

National College of Ireland
BSc in Computing
2015/2016

Sean D'Arcy
X11371436
X11371436@student.ncirl.ie

Bucket

Technical Report



Declaration Cover Sheet for Project Submission

SECTION 1 *Student to complete*

Name: Sean D’Arcy
Student ID: X11371436
Supervisor: Paul Hayes

SECTION 2 Confirmation of Authorship

The acceptance of your work is subject to your signature on the following declaration:

I confirm that I have read the College statement on plagiarism (summarised overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature: _____

Date: _____

NB. If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the College's Disciplinary Committee. Should the Committee be satisfied that plagiarism has occurred this is likely to lead to your failing the module and possibly to your being suspended or expelled from college.

Complete the sections above and attach it to the front of one of the copies of your assignment,

What constitutes plagiarism or cheating?

The following is extracted from the college's formal statement on plagiarism as quoted in the Student Handbooks. References to "assignments" should be taken to include any piece of work submitted for assessment.

Paraphrasing refers to taking the ideas, words or work of another, putting it into your own words and crediting the source. This is acceptable academic practice provided you ensure that credit is given to the author. Plagiarism refers to copying the ideas and work of another and misrepresenting it as your own. This is completely unacceptable and is prohibited in all academic institutions. It is a serious offence and may result in a fail grade and/or disciplinary action. All sources that you use in your writing must be acknowledged and included in the reference or bibliography section. If a particular piece of writing proves difficult to paraphrase, or you want to include it in its original form, it must be enclosed in quotation marks

and credit given to the author.

When referring to the work of another author within the text of your project you must give the author's surname and the date the work was published. Full details for each source must then be given in the bibliography at the end of the project

Penalties for Plagiarism

If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the college's Disciplinary Committee. Where the Disciplinary Committee makes a finding that there has been plagiarism, the Disciplinary Committee may recommend

- that a student's marks shall be reduced
- that the student be deemed not to have passed the assignment
- that other forms of assessment undertaken in that academic year by the same student be declared void
- that other examinations sat by the same student at the same sitting be declared void

Further penalties are also possible including

- suspending a student college for a specified time,
- expelling a student from college,
- prohibiting a student from sitting any examination or assessment.,
- the imposition of a fine and
- the requirement that a student to attend additional or other lectures or courses or undertake additional academic work.

Table of Contents

Executive Summary.....	7
1 Introduction.....	8
1.1 Background	8
1.2 Aims	8
1.3 Technologies	9
2 System	10
2.1 Requirements.....	10
2.1.1 Functional requirements	10
2.1.2 Data requirements	19
2.1.3 User requirements	19
2.1.4 Environmental requirements	19
2.1.5 Usability requirements	20
2.2 Design and Architecture	20
2.3 Implementation.....	21
2.3.1 Login and Register.....	21
2.3.2 The User's Bucket List.....	21
2.3.3 The Twitter Search.....	22
2.3.4 The Idea Lists	22
2.4 Graphical User Interface (GUI) Layout	24
3 Testing.....	27
3.1 Unit Testing	27
3.1.1 <i>Login/Register</i>	27
3.1.2 <i>Navigation</i>	27
3.1.3 <i>Saving, updating and deleting goals</i>	27
3.1.4 <i>Using the Twitter Search</i>	27
3.1.5 <i>Using the Google Maps</i>	28
3.1.6 <i>Viewing the idea lists</i>	28
3.2 Monkey Testing	28
3.3 Customer Testing	29
4 Further development or research	30

5	References	31
6	Appendix	32
6.1	Project Proposal	32
6.2	Customer Survey.....	36
6.3	Project Plan	40
6.4	Monthly Journals	41
	My Reflection	47
	Intended Changes	47

Executive Summary

'Bucket' is an android application with the goal of inspiring adventure and personal growth. It is marketed to all ages, from a child dreaming of what adventures their lives may entail, to an elderly couple looking for fun ways to spend their retirement years. It allows users to create their own bucket list, alter entries and delete them if they so choose. Lists of ideas, broken up into different categories, alongside the usage of the Twitter API inspires users to try new things. It also incorporates Google Maps to present users with a view of what nearby activities are available to them. It was developed in Android Studio using Java and the Android SDK, as well as PHP files to enable interaction with a remote database.

1 Introduction

1.1 Background

From a young age I dreamed of adventure. I would see places like the Alps in films and hear of thrilling activities like sky diving and excitedly add them to whatever scrap of paper that was operating as my bucket list. But every couple of weeks I would hear of something new and go to add it to my list, only to discover it had been lost, taking with it the fulfilment of getting to tick items off my list or even remembering what I had written. There were other problems with paper lists, such as thinking of things to add to the list when you do find it. Solving these problems and inspiring people to live fulfilling lives are the primary reasons I chose to build this app.

1.2 Aims

The aim of the application is to provide the user a number of different services.

Firstly, to save their lists to a remote database, to ensure that even if they lose their phone they will be able to access their lists from any device capable of installing the app. The app allows the user to add new items and edit or delete existing items.

The second aim of the app is to inspire the user. The app has several, frequently updated, lists split into different categories of possible ideas. Users can then search Twitter, from within the app, to see what pictures or stories people may have shared about the items in the ideas lists. For example in the ‘Sightseeing’ category users will find the “See the Aurora Borealis” idea. They can then search that term using the built in Twitter search to see what the public have to say about it. I believe the combination of these two services will prove to inspire users.

Visiting Niagara Falls, or learning a new language are long term goals and can’t be achieved on a Tuesday night after work. With that in mind the final aim of the app is to provide the user with a map of nearby activities that they can partake of, using the Google Maps API.

1.3 Technologies

The App was developed in Java using the Android Studio IDE. For any section of the app requiring a database, such as the login/register or saving the users bucket, I used PHP files saved in cloud 9 ("Sdarcy06"). I used phpmyadmin to gain access to the database. I used the Twitter API to allow users to search terms and see the 20 most recent posts containing the term. I also used the Google Maps API to present users with a map of nearby activities.

2 System

2.1 Requirements

2.1.1 Functional requirements

I will now explain the functional requirements of my app with the help of use cases and UML diagrams ("Flowchart Maker & Online Diagram Software"). They are as follows;

1. Simple Navigation around the App

While straight forward, this requirement is essential to the success of the app, as users will need to be able to effortlessly navigate around the different features of the app.

Use Case

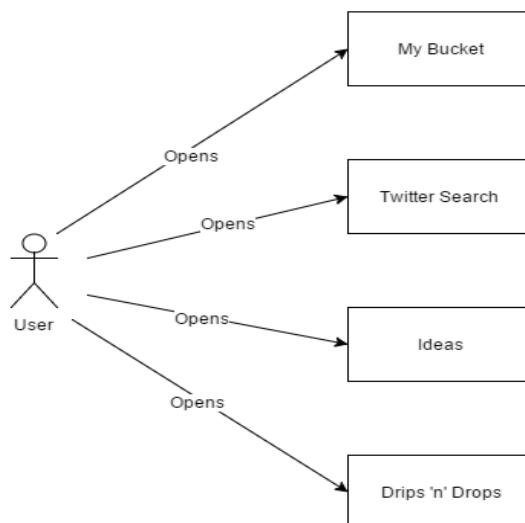
Scope

Enable simple navigation around the app

Description

This use case describes how easy it should be for a user to move between the different features of the app.

Use Case Diagram



Flow Description

Precondition

The system is idle.

