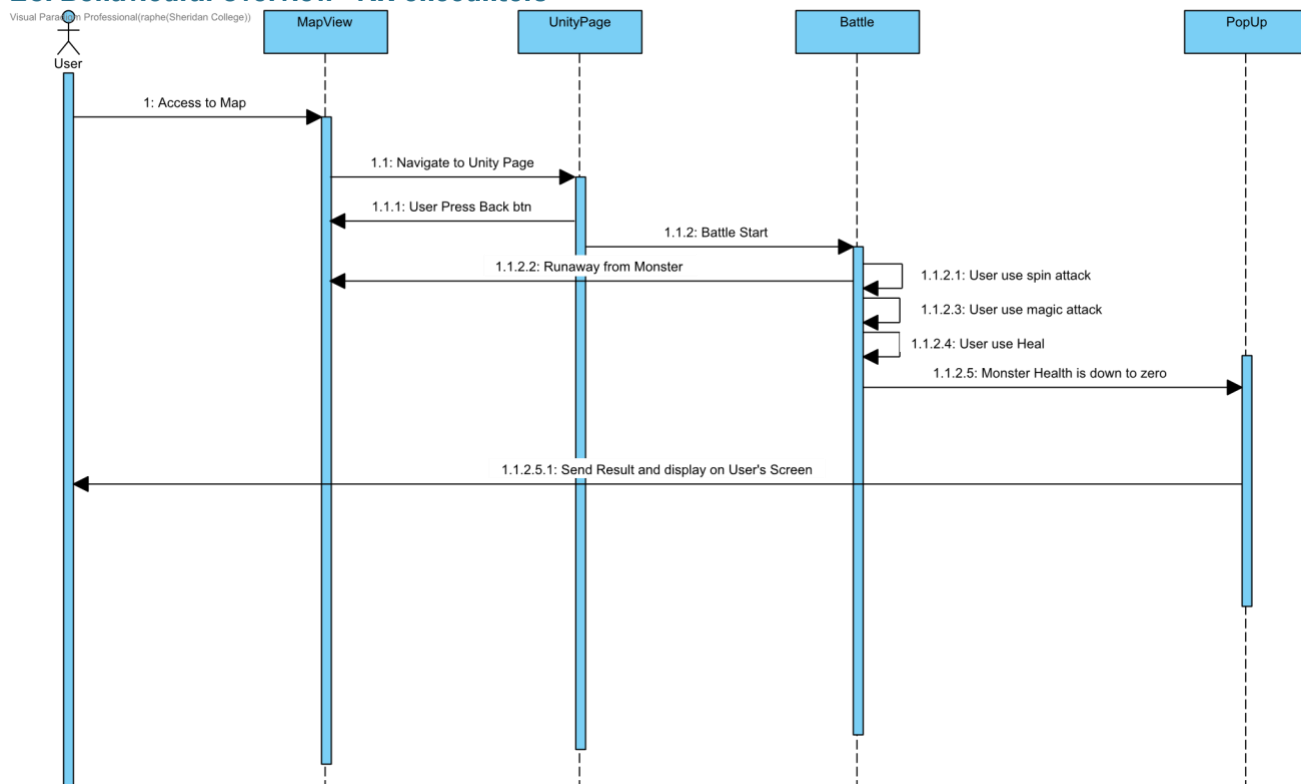
25. Collaboration Overview - Users meets encounters

Visual Paradigm Professional(raphel(Sheridan College))



