FUNCTIONAL SPECIFICATION

AR-T - AUGMENTED REALITY LEARNING APPLICATION FOR TECHNICAL GRAPHICS

STUDENT NAME: RUSSELL BRADY

STUDENT NUMBER: 15534623

SUBMISSION DATE: 30/11/2018

SUPERVISOR: MONICA WARD

TABLE OF CONTENTS

1. INTRODUCTION

OVERVIEW

AR-T is an Augmented Reality based learning app for Technical Graphics. The aim of this document is to serve as a reference for the system design of my proposed application and to list and describe its core features and functions as well as provide a general analysis of the system.

AR-T will come in the form of an Android Application and will be targeted at devices Android 7 or higher (Minimum SDK version 24).

The main aim of the app is to act as an educational tool in aiding the learning of the Junior Cycle Subject Technical Graphics (TG). TG is an introductory subject into the area of 3-D object representation and it is therefore important that students have the best learning experience they can have, to set them up for the subject in the future.

The idea for this project, and for this application, stems from the unsatisfied need for better and more immersive resources for teachers in teaching such an abstract and difficult subject. It is the aim of this app to help students gain a greater grasp of the basic concepts and a greater understanding of 3-D object representation.

BUSINESS CONTEXT

The nature of this application makes it an ideal solution for teachers as a supplementary resource to the material they will be covering in class. Across all subjects in Secondary schools, there are certain supplementary resources which are recommended to use due to their ability to help in the learning process. It is the aim of this application to make the learning experience for this subject more immersive and hands-on as well as being less mundane.

There were several questions which needed to be considered in approaching this project as a learning application and supplementary resource.

DOES THE SOURCE MATCH OUR LEARNING AIMS?

This is a question which is asked of any resource which is brought into the classroom, supplementary or not. Does the source match the objectives that the teacher is trying to achieve? The aim of AR-T is to provide an experience that students will not only enjoy, but that will meet the teachers learning aims for their class also, as this is the main aim of all resources.

DOES THIS SOURCE MEET DIFFERENT LEARNING STYLES THAN THE ONES YOU'VE ALREADY COVERED?

Often, the best supplementary material is that which looks at the subject from a different angle. In this case, AR-T will provide a completely different means of learning the subject. The goal of this project is to allow students the opportunity to observe and interact with actual 3-D object representations in space through Augmented Reality as well as providing features that meet different learning styles and in turn will offer a learning experience that gives students a greater understanding of the subject.

DOES IT MATCH MY LEARNERS' NEEDS AND INTERESTS?

This is often the main reason for supplementary resources. The material on offer in coursebooks may not cover the topic extensively or to a level with which the teacher may be happy. In the case of TG, with such an abstract and 3-D centred subject, it is hard to fully grasp and understand the subject from the pages of a book. The subject is particularly difficult to grasp at the early stages, especially when you are working from principles which are outlined in a book. AR-T will provide an experience which can engage students but also cater for their needs in the learning process.

WHAT ADVANTAGES DOES IT OFFER OVER WHAT IS ALREADY AVAILABLE?

Although AR applications are becoming increasingly popular, there is currently no app like this available as it stands. There are many online resources which offer questions and worked solutions. However, there is no resource currently which incorporates Augmented Reality to try and actively help students understand 3-D object representation in the subject, an area which is paramount in the success of a student.

IS IT APPROPRIATE FOR THEIR LEVEL, AGE AND CULTURE?

It is important that the app be appropriate to the students' ability, combined with their age and culture. It is the aim of the app to first cater for students in the early stages of learning the subject. This means the app must be designed with the appropriate content directed at their ability and age, as well as providing an app which is comprehendible and easy to use for the age profile and by students of all cultures. The app will be designed as so and implemented in development.

GLOSSARY

Augmented Reality: A technology that superimposes a computer-generated image on a user's view of the real world, thus providing a composite view.

App: An application, especially as downloaded by a user to a mobile device. The app will refer to AR-T in this case.

Technical Graphics: Subject where you will learn how to represent 3-D objects on paper and on computer, helping you to develop problem solving and creative thinking skills through the solution of graphical problems.

Cloud Anchor: Cloud anchors are a feature provided by Google to AR apps where you can host your AR object. Somebody on a different device can then also look at your hosted AR object by using a provided Api Key. This will be utilised and implemented in AR-T.

2. GENERAL DESCRIPTION

PRODUCT FUNCTION

There is no other TG resource which uses interactive content combined with augmented reality. This means that this app would be a forerunner into the area. It is the aim of the app that the inclusion of Augmented Reality will offer a different learning style and allow students to fully understand from basic concepts, 3-D object representation.

The closest product there is to this proposed application is DCG solutions. DCG solutions is an online teaching resource which provides interactive solutions to sample questions in the subjects' TG and DCG (Design & Communication Graphics – a follow on to TG). It is a very comprehensive resource and provides step-by-step guides to drawing sample solutions.

SYSTEM FUNCTION

Upon launching the app, a splash screen will be displayed showing the apps logo until the app loads. Once the app loads then the user will be brought into the login/signup screen. This will serve as an interface for both teachers and students.

As a teacher you can sign up with an email and password. Once the teacher has signed up, they can then set up a class, registering all the class details. A key is generated for the class and this key can then be used by the students to become a member of the class. Once the teacher is signed up they are free to sign in and out as they please using their username and password.

As a student you can also sign up with an email and password, however a class key will also be needed upon signing up to put you into the appropriate class with your teacher. Once the class key is entered then the student is free to sign in and out as they please using their username and password.

The teacher can assign questions to the student for them to complete. Once the exercise is assigned then the student can complete it and send a picture to the teacher for them to grade and leave comments.

There will also be learning exercises which will involve the element of AR. Different topics will be covered for the student to access and learn using AR. These topics will explain and go through topics step-by-step and explain to the student as to how to draw the given object.