Activation

The use case starts when the user logs in.

Main flow

1. The user clicks a button on the homepage
2. The feature opens

Alternate flow:

A1: <Using the dropdown menu>

1. The user makes use of the dropdown tab menu in the toolbar to open another feature
2. The feature opens.

Termination

The use case ends when the user logs out

2. To be able to register and login

The user's ability to register and login. Without this requirement, the user will never gain access to the other features of the app.

Use Case

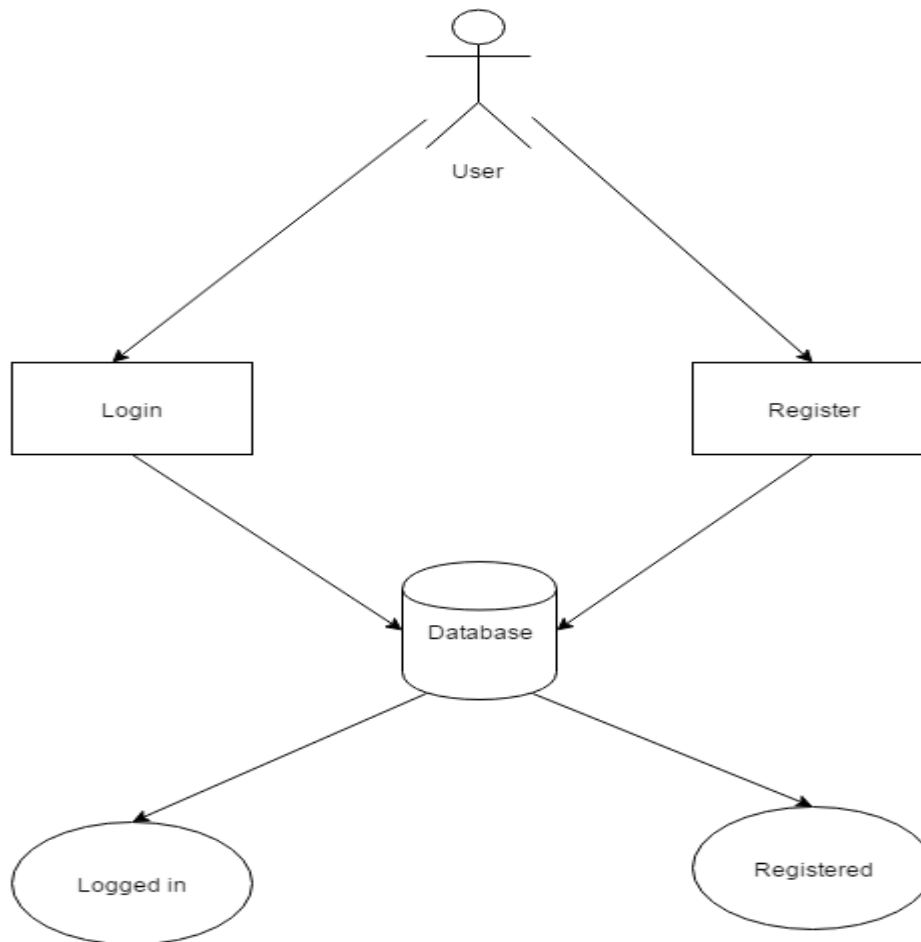
Scope

To allow the user to register for an account and then log in.

Description

This use case describes how a user will log in or register

Use Case Diagram



Flow Description

Precondition

The system is idle.

Activation

The use case starts when the user attempts to register or log in.

Main flow

1. The user presses the register button
2. The user chooses a username and password
3. The system checks if the username is already in use
4. The user proceeds to log in with those credentials
5. The system verifies the credentials
6. The user is sent to the homepage

Alternate flow

A1 : <Unable to register>

1. The user selects a username that is already in use
2. The system prompts the user to select another username
3. The user tries again

A2 : <Unable to login>

1. The user enters the wrong credentials
2. The system notifies the user that either the password or username has been entered incorrectly
3. The user tries again

Termination

The use case terminates when the user has successfully logged in

Post condition

The system presents the homepage

3. To have user goals saving to a remote database

This is a use case for saving user's bucket list goals to a remote database. It is the main functional requirement.

Use Case

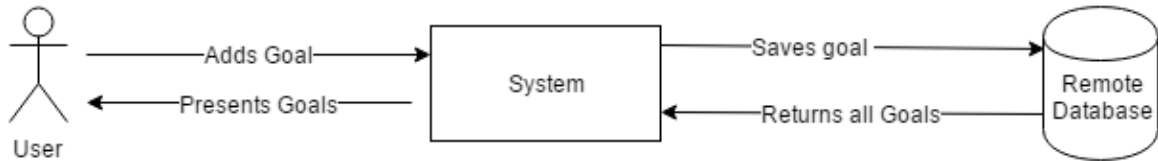
Scope

To save user goals to a remote database.

Description

This use case describes how the system will save and retrieve the items from the database.

Use Case Diagram



Flow Description

Precondition

The user is in the 'My Bucket' section of the app.

Activation

The use case starts when the user attempts to save an item.

Main flow

1. The user enters an goal into the text field
2. The user clicks "Add Goal"
3. The system sends the goal to the PHP files
4. The goal is then saved in a remote database

Alternate flow

A1 : <To see goals >

1. The user clicks "View all goals"
2. The system sends a query to the database to return all the user's goals
3. The system displays the goals

A2: <Connection Lost>

1. User clicks either "Add Goal" or "View all goals"
2. The system cannot connect to the database
3. The system presents an error message to the user

Termination

The use case is terminated when the user exits the feature

Post condition

The system loads the feature the user selected.

4. To present nearby activities using google maps

An important requirement as it is one of the things that differentiates the app from its competition.

Use Case

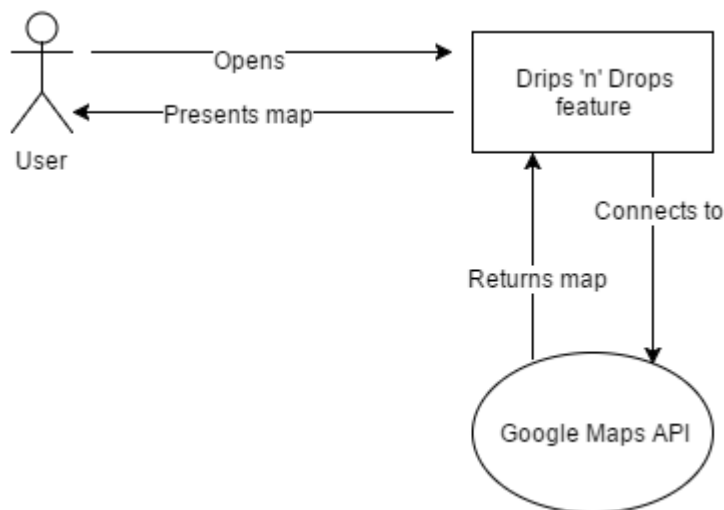
Scope

To find the users location and present nearby activities

Description

This use case will show how the user will be presented with a map of activities in proximity to their current location.

Use Case Diagram



Flow Description

Precondition

The user is logged into the app

Activation

The use case starts when the user opens the 'drips 'n' drops' page

Main flow

1. The user opens the feature
2. The system connects to the Google Maps API
3. The API returns the users current location and displays it
4. The pre-set markers showing nearby activities are presented to the user

Termination

The use case is terminated when the user exits the feature

Post condition

The system loads the feature the user selected

5. Using the search feature of the Twitter API

This is an important requirement as it will help to inspire the user.