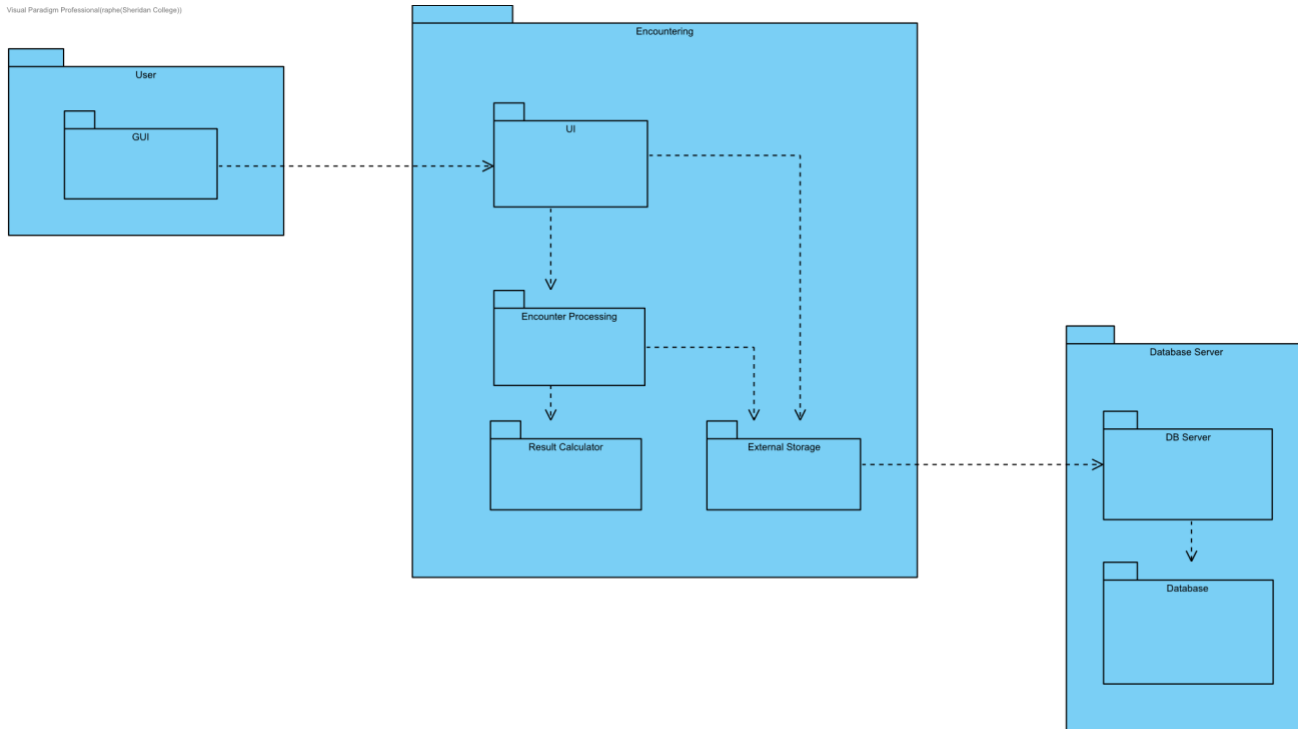
26. Behavioural Overview - AR encounters

Visual Paradigm Professional(raphel(Sheridan College))



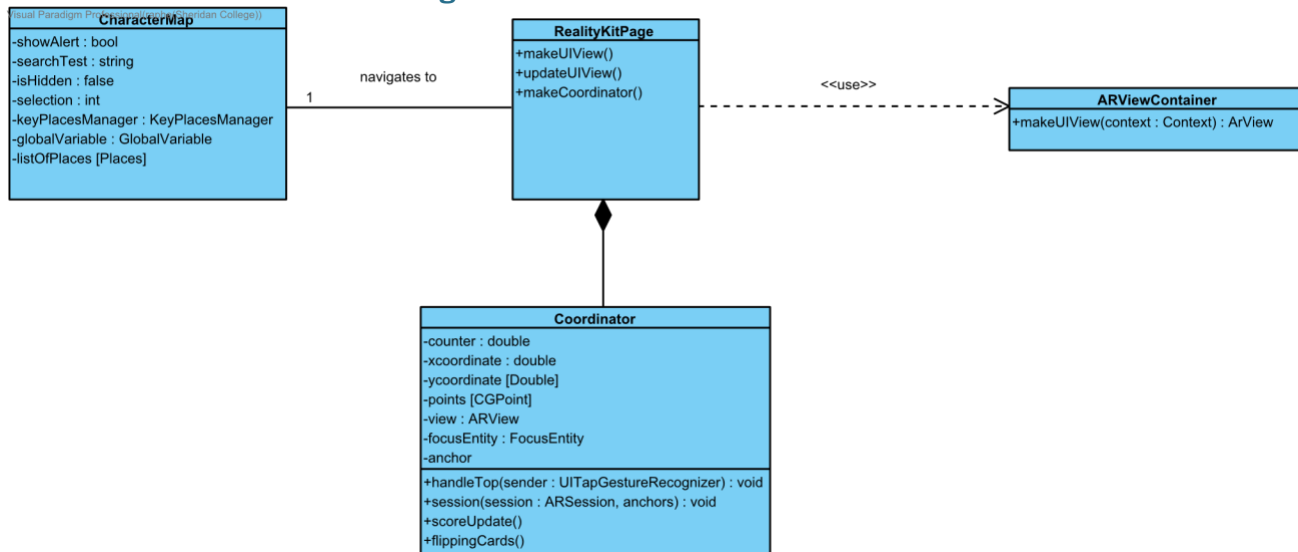
27. Collaboration Overview - AR Encounters