Another feature which will be available to the class will be the ability to share an object representation between multiple devices. This means that numerous students and the teacher can use their phone to look at the same object, and from different angles. The idea behind this is to provide an interactive learning experience as it allows the teacher to explain the object representation to multiple students using different devices.

USER CHARACTERISTICS AND OBJECTIVES

The system will be aimed at two sets of people, namely the students and the teachers. The students of a secondary school generally range in age from 13-18 years old. The app would also be used by teachers of Technical Drawing. The app will be available on mobile and tablet devices. Tablet devices are becoming increasingly common in schools and it is important that the app is usable on both types of device.

The objectives of the system from the teachers' point of view is to have a resource which can be used to keep track of students' progress in terms of learning course content and topics, as well as completing sample questions which are provided to the students.

The objective from the students' point of view is to have an app which is fun to use and helps them to understand course content in an interactive manner. It will be important that the app utilises learning styles which aren't attainable by a book or normal online resource, otherwise it holds no advantage.

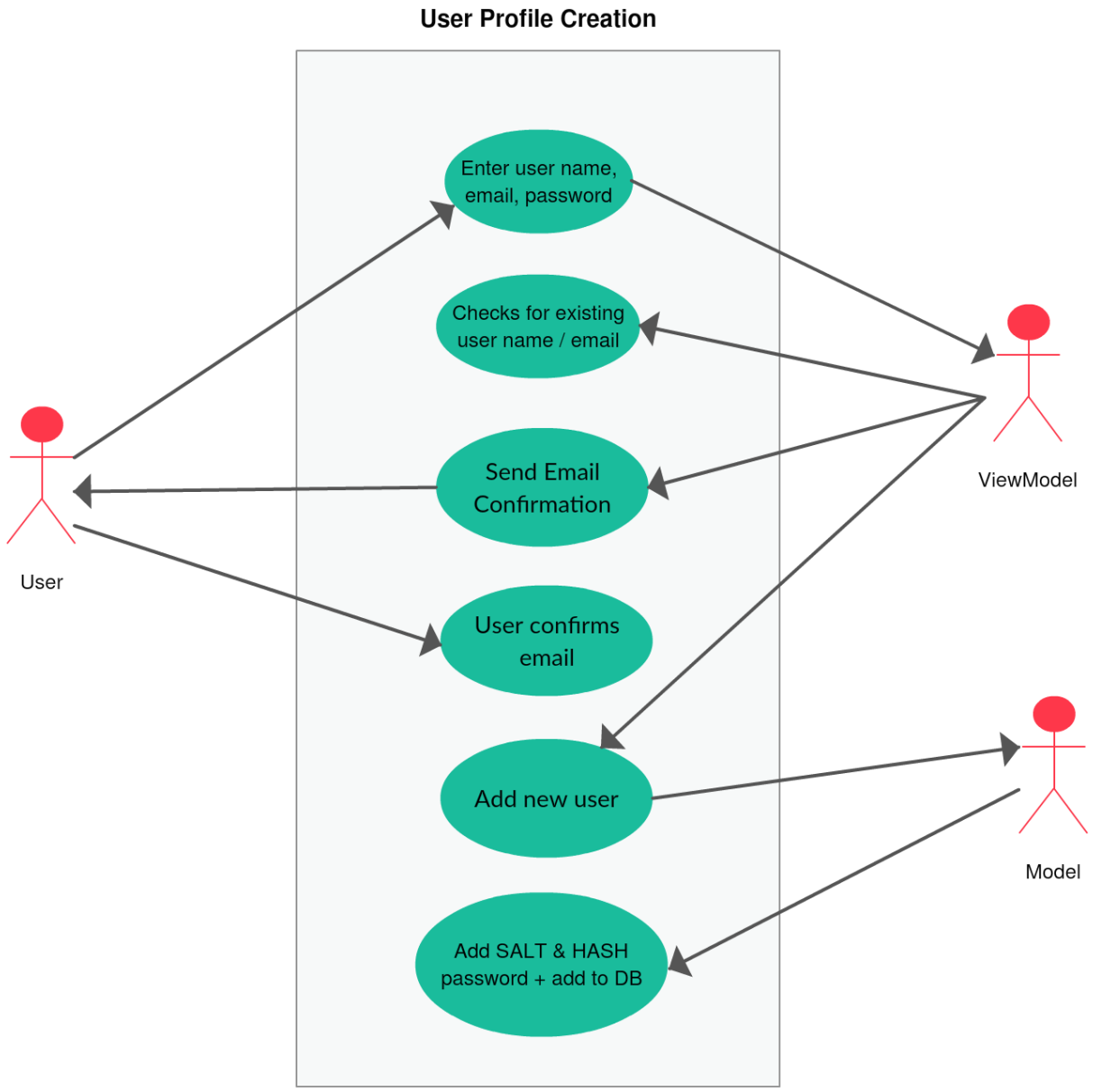
OPERATIONAL SCENARIOS

For the purpose of this Functional Specification I will provide some of the operational scenarios which will be included in the app. These will only be a subset of the functionality available however they should give some insight as to the workings of the application.