Use Case

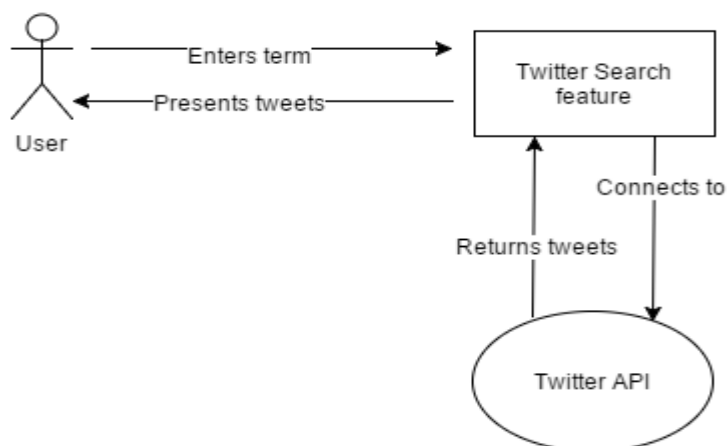
Scope

To allow users to avail of the Twitter search feature from within the app

Description

This use describes how the user will be able to use the Twitter search to research ideas, within the app

Use Case Diagram



Flow Description

Precondition

The user is logged into the app

Activation

The use case starts when the user opens the 'Twitter Search' feature.

Main flow

1. The user opens the twitter search
2. The system will ask for authorization to access the user's twitter account
3. The system loads a selection of recent public tweets using a pre-set term, such as "#MountEverest"
4. The user enters their own term into the search bar
5. The system presents a selection of recent public tweets using the term the user entered

Termination

The use case is terminated when the user exits the feature

Post condition

The system loads the feature the user selected

6. Pulling lists of ideas from a remote database

This is an important requirement as it will help to inspire the user.

Use Case

Scope

To pull lists of ideas from a remote database

Description

This use case describes how the user will be presented with constantly updated lists of ideas, separated into different categories

Use Case Diagram



Flow Description

Precondition

The user is logged into the app

Activation

The use case starts when the user opens the 'Ideas' feature.

Main flow

1. The user selects a category, such as "Thrill Seekers"
2. The system queries the database for the appropriate list
3. The system presents the list

Termination

The use case is terminated when the user exits the feature

Post condition

The system loads the feature the user selected

2.1.2 Data requirements

2.1.3 User requirements

The user will need the following requirements in order to avail of the services provided by the app.

An Android Phone The app is being developed in Android Studio and will only be available to those with an Android mobile phone.

Play Store The user will need to have access to the play store application in order to download the app

Internet Access The user will need internet access to download the application from the play store. Having constant internet access will also maximise user satisfaction from the app, due to the app needing to send and receive data from a database.

2.1.4 Environmental requirements

Windows The application will be developed using the college machines and my own laptop which use the windows operating system.

Multiple Android phones I will need to test the application on several different devices to ensure it is working properly

Android Studio The app will be developed in Android Studio

Android SDK This development kit provides the tools necessary to complete the app

SQL Database This will be used to store the bulk of the user's data

Internet Access This will be required to set up the database and download various android plug-ins

Twitter API This will be required for the 'Twitter Search' feature

Google Maps API This will be required for the 'Drips 'n' Drops' feature

2.1.5 Usability requirements

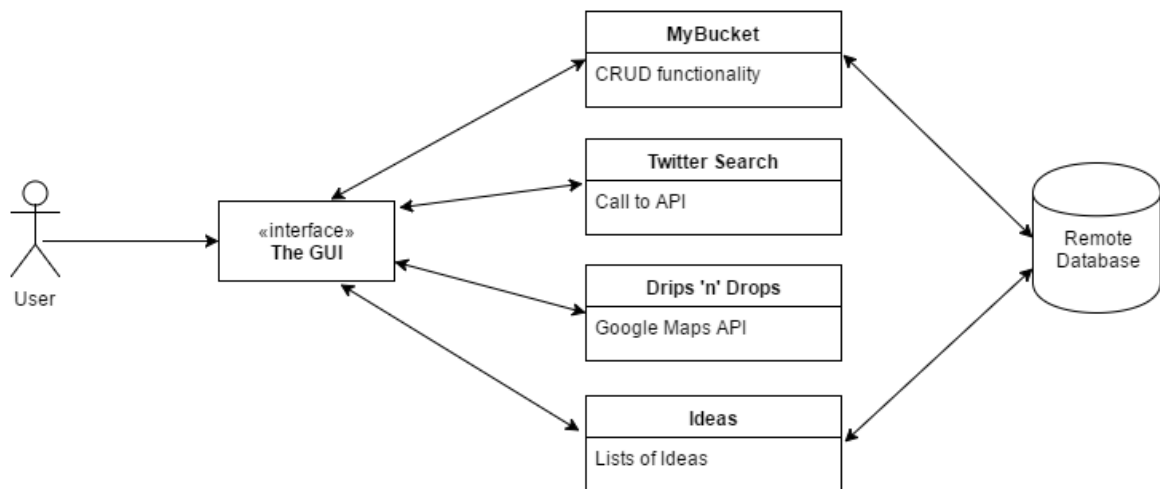
Easy to understand Due to a customer base that could include people who are unfamiliar with smart phones, the GUI must be very simple and easily understood

Attractiveness The layout of the application should be appealing. The screen should be bright and colourful.

Learnability The app's functions must be easy to learn and the help given to users must be as clear as possible.

2.2 Design and Architecture

The application has a simple layout where the user is met with a homepage having logged in. There are buttons that link to each aspect of the app such as 'my bucket', which will lead the user to their personal list of goals. I will include screenshots of the GUI in this document to better show how I will design the app.



The user interacts with a GUI that has many different sections or classes. Some of the classes then connect to databases. The main SQL database will hold all data including the user's login credentials as well as their bucket list items and the suggestions lists. It will let users add and delete items.

2.3 Implementation

2.3.1 Login and Register

To allow the user to login I set up a series of PHP and Java files with the help of some online tutorials ("Android Login/Register App With Mysql-01- Meet The Application"). I set up an xml layout that prompts the user to enter their username and password. The corresponding Java file passes these values to a PHP file in cloud 9 to be authenticated ("Sdarcy06"). The PHP file checks the database to see, first that the username exists, and second that the password matches it before returning either a success or failure message to the Java file in Android Studio. Here is the query the PHP file uses;

```
$query = "SELECT * FROM `userinfo` WHERE `username`='".$username.'" and  
`pass`='".$pass.'";
```

The Java file then either displays an error message or opens the main activity of the app. If the user has no login details they can register by entering a username and password and clicking the register button. This time the system expects the user to enter the password twice to ensure it is what they want it to be, and then checks the database to make sure the username does not already exist before registering the user. If the username doesn't exist and the passwords match, it enters the user's credentials into the database with the following query;

```
$query = "SELECT * FROM `userinfo` WHERE `username`='".$username.'" and  
`pass`='".$pass.'";
```

The system then returns the user to the login page.

