TECHNICAL MANUAL

WRITTEN BY SAM COOGAN & ALEX RANDLES

INTRODUCTION

OVERVIEW

1. The system which we developed was designed mainly for sufferers of Logopenic Progressive Aphasia - a form of Alzheimer's disease which occurs at a much younger age. The system could also be used by any person(s) who suffer from word retrieval problems.

   The system works on a carer/patient relationship. The carer is required to register a username and password which will be unique to them. This will be stored in our sqlite database. A carer can only access the system once they have registered successfully.

   A carer will be responsible for add and removing patients from their care. When adding patients, they must ensure that they have selected the appropriate preferences such as captions on/off and setting their pictures for each of their patients. Once they have added all the patients which they care for, they can show them their customised tiles that may or may not contains captions. They can either allow the patient to click on the tiles they want to find the word for or they could do this for them.

GLOSSARY

- **App**
  A mobile app which is a computer program designed to run on a mobile

device such as a phone/tablet or a watch. In this case we are referring to our Word Retrieval App.

- **GUI**

  Graphical User Interface. The user interface which allows the user to interact with the underlying code.

- **Sqlite**

  SQLite is a opensource SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation.

SYSTEM ARCHITECTURE

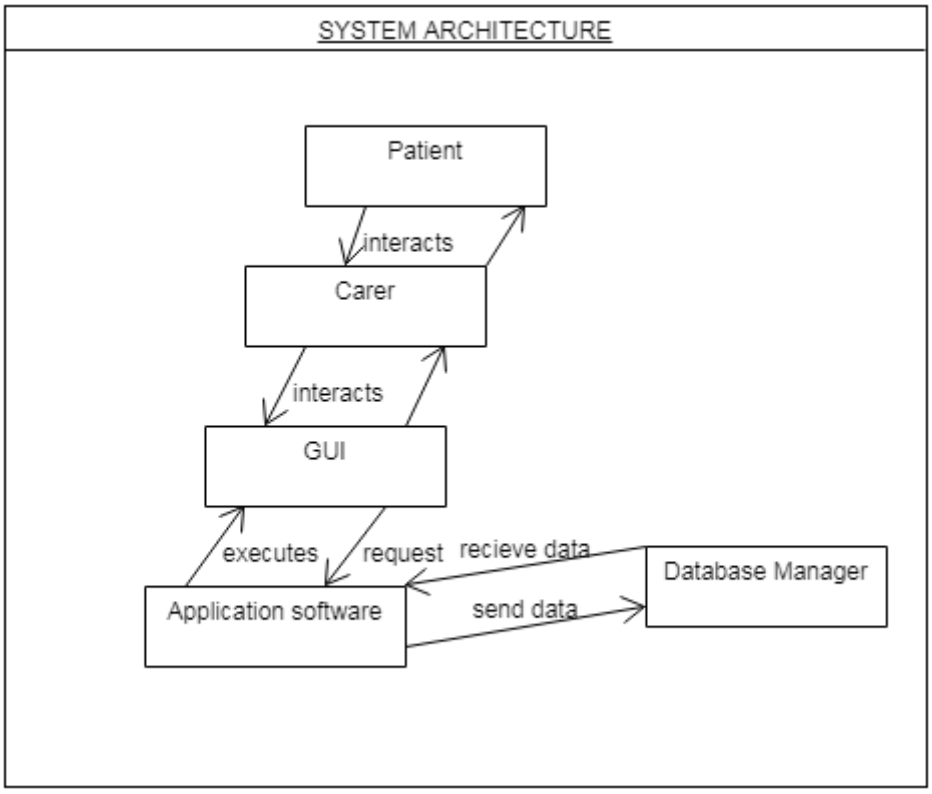


Figure 1

Figure 1 shows the architecture of our final app, there exist different modules of the application shown in the above diagram.

1. **Patient**

The patient is the end user of the app, they are the users who the app is designed for and they interact with the GUI through their carer.