USE CASE 1 – USER PROFILE CREATION

Goal in Context	User wishes to create a profile for the application.			
Scope & Level	Account Creation – Client and Database			
Preconditions	User has email not already associated with account. Student must know class key.			
Success End Condition	User has successfully created a profile with a username ID and password (class key for student).			
Failed End Condition	Profile creation failed.			
Primary Actors	User – Teacher / Student		**Secondary Actors**	ViewModel Model
Trigger	User chooses "Sign Up"			
DESCRIPTION	**Step**	**Action**		
	1	User enters username, email & password		
Extended	2	ViewModel checks email / username validity		
	3	Controller sends email to user's email address		
Extended	4	User confirms account via email		
	5	User information is passed to the Model		
	6	Model adds SALT, hashes password and stores in database		
EXTENSIONS	**Step**	**Branching Condition**		
	2.1	Username / password already exists in the system		
	4.1	User doesn't receive email		
EXTENSIONS	**Step**	**Branching Action**		

	3.1	User is asked to provide different email / username
	4.1	User tries again



USE CASE 2 – ACCESS AR CONTENT

Goal in Context	User wants to open, view and interact with AR content		
Scope & Level	Client & Database		
Preconditions	The user is signed up and logged in		
Success End Condition	AR content loads and is rendered on the device		
Failed End Condition	AR content fails to load and is not displayed		
Primary Actors	User - Student / Teacher	**Secondary Actors**	Database
Trigger	User enters the app		
DESCRIPTION	**Step**	**Action**	
	1	User clicks on learning section of app and choose a topic area	
	2	User clicks on an example and then clicks on button to open the AR View	
Extended	3	AR image fetched from database and is returned to the view	
	4	AR model is rendered in the view and user can anchor it to a flat surface	
	5	Once the model is anchored the user can interact with the learning content	
Extended	6	User interacts with user content	
EXTENSIONS	**Step**	**Branching Condition**	
	3.1	ViewModel fails to fetch AR image from database	
	6.1	User sets up cloud anchor for others to join in interaction	

EXTENSIONS	Step	Branching Action
	3.1	Error message is thrown giving reason for failure (e.g. no internet connection)
	5.1	Other users can open app on their phone and share this AR space

CONSTRAINTS

TIME

Due to the nature of the project and the limited time that is available, development of the product will be very time constrained. There is only so much that can be done within the allotted time in a project which is so vast and has so much room for expansion. The app will contain all the essential features and will be designed so that extension is easy in the future. Aside from development of the product, the time needed to learn and understand the implementation of AR in the first place must also be taken into consideration. This is an area with which I have no prior knowledge and so will take time to get used to.

TESTING

Since this project is an educational tool and learning application for teachers and students, it will be important that I test the app using real users studying and teaching the subject. This usability testing will also take up considerable time and will involve requesting ethical approval. Apart from this, the application will need to be thoroughly tested to ensure that the AR element works under all types of conditions, e.g. lighting conditions, and that all other parts of the app work as expected.

3. FUNCTIONAL REQUIREMENTS

The following functional requirements include software and hardware interfaces, as well as the main functionality of the app itself.

EXTERNAL INTERFACES

SOFTWARE INTERFACES

Android API	To develop an Android app, the latest API will be required.
Google Core	Google Core is an AR library which will assist in the development of the augmented reality component of the app.
MySQL Database	This database will be used to hold user information and credential, as well as data pertaining to completed questions and learning material.
Rest Api	Rest Api on the server end will take requests from the android app and then access the database from here. I will use Node.js as my Rest Api.
Push Notifications	In order to set up push notifications I will need to have a server which listens for events which are fired and then notifies all other devices. I will use Firebase Cloud Messaging (FCM) for this.

HARDWARE INTERFACES

Android Device (7+) In order to use this app, the user will need to have an Android device which is Android 7 or higher (Minimum SDK version 24). I will also need this while I am developing the app.

OPERATIONAL ATTRIBUTES

PERFORMANCE & RELIABILITY

Performance and reliability will be very important in the success of this app – especially in relation to the AR aspect to the app. 3-D rendered models can be very memory expensive and it will be important that this be handled appropriately. 3-D models will only be loaded on request as it would be too expensive to try and pre-load models.

The performance of the app would suffer if it was tasked with loading 3-D models in a background thread. As well as this, reliability might suffer if memory wasn't managed efficiently as you could end up with a full cache causing the app to completely slow down.

Users history will be handled by running a background thread to preload the users' history, so it is available to them upon opening the appropriate page of the app. The data will be passed into the local SQLite database using Room persistent library. Data will then be loaded on request from the local database.

MAINTAINABILITY

Maintainability is a measure of the ease to modify code. Higher maintainability means less time making changes. I will be implementing a MVVM design pattern to help in the implementation of a modular design to cater for easy expansion and scalability. This design will be expanded further in the system architecture.

As features are expected to evolve over time, interfaces and class abstraction should be utilised rather than very specific, non-generic classes. As well as this, business logic should be kept away from the view as this allows testing of the code and easy change of the UI.

PORTABILITY

Portability will also be considered. Although development of an iOS / Web based app is not in the scope of the project backend components should be designed and configured that any device would be able to connect. This would make possible expansion to other platforms in the future easier.

SCALABILITY

For this project, the app will not cover all possible topics in Technical Graphics. This app will provide a snapshot of the scalability and vast expansion possible. The app will include a small number of features developed to a high standard to act as a proof of concept as to the benefit of using an Augmented Reality app in the teaching of the subject.

SIGNUP / REGISTRATION

DESCRIPTION

One of the most fundamental aspects of this application is that it acts as an interface between teachers and their students. It is important to make this distinction from signing up and to then allow the app to act accordingly depending on the user it is dealing with. Every user will have to sign up at the start, teachers with a username and password, and students with a username, password and class key which will be provided to the teacher once they set up their class on the app.

CRITICALITY

This is an essential feature to the app as without the teacher / student interface it would not function correctly as a teaching resource. If this feature was omitted, the application would not truly reflect the inspiration behind the idea. As well as this, the app is not intended for anonymity so registering is an important process.

TECHNICAL INFO

The design of the database will be very important in allowing correct registration of teachers and students. It will also be important that there be some level of abstraction between the two. The appropriate information needs to be populated correctly in the database as the users' credentials is then linked to and used by other features in the app.