2.3.2 The User's Bucket List

The user can add goals to their own bucket list, update those items or delete them if they see fit. For each of these features I have a Java method that connects a PHP file in cloud 9 to interact with the database, using online tutorials (Khan). For

example to add a goal, the user enters a title and description of the goal and clicks 'Add Goal'. The java method saves the entered values into a hash map and sends them to the PHP files like so

```
HashMap<String,String> params = new HashMap<>();
params.put(listConfig.KEY_GOAL_GOAL, goal);
params.put(listConfig.KEY_GOAL_DESC, desc);
RequestHandler rh = new RequestHandler();
String res = rh.sendPostRequest(listConfig.URL_ADD, params);
return res;
```

The Request Handler class opens a connection to the PHP files. The PHP files then inserts the values into the database. The user can update their goals in the same way. To delete a goal, I have a deleteGoal Java method that passes the goal's Id to a PHP file which then uses the following query to delete the goal in the database that has the corresponding Id.

```
DELETE FROM items WHERE id=$id;
```

2.3.3 The Twitter Search

For the search feature the app utilizes the Twitter API. I followed a set of tutorials to complete this section of the App ("E-04 Part2 Twitter Fabric Sdk Searching For Twitter Feeds"). The user must enter their Twitter login credentials in order to authorize the app to access Twitter. The app will then load the 20 most recent tweets that use a particular term. The term is set to "#Everest" originally, but there is a search bar that allows the user to search any term they want. The app gets the term the user entered and passes it to the Twitter API. The API then returns a list of recent tweets that include the term the user searched like so;

```
final List<Tweet> tweets = searchResult.data.tweets;
adapter.getTweets().addAll(tweets);
adapter.notifyDataSetChanged();
```

2.3.4 The Idea Lists

When the user enters the Ideas feature they are met with three category buttons. Each button starts an intent that opens a new activity. When the user starts the

activity the Java class pulls the relevant list from the database using a PHP file with the following query;

```
SELECT * FROM tablename
```

The java class then takes the returned data and displays it in a ListView like so;

```
ArrayList<HashMap<String, String>> list = new
ArrayList<HashMap<String, String>>();

JSONObject jo = result.getJSONObject(i);
String idea = jo.getString(listConfig.TAG_JSON_THRILLIDEA);
String desc = jo.getString(listConfig.TAG_JSON_THRILLDESC);

HashMap<String, String> ideas = new HashMap<>();
ideas.put(listConfig.TAG_JSON_THRILLIDEA, idea);
ideas.put(listConfig.TAG_JSON_THRILLDESC, desc);

list.add(ideas);
```

2.3.5 The Drips ‘n’ Drops

When the user opens the ‘drips ‘n’ drops’ feature they are presented with a map that picks up their location. It then displays a set of markers to locations of nearby activities, using online tutorials (Trebilcox-Ruiz). Using the LocationServices the user’s location is picked up, like so;

```
mCurrentLocation = LocationServices
    .FusedLocationApi
    .getLastLocation( mGoogleApiClient );
```

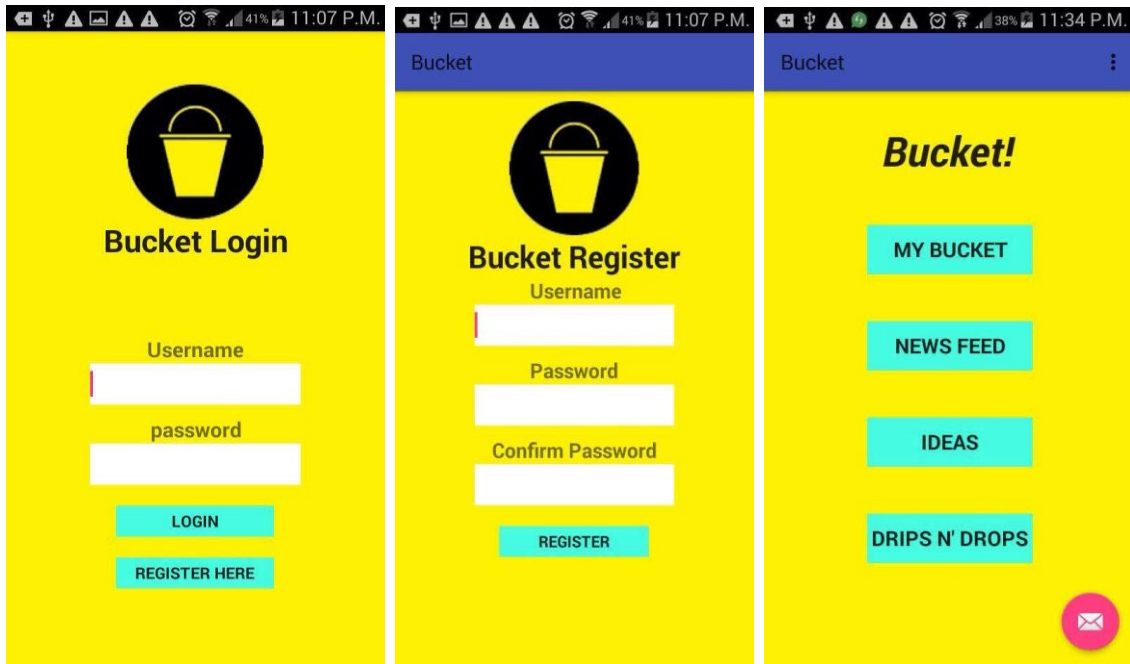
The map has a set of hard coded markers like so;

```
private static final LatLng Pavilions = new LatLng(53.454439, -
6.219055);

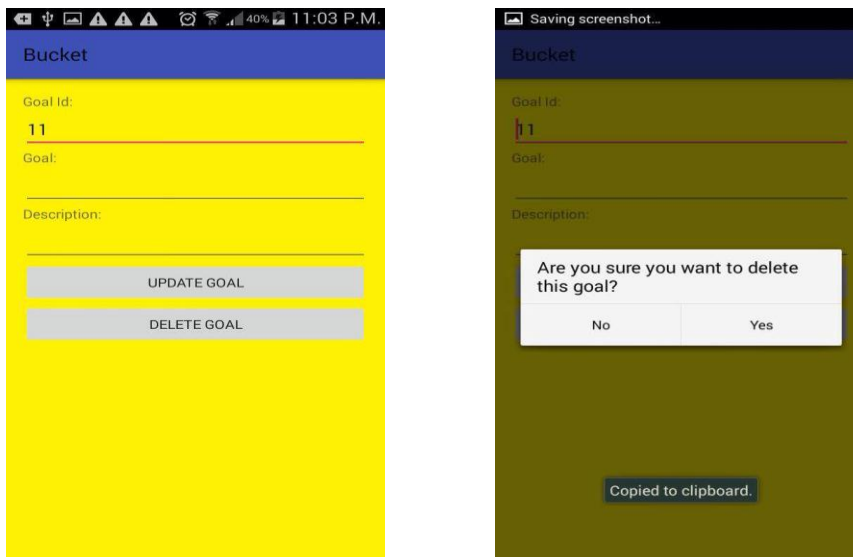
bucketMap.addMarker(new
MarkerOptions().position(Pavilions).title("this is a marker"));
bucketMap.moveCamera(CameraUpdateFactory.newLatLngZoom(Pavilions,
13));
```