Visual Paradigm Professional (http://www.sheridan.ca)

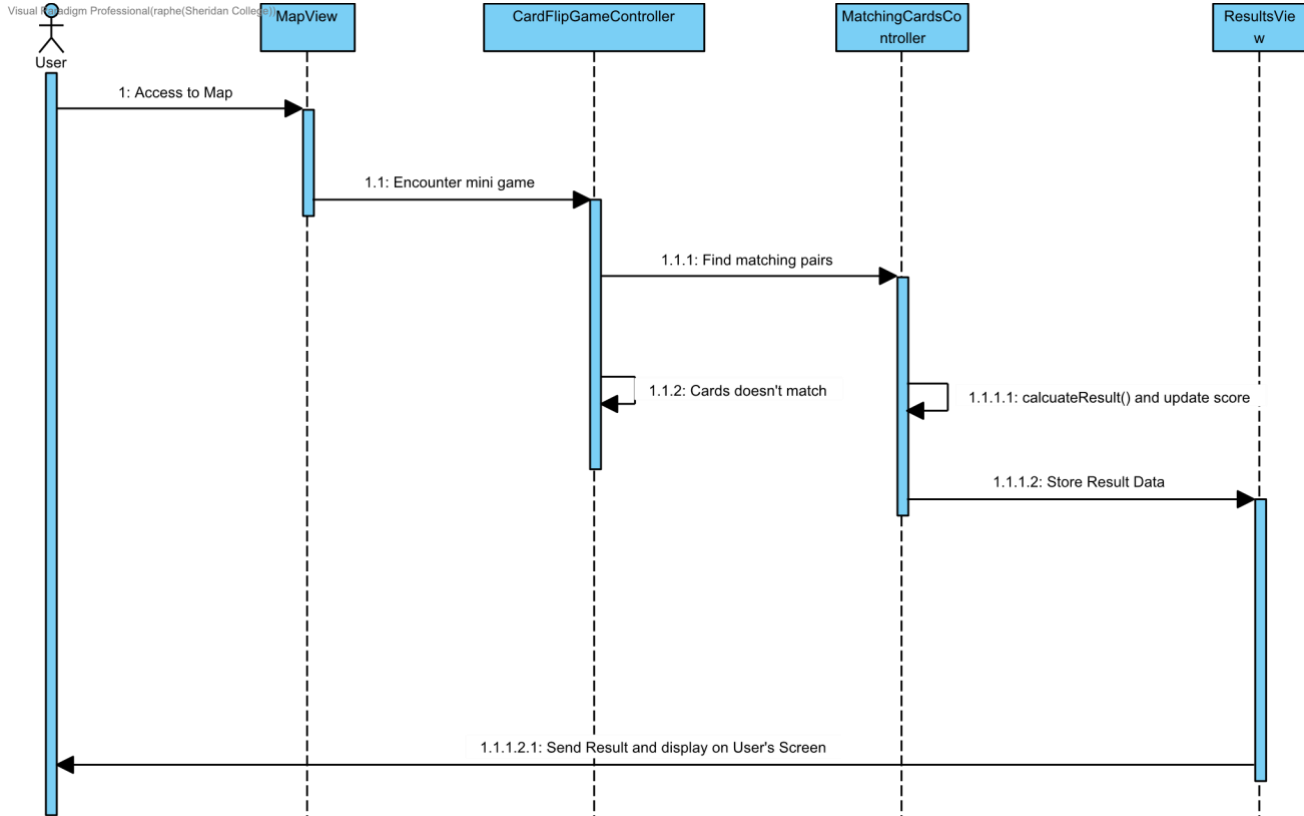


28. Structural Overview - AR Mini game

Visual Paradigm Professional (http://www.sheridan.ca)