LOGIN / LOGOUT

DESCRIPTION

Once users are signed up they can then login using their credentials and logout of the app from settings. Since users should be allowed to log in and out of the app for privacy purposes, a log in interface would be necessary to deal with this. The inclusion of this feature would also mean that one account would not necessarily be linked to one phone and that numerous people could sign in and out of the same device.

CRITICALITY

Following on from signing up, the login functionality would be quite important. It is an important feature as it provides an element of privacy and security in the app.

TECHNICAL INFO

Login will involve querying the database with the username and password. Successful login will bring you into the appropriate interface, depending on whether you are a student or teacher. Unsuccessful login will throw an error. Logging out will bring the user back to the login screen where it will not be possible to gain access to any area of the app without first providing your credentials.

AUGMENTED REALITY VIEW

DESCRIPTION

When the user accesses the learning part of the app, they will be able to open AR content. This view will open the users' device camera to display and will then render the chosen 3-D object representation. The user can choose where they want to place the object and they will be able to 'stick it' to any flat surface. Once they have placed the object in a fixed position then they will be able to interact with the object. There will be animations to help students with the learning topic and the user will also be able to interact with the object by touching the screen and carrying out other actions (One may be waving your hand in front of the camera).

CRITICALITY

This is a key feature in the app and is essential in providing the learning experience. Part of the objective of the app is to try and utilise a different style of learning and to match the needs and interests of both the teacher and the student. The element of AR gives this app its advantage and so, the inclusion of this main element would be critical in the success of the app.

TECHNICAL INFO

To implement this feature, I will make use of the AR library, AR Core. A game engine like unity may also be needed to help in the development of the AR.

MULTI-USER AUGMENTED REALITY VIEW

DESCRIPTION

This multi-user experience will allow me to connect multiple users together over the web to create multi-user AR experiences which share a synchronized augmented space. Users on multiple devices will be able to share the same content across their devices. This provides huge learning

possibilities as teachers and students can collaborate and talk about the 3-D object representation as well as interacting with them.

CRITICALITY

This feature is not a critical component to the app however it provides cool functionality and would offer a fantastic opportunity for collaboration on a topic in the subject. This feature would help the app to meet a different learning style, namely a more interactive and collaborative learning style.

TECHNICAL INFO

To implement this feature, I will need to implement a 'Cloud Anchor'. This will allow me to create the multi-user element of the application where devices can share an augmented environment. I will also need to provide the relevant interface to allow multiple users to easily share this multi-user view.

POST CONTENT

DESCRIPTION

There will be an interface where teachers and students can post content onto the app. This will help the app to act as a complete educational tool. Teachers will be able to post a sample question onto the app for students to complete and in turn the students complete the question and post a picture of their solution where it can then be graded with comments from the teacher.

CRITICALITY

This requirement is not essential to the learning aspect of the app, however looking at the app as an educational tool you can see how this would provide some important functionality. Being able to have an interface where students and teachers can communicate as well as post problems and solutions helps the learning process.

TECHNICAL INFO

This functionality will rely on the database. Questions posted by the teacher will need to be held in the database under an appropriate table of questions related to a certain class. Solutions related to a certain question which students then post will need to be held in the database under an appropriate table of solutions linking to the relevant class and question.

VIEW CONTENT

DESCRIPTION

Inside the main app, users, be it teacher or student will be able to check the history of past questions and solutions. Teachers will have access to all the questions they posted as well as all the solutions

which were handed in relating to that question and the feedback which was given. Students will be able to see all the questions which were assigned to them as well as their solution and any feedback which was given. All this previously posted content can be viewed again but not edited. The implementation of this history log will be in the form of a Recycler View which will cycle down through the history in order of date.

CRITICALITY

Since users will be able to post content it would be implicit that the ability to view previous content which was completed should be available. This feature offers teachers the ability to keep track of students' progress and assess how they are getting on in the subject. For the student, it provides a place where they have all their past solutions stored for reference in the future and can be useful in studying for exams etc.

TECHNICAL INFO

This functionality will rely heavily on storing the data from all previous questions and solutions and having a reference to the appropriate teacher / student of a class. When a student or teacher logs in to the app, all their data relating to previous questions and solutions should be fetched from the database based on their username or some other primary key. All the history should then be available to them in the appropriate screen of the app, preloaded on a background thread.