2.4 Graphical User Interface (GUI) Layout

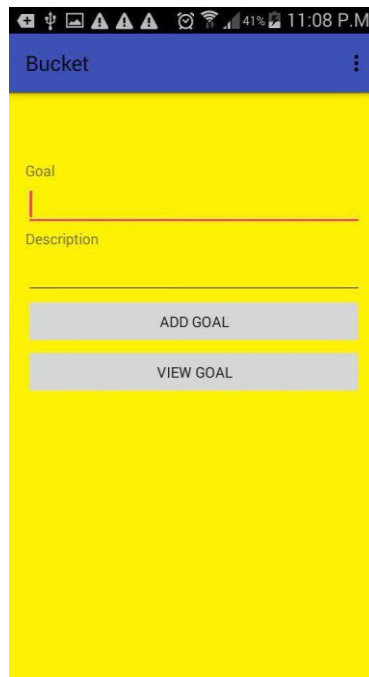
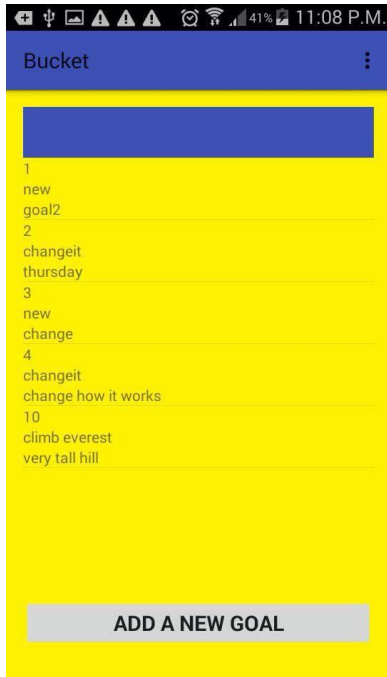
The Login, Register and Home Page and pages



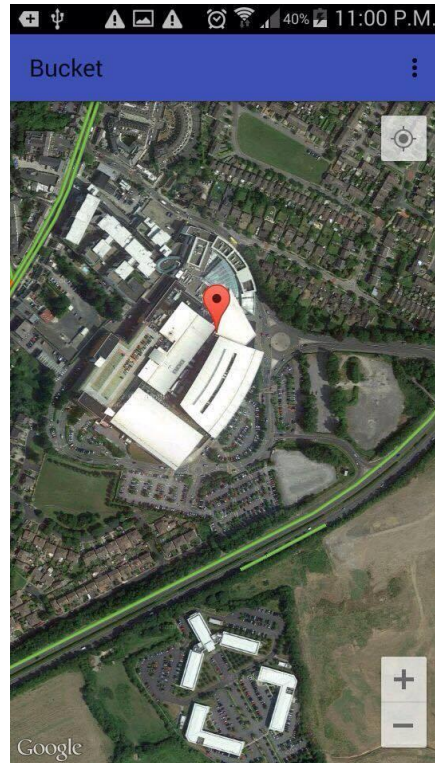
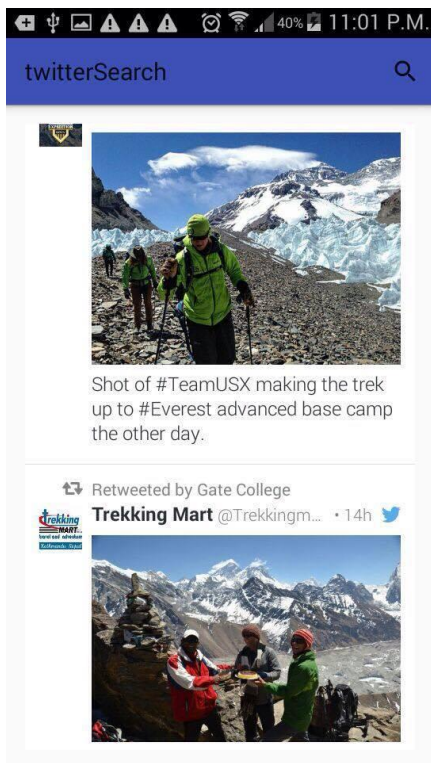
The User's Bucket – Update or Delete



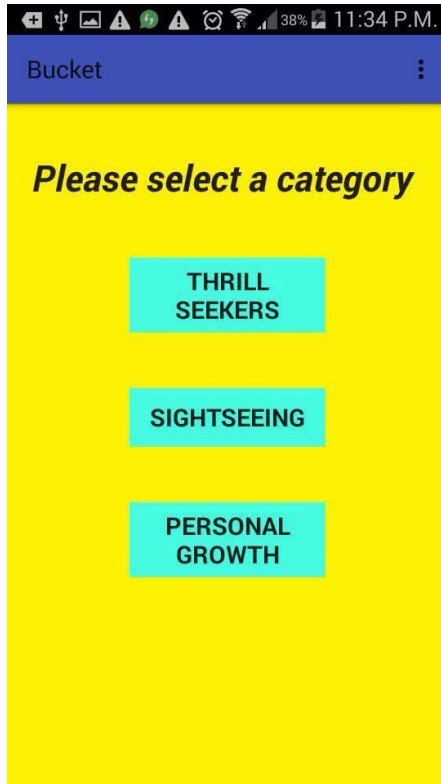
The User's Bucket – View and Add



The Google and Twitter API



The Idea Features



3 Testing

To test the app I used three separate methods. Firstly I asked 3 men with an understanding of computers to test each section of the app individually to assess how well it performs. I then ran several Monkey tests.

Lastly I gave the app to a variety of potential customers, of different age groups, and asked them to spend some time using the app and then fill out a short survey.

3.1 Unit Testing

The three men I asked to perform these tests are studying a computing course in DCU.

3.1.1 Login/Register

Each of the three found it easy to login and register. One of the men felt it took too long for the system to complete the login. This was down to a 5 second delay I had programmed into the login method, so it could display a “processing login” message. I shortened the delay to 2.5 seconds.

3.1.2 Navigation

Initially, in order to open a new feature, the user would need to return to the homepage first. All 3 of the men doing the testing felt this to be inconvenient. As a result I added a dropdown menu in the toolbar at the top of each screen that allows users to directly open any page they want.

3.1.3 Saving, updating and deleting goals

Each user saved, updated and deleted five goals and felt the app functioned quickly and correctly so I took no further action.

3.1.4 Using the Twitter Search

Upon testing the twitter search feature, two of the men were satisfied with how well it performed. One felt displaying the 10 most recent results was not sufficient, so I decided to double it to the 20 most recent.

3.1.5 Using the Google Maps

In the initial version, the 'drips 'n' drops' feature that incorporates the Google Maps API, did not have the capability of picking up the users current location. Instead it was set to a certain place and allowed users to enter their location. The men testing the feature each remarked that they would be more inclined to use it if they didn't have to enter their location. I spent some time over the following days working on it and managed to alter it so it does pick up the user's location.

3.1.6 Viewing the idea lists

The final feature I asked the men to test was the 'Ideas' page. It consisted of a list of ideas to possibly be added to user's bucket or to be searched using the Twitter search.. This is a fairly straight forward function and therefore the only suggestion the men had was it would be nice if I split the list into different categories, which I have done.

3.2 Monkey Testing

Monkey testing is a software testing technique in which the testing is performed on the system under test randomly. The Input data that is used to test is also generated randomly and keyed into the system ("Monkey Testing"). I ran this test three times. First I had it perform 500 tasks, then 1000, then 2000. The app withstood all three tests without crashing showing its strength against stress testing, as shown in Figure 7 in the appendix.