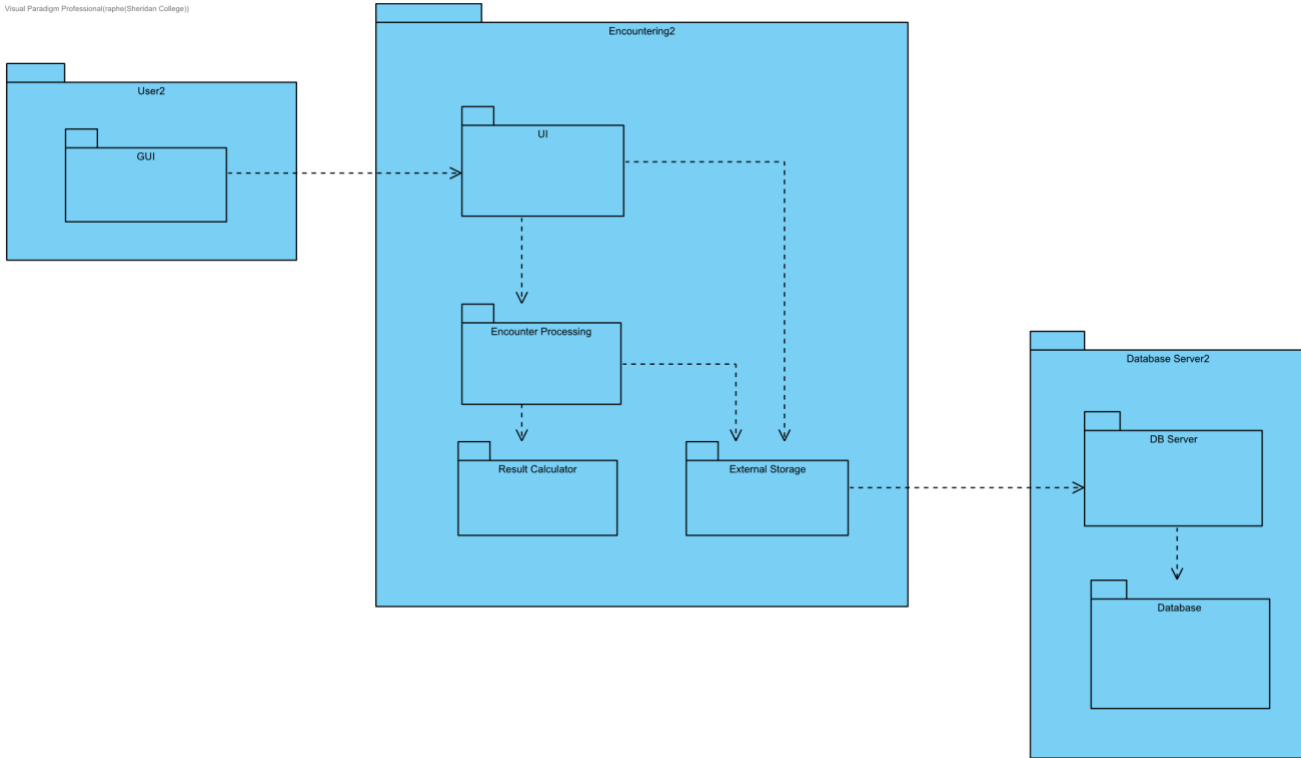


29. User Plays Flip Card mini game Sequence Diagram