4. SYSTEM ARCHITECTURE

SYSTEM ARCHITECTURE COMPONENTS

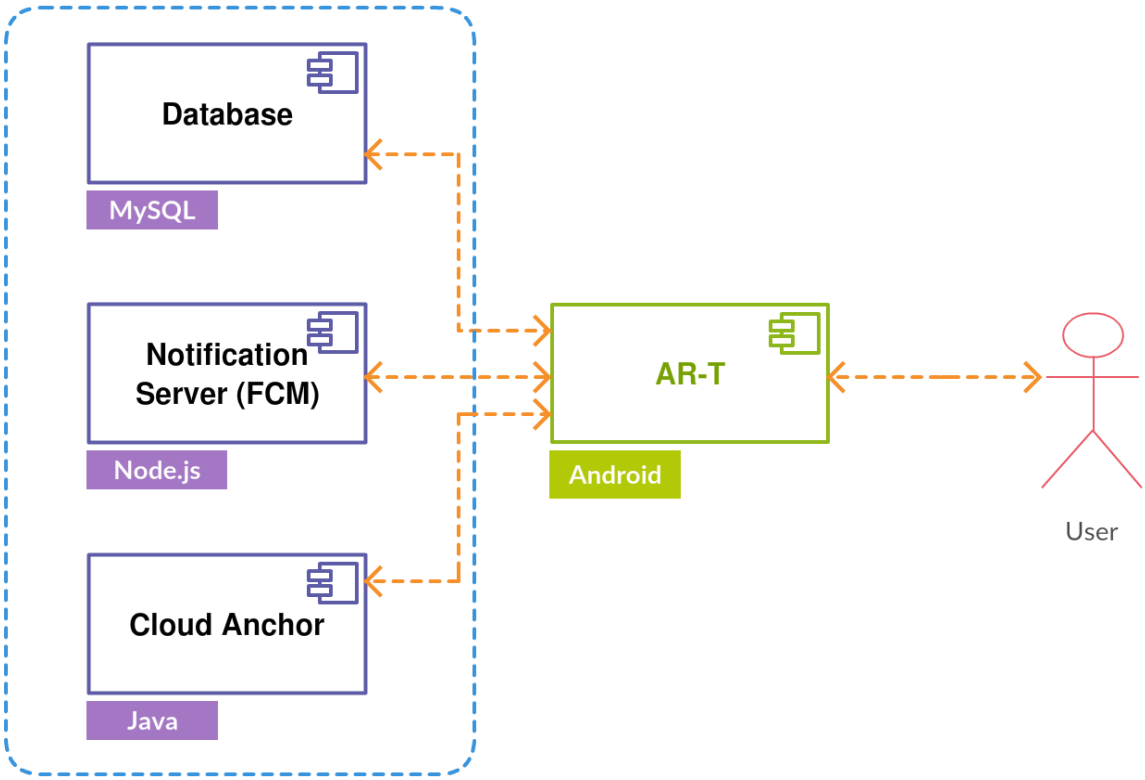
AR-T is an Android Application and as such, the Android App will form a major component in the overall system. The app will be written in Java and will implement ARCore to make AR development possible. However, the application will not be able to function correctly without the help of some other components which will make up the rest of the system architecture.

As there are users involved in the system a Database will be needed to hold user credentials and other user data relating to the app. Such data may include previous drawings completed for certain questions given. The AR images will be quite memory expensive and may also need to be stored in the Database.

The system will include a remote server to handle implementation of push notifications on devices. The server code will be written in Node.js and will be based on Firebase Cloud Messaging (FCM) to send the push notifications.

To handle a multi-user experience, I will need to implement a cloud anchor. These are anchors that are hosted in the cloud and can be used / resolved by multiple users to establish a common AR space across users and their devices. The Api Keys for this can be stored in my database but Google host the actual anchors.

The diagram below shows a very high-level system architecture which includes the systems main components.



SYSTEM ARCHITECTURE DESIGN

AR-T is an Android Application and as such, will be designed using Android design principles, namely the Model-View-ViewModel (MVVM) design pattern. There are many different design patterns and architectures to choose from currently, however the MVVM design pattern is currently the most popular and widely adopted amongst the community currently and there is a lot of online support in implementing the architecture.

WHAT IS MVVM?

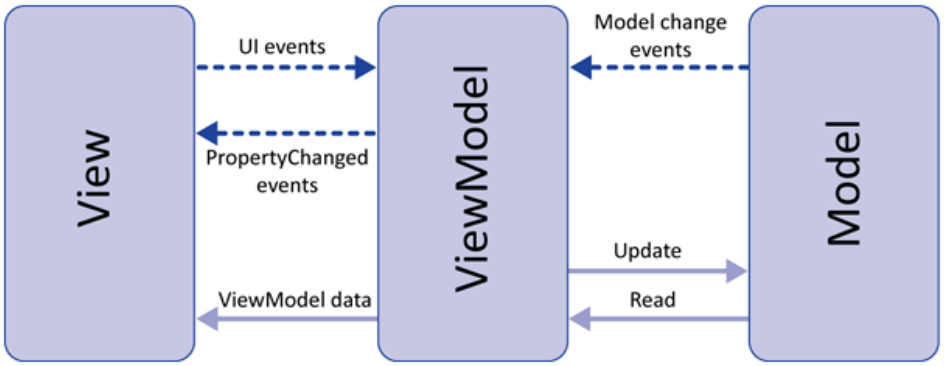
As mentioned above, MVVM stands for Model-View-ViewModel and the architecture consists of these three main components.

Model The model represents the data and business logic of the app. Data is exposed to the ViewModel through observables. The model can also act as an abstraction layer between the data and the database. Local models will 'model' the data in the database.