3.3 Customer Testing

Having received some helpful feedback from the unit testing, I moved on to customer testing. I selected a group of potential customers to test the app as shown in Figure 1 in the Appendix. With the app being aimed at both young and elderly consumers, I showed the app to four people over the age of 50, three between 30 and 50 years old and five between 16 and 30 years old.

Each person spent 15 minutes using the app and seeing what it has to offer and then filled out a short survey using an online survey site ("SurveyMonkey: Free Online Survey Software & Questionnaire Tool"). The results of which show the following;

- Every person who tested the app found it useful and would install and use it should it become available. Shown in figure 2 and figure 5 in the Appendix.
- On average, over 50% strongly agreed that the app was easy to navigate, easy to understand and had an attractive UI. None of the 12 who tested the app disagreed with any of the above statements. Shown in figure 3 in the Appendix
- 100% of those who tested the app felt it adequately achieved its four main functions, shown in figure 4 in the Appendix
- No potential customer felt there was anything the app was missing in order to make it worth using, with one person commenting "The app is excellent, I would happily use it again". Shown in figure 6 in the Appendix.

4 Further development or research

There are many ways this app could expand beyond the scope of this project. For one, I am developing the application to be available to those with android phones. Expanding the app to accommodate IOS users would greatly broaden the potential customer base.

Additionally, I will be including a section in the app named 'drips n' drops' that will allow users to see nearby activities. This will be dependent on a list of coordinates. This list can always get bigger to include more and more places and offer users an increased amount of options.

Moving even further into future possibilities, the app could include its own section where users can book flights, hotels or tours to help them realize some of the dreams they have included in their lists.

5 References

1. Khan, Belal. "Android Mysql Tutorial To Perform Basic CRUD Operation". *Simplified Coding*. N.p., 2015. Web. 9 May 2016.
2. Trebilcox-Ruiz, Paul. "Getting Started With Google Maps For Android: Basics". *Code Envato Tuts+*. N.p., 2015. Web. 9 May 2016.
3. "E-04 Part2 Twitter Fabric Sdk Searching For Twitter Feeds". *YouTube*. N.p., 2016. Web. 9 May 2016.
4. "Android Login/Register App With Mysql-01- Meet The Application". *YouTube*. N.p., 2016. Web. 9 May 2016.
5. "SurveyMonkey: Free Online Survey Software & Questionnaire Tool". *SurveyMonkey.com*. N.p., 2016. Web. 9 May 2016.
6. "Sdarcy06". *C9.io*. N.p., 2016. Web. 9 May 2016.
7. "Flowchart Maker & Online Diagram Software". *Draw.io*. N.p., 2016. Web. 9 May 2016.
8. "Monkey Testing". *www.tutorialspoint.com*. N.p., 2016. Web. 10 May 2016.

6 Appendix

6.1 Project Proposal

Project Proposal

Bucket List App

Sean D'Arcy, x11371436, sdarcy06@hotmail.com

BSc(Hons) in Computing

Networking and Mobile Technologies

21/09/15

1. Objectives

The objective of my software project is to build a mobile application for android called Bucket List. I will aim to give it the following functionality;

- Allow users to register and log in
- Have a homepage with buttons linking to each section of the app
- A "my Bucket" button which contains the users personal bucket list. The list will have all the things the user has accomplished, and when he/she clicks one, for example "Visited Niagara Falls", there will be all the pictures the user uploaded to match the accomplishment. It will also contain the things the user still wants to do.

- An ideas button that contains a wide array of different things to add to your own list, from smaller things like “watch the sun set” to giants like “Climb Everest”
- I will attempt to have a newsfeed where users can see each other’s lists, pictures etc.
- Lastly I will try to have a button with a name like “mini bucket” with smaller scale, more everyday activities like what good movies are out, where good deals on bowling or in bars are on certain nights, because you can’t go to college or work until 5 on a Wednesday, climb Everest, and be back in time for dinner!
- I will attempt to have the app linked up to a database that will store all bucket list entries

2. Background

I have been fascinated by the idea of a bucket list ever since I went to see the movie starring Morgan Freeman and Jack Nicholson eight years ago. Ever since I have kept track of the things I have done to occasionally look back over. It has pushed me to go places and do things that I believe I would not have done otherwise, such as spending summers in America and Canada respectively, going to Niagara Falls, road tripping from San Francisco down highway 1 to San Diego, going gorge diving etc.

I find it to be very rewarding to be able to reminisce over some of the things you have done and places you have seen. So too is it exciting to dream up other items you could tick off your list in the future. My one problem would be that, while everyone has hopes and dreams, very few have taken the time to plan exactly what they would like to do. Therein lies the reason for me wanting to build this application. It will allow users to constantly be presented with ideas and inspired to try new things, as well as having a detailed list (with pictures if you took some) of all the user’s achievements.

I think it would make for a very interesting application and I am excited to begin work on it.

3. Technical Approach

From some of the research I have done I believe there will be plenty of information to put into my app as there are dozens of existing websites with tips on what to have on your bucket list. I will attempt to code the application in Java using Eclipse. The application will need to be linked to a database that stores all the users log in information, their bucket list items as well as things like different ideas. I will attempt to use plug-ins and tutorials to enable the application to have an operational news feed and allow users to purchase things like cinema tickets.

4. Special Resources required

I don't think I will need any external equipment. I may need to use tutorials and possibly read books in order to effectively accomplish the goals I have set myself.

5. Technical Details

I will be using Java for the most part while coding the application. I will be using the Eclipse IDE for this project. There are a series of Android Support Libraries that according to the android developers website "that provide backward-compatible versions of Android framework APIs as well as features that are only available through the library APIs."

I have never attempted to code an android application before so much of this is new to me but if this framework isn't working for me I have experience using the bootstrap framework and will attempt to use that instead.

There are tutorials on android hive that detail how to go about coding a news feed for android applications. I will use these for my applications.

6. Evaluation

To evaluate how this application works when completed I will try a series of different tests like having several people attempt to login, save their list, and test its functionality at the same time. I will then ask them to complete short surveys to see what they did and did not like, or comment on what they felt worked well and what didn't.

I will ask them to rank it from 1-5 in categories like scalability, how user friendly it is, how quick it was, entertainment level etc.

I will try to get a good mixture of people with developing experience and those without it for my testing to get the best results before I complete the project.

Signature

Sean D'Arcy

X11371436

6.2 Customer Survey

For the purposes of testing I asked several individuals to spend some time using the app and then to fill out a survey. The results were as follows;

Question 1.