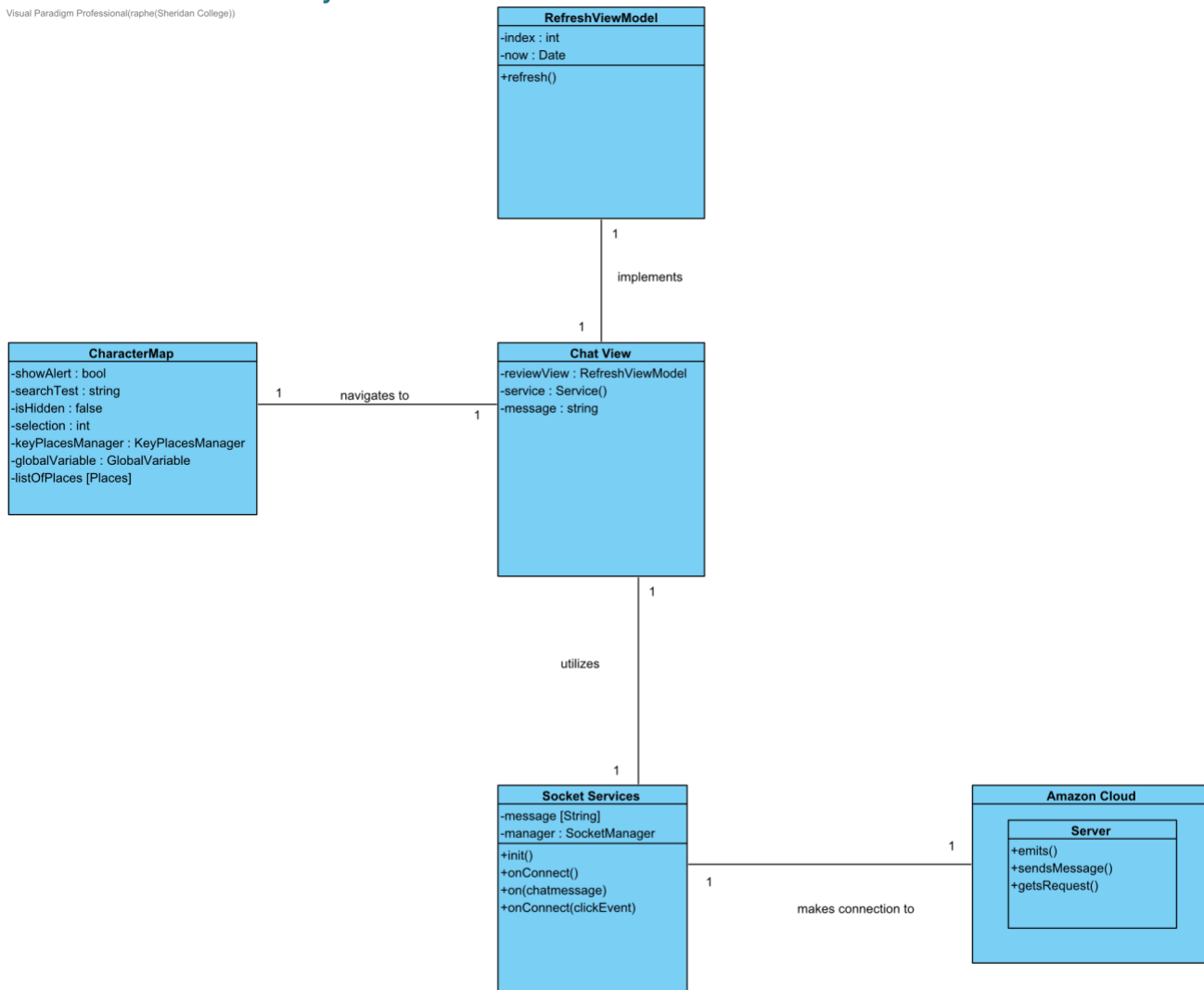
30. Collaboration Overview - AR Encounters2