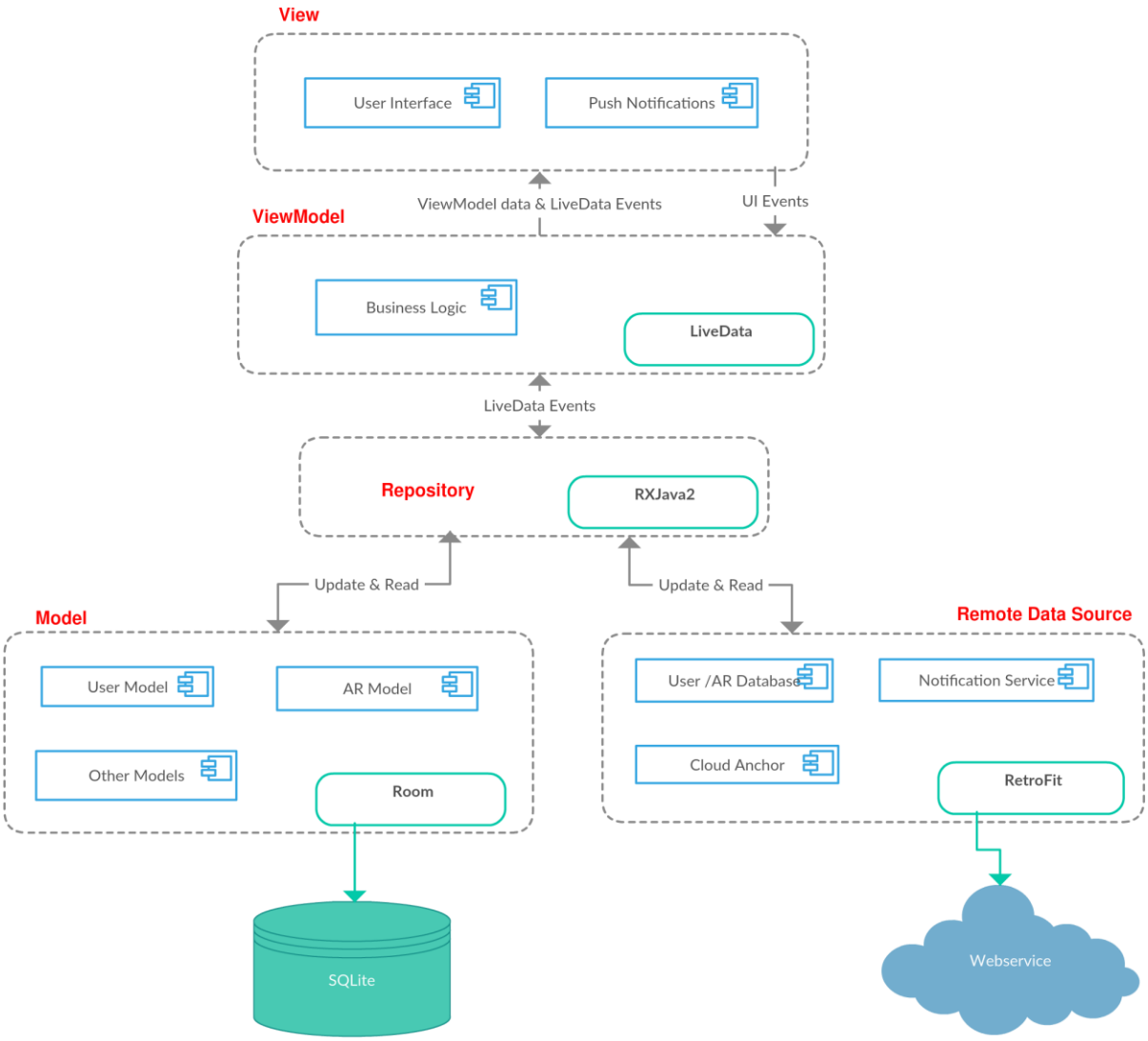
View The view informs the ViewModel about the user's actions. It also observes the ViewModel to get data to update the UI elements. It is the presentation layer for the model and is essentially what the user will see when using the device. Minimal activities with many fragments will be utilised to make switching between views quicker and more fluid.

ViewModel The ViewModel exposes streams of data relevant to the View. The ViewModel retrieves the necessary data from the Model, applies the UI logic and then exposes it to the View for it to consume. Similar to the Model, the ViewModel exposes data via Observables. There will be ViewModels for every fragment which will retrieve and expose the relevant data to that fragment (view)

The following diagram shows the three components and their interactions as explained above:



MVVM ARCHITECTURE IN AR-T



AR-T will be broken down into multiple modules working together to form a larger system. This modularity will be key in the apps extensibility in the future. Several components will be remote – these are the Database, Notification Service and the Cloud Anchor. The app will also have a SQLLite database locally to handle any data which needs to be stored on the device full time, or when it is fetched from the server. I will then implement the MVVM architecture to bring together all these components into one seamless system.

In the android app I hope to only have a minimal number of activities (e.g. Login Activity, Main Activity, AR Activity). This is because Activities are very expensive and need intents to move from one activity to another. Fragments will be created on top of the activities as reusable UI layers which with their own logic embedded in them. Each fragment will have its own ViewModel which will carry out this UI logic and pass the output to the view to be rendered on the screen. This will be done through LiveData / Observable events.

There will be several repositories in the app. Each repository will hold a specific purpose for the information it needs to handle. The repository provides an abstraction layer between the App and the remote data source. Retrofit will act as the HTTP client which will interact with the RESTful API on the server and the observables will pass information back to the ViewModels from the data repositories.

Implemented correctly, this architecture provides a perfect balance between modularity and a well-integrated application.