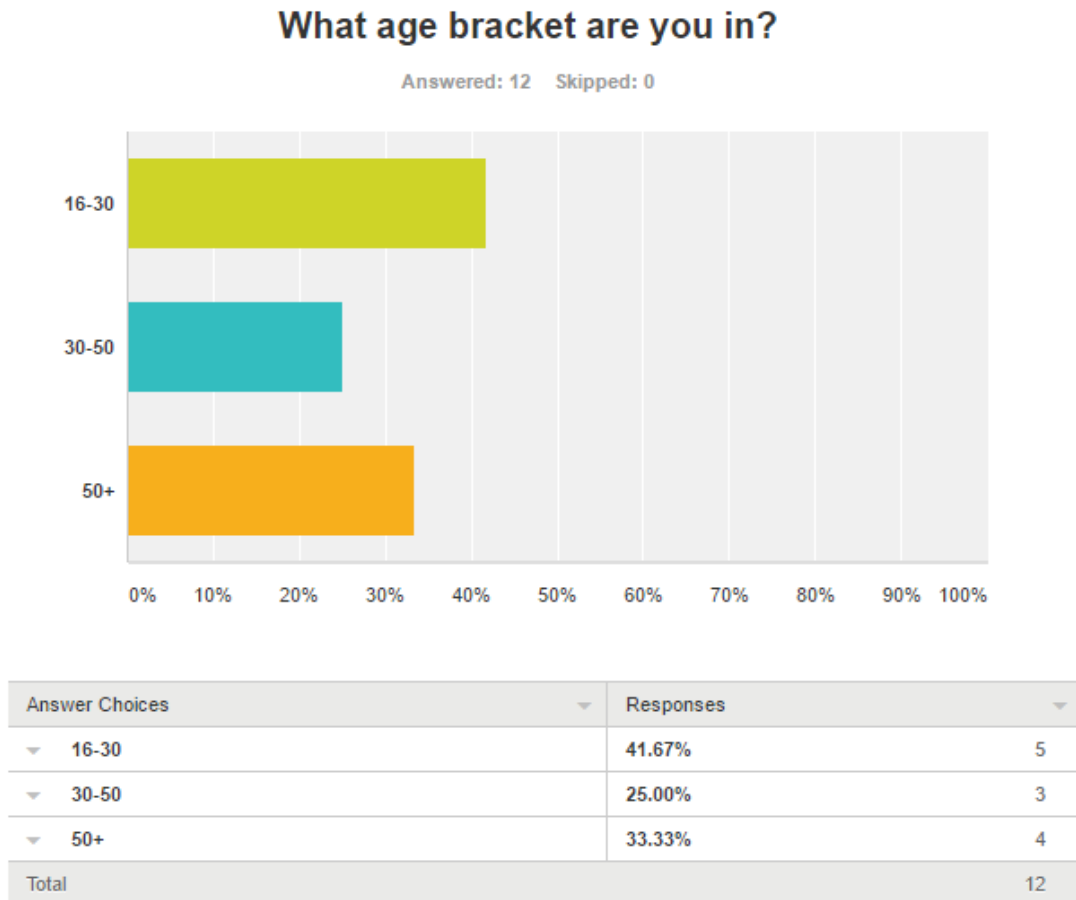
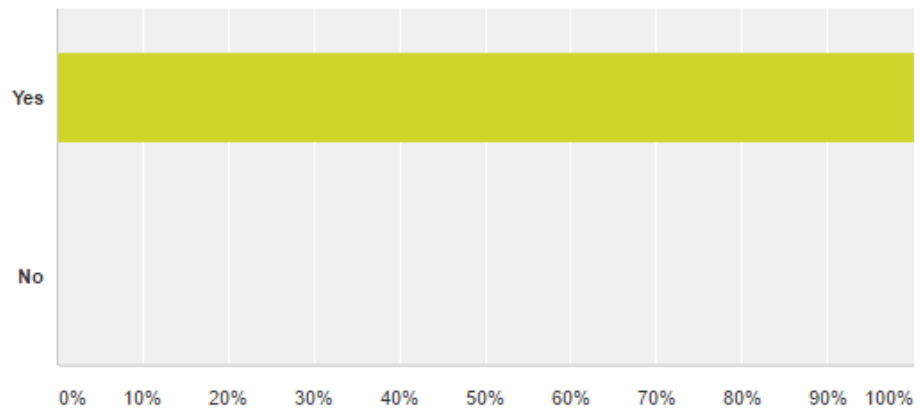


Figure 1

Did you find the application useful?

Answered: 12 Skipped: 0



Answer Choices	Responses
Yes	100.00% 12
No	0.00% 0
Total	12

Figure 2

Please rate the following criteria

Answered: 12 Skipped: 0

	Strongly Agree	Agree	Disagree	Strongly Disagree	Total	Weighted Average
The app executed tasks quickly	58.33% 7	41.67% 5	0.00% 0	0.00% 0	12	1.42
The app was easy to navigate and understand	58.33% 7	41.67% 5	0.00% 0	0.00% 0	12	1.42
The app had an attractive layout	50.00% 6	50.00% 6	0.00% 0	0.00% 0	12	1.50

Figure 3

Do you feel the app adequately...

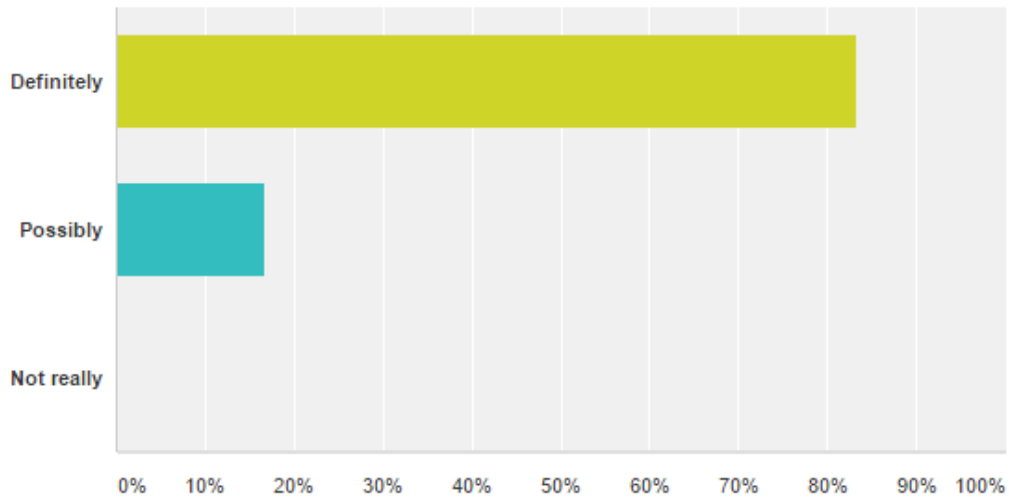
Answered: 12 Skipped: 0

	Yes	No	Total	Weighted Average
Saved and retrieved your bucket list items	100.00% 12	0.00% 0	12	1.00
Presented you with ideas as to what to add to your list	100.00% 12	0.00% 0	12	1.00
Allowed you to learn more about some of its suggestions	100.00% 12	0.00% 0	12	1.00
Made you aware of any nearby activities that you can take part in	100.00% 12	0.00% 0	12	1.00

Figure 4

Would you be likely to use this application?

Answered: 12 Skipped: 0



Answer Choices	Responses	
▼ Definitely	83.33%	10
▼ Possibly	16.67%	2
▼ Not really	0.00%	0
Total		12

Figure 5

Is there anything you feel the application is missing?