Visual Paradigm Professional (rshe@Sheridan College)

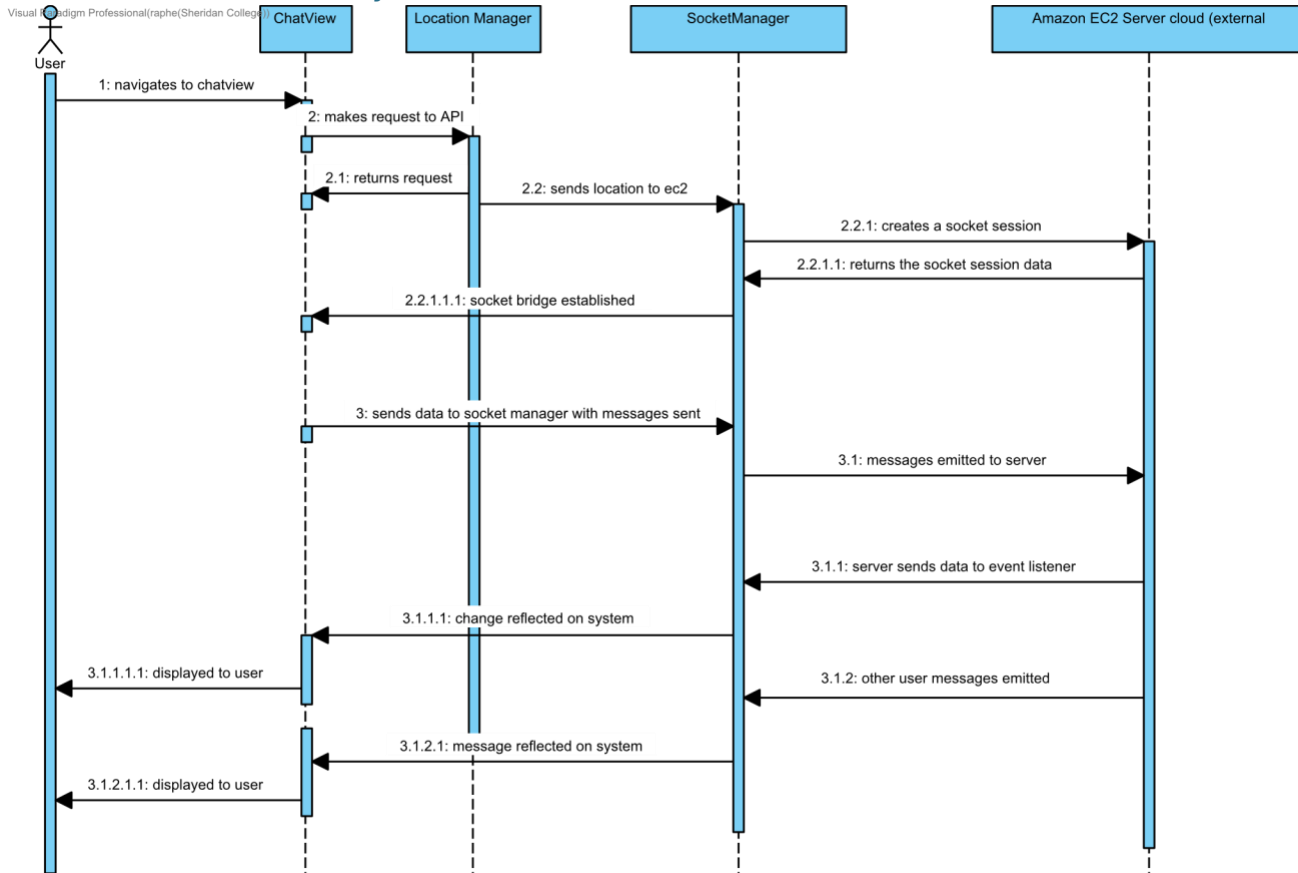


31. Structural Overview - System interacts with Cloud

Visual Paradigm Professional(raphel(Sheridan College))



32. Behavioural Overview - System interacts with Cloud



33. Collaboration Overview - Cloud Connections