5. HIGH-LEVEL DESIGN

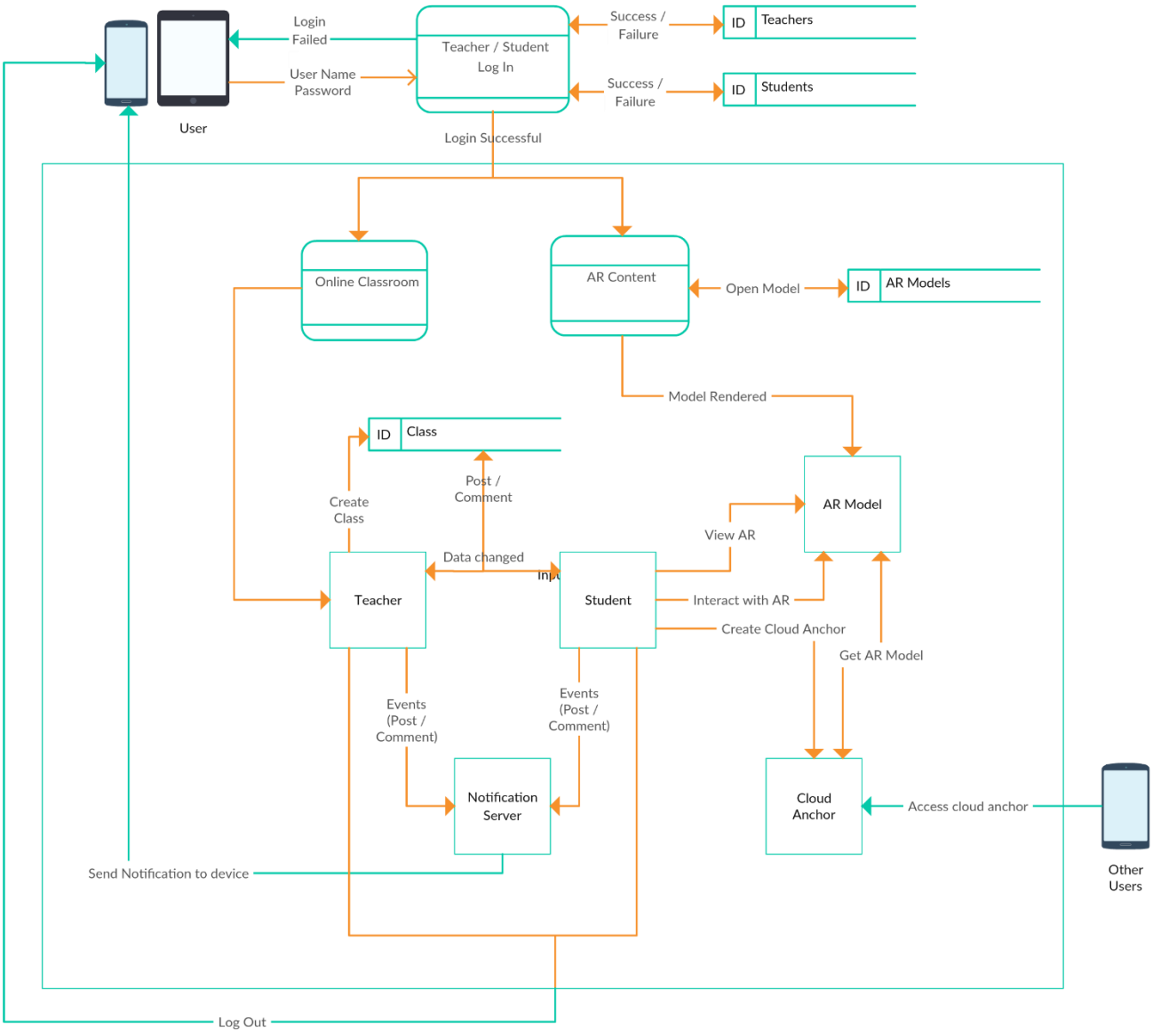
As can be seen from the relevant diagrams in the System Architecture section above, there are several different components which will be working together to form the overall system. The system design will involve bringing all these components together yet allow the system to be modular in fashion and open to extensibility.

Expanding on the architecture mentioned in the section above, I will have a remote data store and a local repository. The reason for this is to support offline mode. I will have specific repositories which handle both cases and then a repository which then handles data from both (Local & Remote).

Now that the system architecture is clearer, and the relevant components have been outlined it is important to define the flow of data between these components as the user works their way through the app. The best way to model this interaction is using a data flow diagram. The following DFD will provide an overview of the system in terms of the information which gets passed through the system and is sent to / fetched from different components.

The DFD includes a user which logs as either a teacher or student. The appropriate table in database will be checked and return a failed / successful login. If the user logs in successful they can access the classroom or the AR content. Teachers and students will have slightly different interfaces as they will be allowed to do slightly different things.

DATA FLOW DIAGRAM



6. PRELIMINARY SCHEDULE

OVERVIEW

Below is a Gannt Chart which has been created to represent the intended of the project. It will be important to stick to the timeline as much as possible to ensure all parts of the project are covered and time doesn't run short at the end.

In completing the project, an Issue board might be useful in managing issues and bugs. Implement an agile process will also be important so that task completion can be split up into manageable blocks. Gitlab has its own issue board so I will be sure to utilise it.

MAJOR TASKS

There are several major tasks which will make up my project and once completed in their entirety should lead to the completion of the project. These are the tasks which have been identified as being key to the success of the project and the success of the finished application. Some of the tasks don't even pertain to the actual development of the application but are important in the scheme of the project.

COMPLETION OF FUNCTIONAL SPECIFICATION

Completion of the functional spec will be key in going forward with the development of the application. The system architecture, high level design and functional requirements will all be referenced in the design and implementation of the app. No development will start until this is completed as it is important to have a clear idea of the system architecture and components.

RESEARCH & DESIGN

Research and design of the app is ongoing and will continue until the components of the app have been finalised. Once sufficient research has been carried out on implementation then development can start on these features. Here are the main areas of research:

App Interface: The design of the app interface will need to be majorly considered as it will be of vital importance that the application be as usable as possible to cater for students of all kinds as well as the teacher. I will look at the design of the initial app interface first and then begin development of the app.

Augmented Reality Technology: Research into the area will involve figuring out how AR can be best implemented in the app to give a great user experience. Google have many talks on AR app design and I am looking at these to gain insight on design principles and best practices. Research into this area will take time due to the fact it is a new and emerging field with which I have no prior knowledge.

Backend: Research in relation to the backend will involve finalising the backend components which will be used to store and access information remotely. As well as this the design of the backend components and how the app will interact with them will be an important decision as the app will be relying on these backend components. Once components have been decided and he app interface has been started, I can begin integration with the backend components.

Pedagogy: Research into the pedagogy element of the application will be one of the most important areas of research to be undertaken. I am currently in contact with a TG teacher to help me in my research and will be in contact with others to try and gain as much insight as to how the app can best introduce a great learning experience. This process will continue throughout the project and will be a continual collaborative and learning process.

DEVELOPMENT OF CORE FEATURES

Once the appropriate research has been carried out on the areas mentioned above and the various components of the project have been finalised then the development process can begin. It will be important as mentioned above to break the development process down into its core components. These components and their timelines will be outlined in the Gannt chart below. I hope to follow a clear process of development keeping on schedule as there are several components following development which need time allocated to them.

TESTING

Testing will form a vital part in the development of this project. The testing phase will involve several components and will not be limited to the end of the project but will be an ongoing process throughout the project lifecycle.