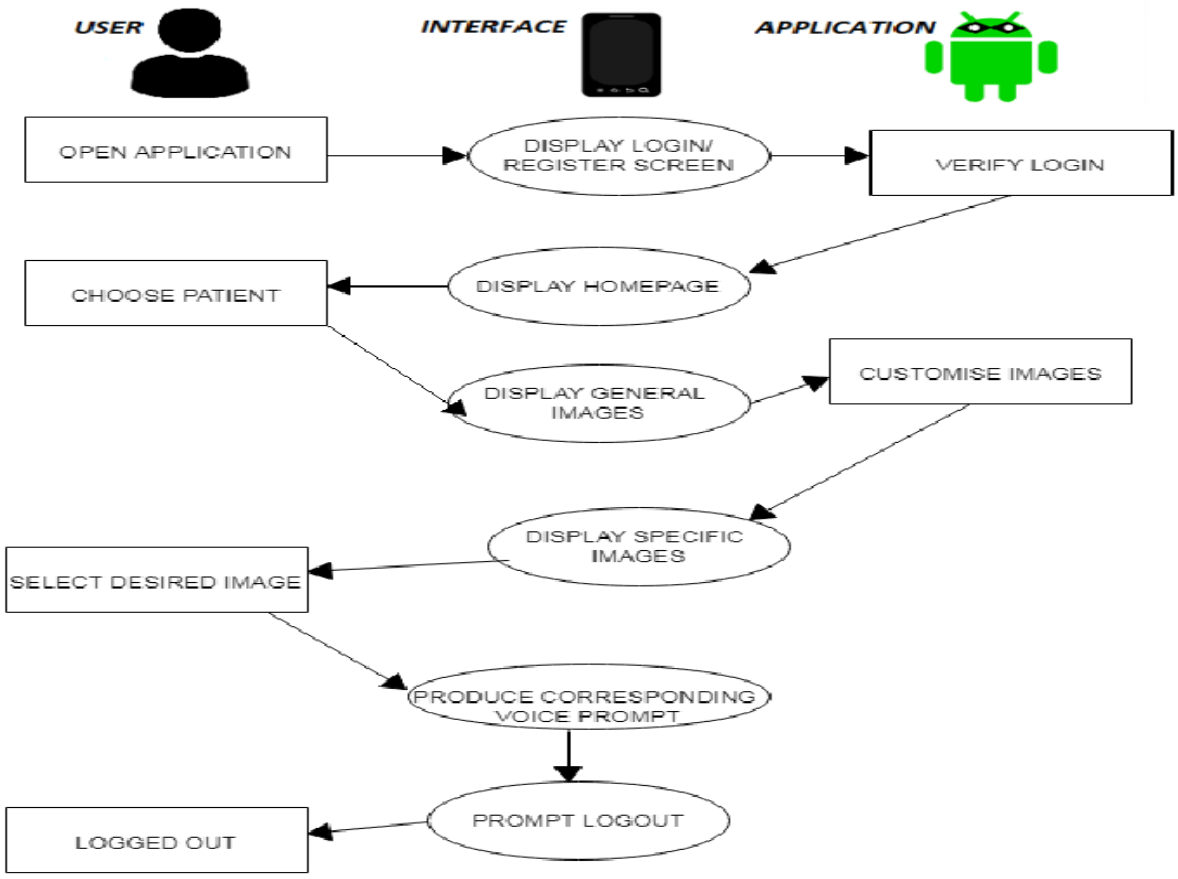
2. **Carer**
   The carer will care for a certain number of patients, they will help their patients to interact with the GUI of the app.

SYSTEM ARCHITECTURE

3. **Application software**
   This part of the software will contain all the code for our Application. This will decide how our GUI reacts when the user interacts with it. This code will be written using the Android Visual studio software and is one of the most important parts of the system.

4. **Database Manager**
   The database manager will manage all of the data that will be held in the database. The data will be sent and received by the application software of the system. The database follows clearly defined ER and business rules. We have implemented our database using sqlite.

HIGH LEVEL DESIGN



HIGH LEVEL DESIGN

• Step by step high level design

1. User opens app by pressing icon on phone
2. Login/Register page is then displayed
3. If login is successful, there custom homepage is displayed
4. The carer will then select the patient they are currently caring for

5. Their customised tiles will then be displayed
6. The carer will then select the image the patient wants
7. The caption and/or voice prompt will then interact with the patient

Problems & Resolution

- ## Problem 1

  Generating a voice prompt from the android device and ensuring it worked correctly each time

Solution 1

We discovered a media player class which is available in android studio and is already built in. Each time an image was pressed it would activate a button onclick function which would in turn start the media player and then stop it.

- ## Problem 2

  Making our App language independent.

Solution 2

This was a very tough task for us as it required us to ensure part of our App were not hard coded. We wanted Spanish speaking patients the ability to use the App. We did this by setting the locale to Spanish if this was selected and then referencing our Spanish strings.xml file instead of our English one. This took a bit of redesigning but was successfully completed.

- ## Problem 3

  Creating login/register database and implementing it correctly

Solution 3

This problem was solved by a huge amount of research on the internet. We firstly had to understand what was used to create a database in android studio and then learn how to do this. We decided to use sqlite to create our database and used correctly defined tables and ids.

Problems & Resolution

- ## Problem 4

  Turning captions on/off depending on patient's settings.

Resolution 4

We created a drop down menu for the captions when the carer is setting up their patient's preferences. If the patient's preference is set for no captions, then the Captions string will equal "OFF" and then all the text will be set to the empty string. If the captions are turned on then the captions will be retrieved accordingly from the strings.xml file.

- ## Problem 5

  Laying out images correctly and in a grid layout.

Resolution 5

We tried laying out our images in a constrained layout and we found out using this method would not work for laying out the images in our App. We decided to use a grid layout which worked much better. We did this by doing research on how to set up a grid layout and then implementing it within our layout.

INSTALLATION

Required software

- Android device running Android 5.0 Lollipop or newer
- Internet connection

STEPS

1. Access browser on your android device
2. Open http://student.computing.dcu.ie/~coogans2/wordup/
3. Click on the download link for version v0.2
4. Press ok on any permission questions
5. Press the install button
6. App is now ready to run

ER DIAGRAM & BUSINESS RULES

Business Rules

- A patient is the end user of the App, they will interact through the App by the carer.
- A carer is the user which will interact directly with the App and the patient.
- A topic is a group of items which are similar to one another.
- An item is article or unit which belongs to one or more groups.
- A group is a collection of items.

ER Diagram