The testing process will consist of (but not limited to) the following:

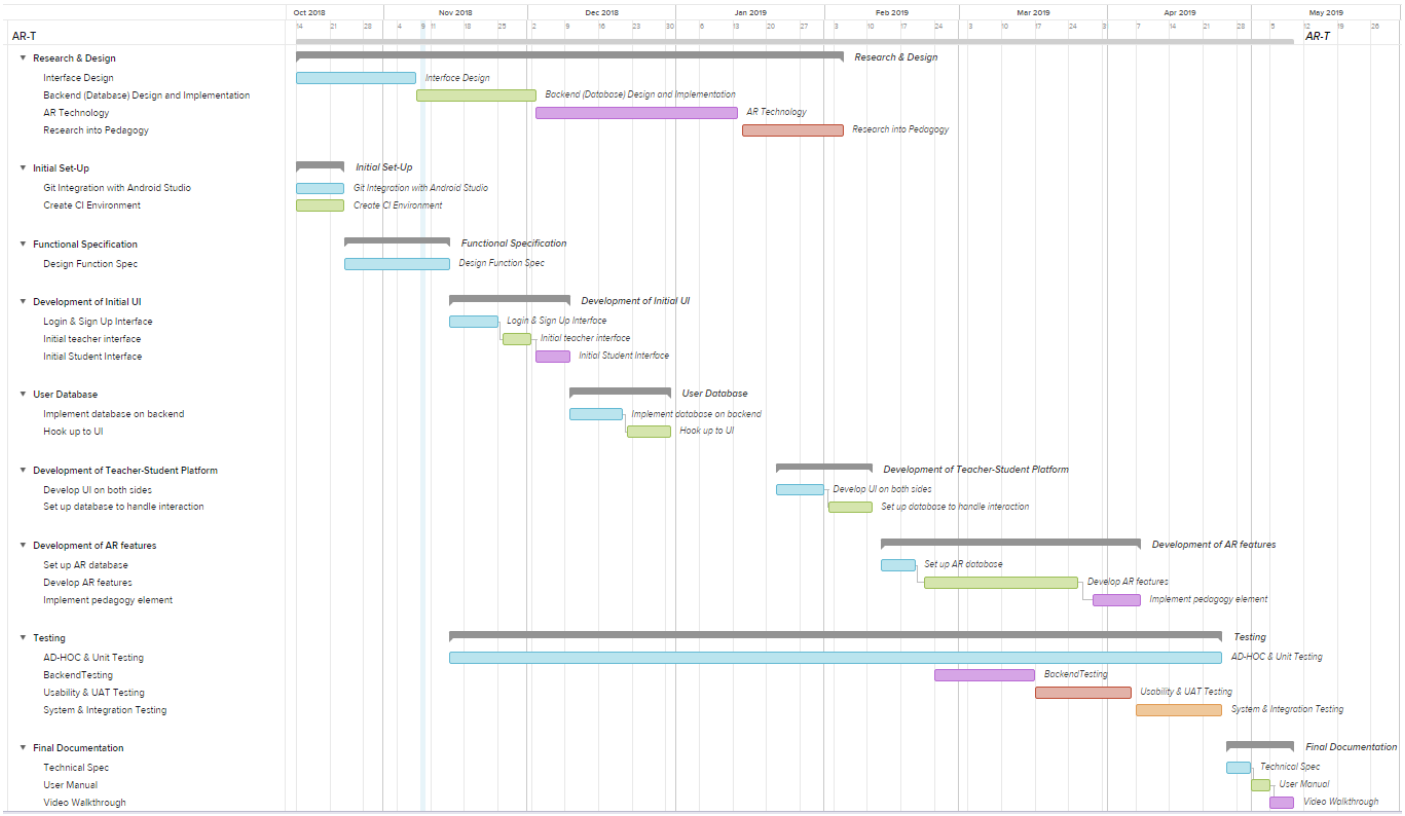
- Continuous Ad-Hoc Testing during development
- Unit Testing
- Backend Testing
- Usability and User Acceptance Testing
- Integration and System Testing

As part of the testing process I will be setting up a Continuous Integration (CI) environment to allow constant testing of code throughout the development process. I will do this through GitLab's built-in CI platform.

FINAL DELIVERABLES

For the end of the project there are several final deliverables which will need to be given time consideration. The first deliverable is a video walkthrough. Completing this walkthrough will take considerable time and will also depend on having all features of the project finished. Aside from the video walkthrough, there is a technical guide and user manual which need to be included as part of the final project deliverables. I have allocated time for all of these at the end of the project, but it is vital that I stick to schedule to ensure I don't end up under serious time pressure trying to complete these key project deliverables.

GANNT CHART



7. APPENDICES

REFERENCES

Pedagogy research - https://www.shmoop.com/teachers/curriculum/textbooks-resources/supplementary-resources.html

https://www.teachingenglish.org.uk/blogs/davedodgson/bringing-outside-%E2%80%93-choosing-supplementary-resources-classroom

Gantt Chart - http://prod.teamgantt.com

Diagrams – https://creately.com

Cloud Anchors - https://codelabs.developers.google.com/codelabs/arcore-cloud-anchors/index.html?index=..%2F..io2018#0

ARCore - https://developers.google.com/ar/

Rest Api using retrofit - https://medium.com/@saquib3705/consuming-rest-api-using-retrofit-library-with-the-help-of-mvvm-dagger-livedata-and-rxjava2-in-67aebefe031d

Push Notifications - https://medium.com/@salonikogta/beginners-guide-to-implementing-push-notifications-in-android-e896ef54b831

Remote / Local Storage - https://medium.com/@eslam.hussein/dominate-remote-local-data-with-rx-retrofit-room-mvp-f2b13a0ac27b