Answered: 6 Skipped: 6

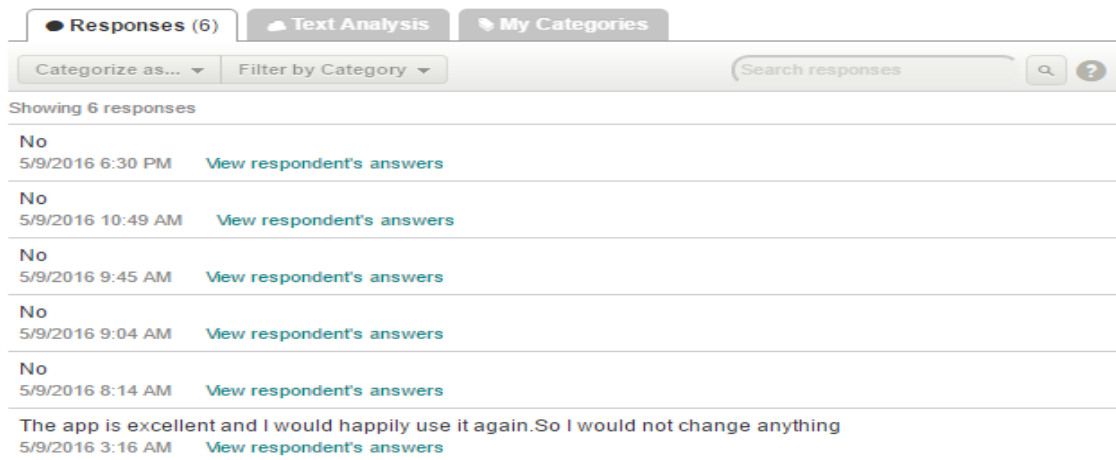


Figure 6

6.3 Project Plan

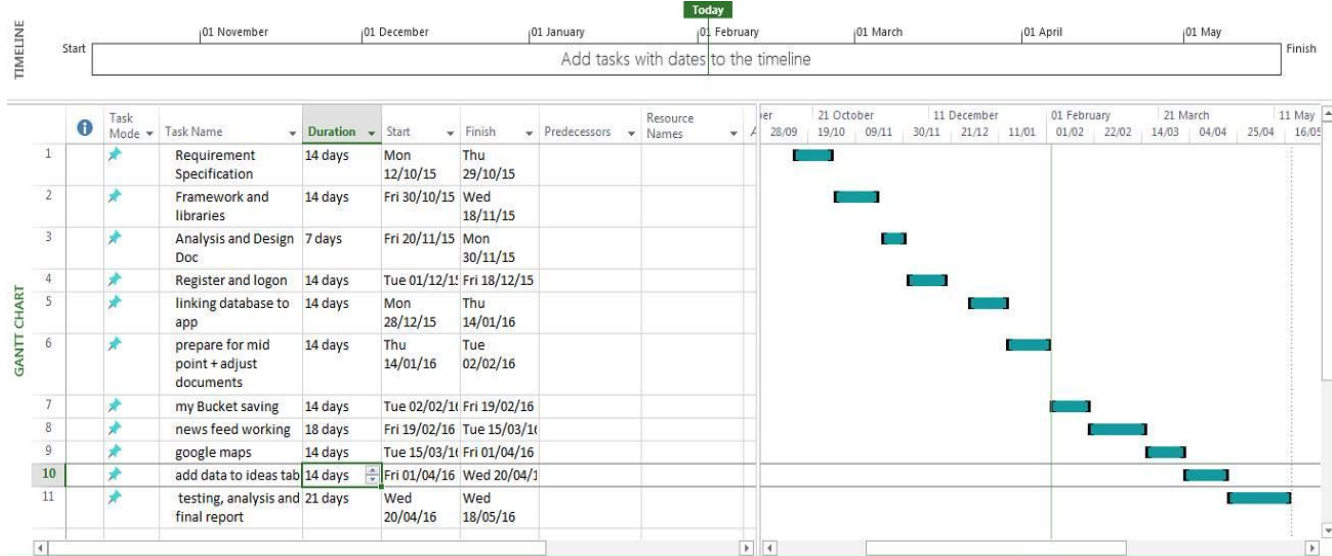


Figure 7

```

:Sending Touch (ACTION_UP): 0:(922.4735,384.54367)
:Sending Trackball (ACTION_MOVE): 0:(-4.0,1.0)
:Sending Touch (ACTION_DOWN): 0:(845.0,355.0)
:Sending Touch (ACTION_UP): 0:(834.1854,368.65674)
:Sending Flip keyboardOpen=false
:Sending Touch (ACTION_DOWN): 0:(366.0,39.0)
:Sending Touch (ACTION_UP): 0:(370.29126,27.79942)
:Sending Touch (ACTION_DOWN): 0:(1163.0,582.0)
:Sending Touch (ACTION_UP): 0:(1163.7253,554.87885)
:Sending Touch (ACTION_DOWN): 0:(1099.0,513.0)
:Sending Touch (ACTION_UP): 0:(1101.5159,512.61194)
:Sending Touch (ACTION_DOWN): 0:(268.0,760.0)
:Sending Touch (ACTION_UP): 0:(265.37912,762.5319)
:Sending Touch (ACTION_DOWN): 0:(1243.0,45.0)
:Sending Touch (ACTION_UP): 0:(1227.1942,25.70639)
:Sending Flip keyboardOpen=true
:Sending Touch (ACTION_DOWN): 0:(800.0,22.0)
Events injected: 2000
:Sending rotation degree=0, persist=false
:Dropped: keys=4 pointers=10 trackballs=0 flips=0 rotations=0
## Network stats: elapsed time=14180ms (0ms mobile, 0ms wifi, 14180ms not connected)
// Monkey finished

```

Figure 8

6.4 Monthly Journals

Reflective Journal September

Student name: Sean D’Arcy X11371436

Programme: BSc in Computing

Month: September

My Achievements

This is the first reflective journal and therefore the very beginning of my software project. As a result my achievements have been limited to actually coming up with what I felt was a viable idea and then expanding on it enough to create a project proposal. I completed the project proposal and submitted it. I included in the proposal a Gant chart which gave me my first opportunity to give a higher level time scale to the project.

My Reflection

I believe it is a strong idea and was happy with how the proposal and chart turned out. I am now waiting to be assigned a supervisor to give me some feedback on whether or not this is something I can continue with.

Having completed and seemingly submitted the proposal with plenty of time to spare I started today, Monday the 5th of October, in nightmarish fashion when I realised I had not completed the submission process, having only actually submitted a draft. I only notice now because I have been away for a week, the very reason I submitted so early.

Intended Changes

So one change rings through as clear winner for this entry, which is to make sure to double check all submissions!! Hopefully this will prevent any further Monday morning panic attacks!

Supervisor Meetings

I have not been assigned a supervisor yet.

Reflective Journal October

Student name: Sean D Arcy X11371436

Programme: BSc in Computing

Month: October

My Achievements

This month I have been primarily working on the requirements specification for the application and drilling down into exactly what I want my App to be able to accomplish. I have over two thirds of the document and intend to send it to my supervisor before submitting, just to see if he feels it is a high enough standard of work. I have also began to think about what parts of my app will need a database and how exactly I am going to go about coding the App

My Reflection

The document looked fairly straight forward upon first inspection but once I delved into it I found it to be more demanding than I had thought. Again, not leaving it to the last couple of days has been the saving grace as it allows me to not only do it well but to have it checked before submitting. I have found that alongside many helpful online resources, pluralsight will probably be my main point of research when trying to code this app.

Intended Changes

I have been wrong before but I feel this month has going reasonably well and therefore don't see any outstanding changes I would make should I be asked to start the month again.

Supervisor Meetings

I have been assigned Paul Hayes. We met on Tuesday the 24th of October to discuss the project proposal. Paul gave me some helpful tips on how to go about the requirement specification and highlighted parts of the App that may need reviewing such as whether or not to use an RSS news feed.

Reflective Journal November

Student name: Sean D'Arcy x11371436

Programme: BSc in Computing

Month: November

My Achievements

This month I began the process of creating the login and register page and have it saving to a database. I have been using PhpMyAdmin and PHP files as of yet but I have recently been thinking that I should have some sort of framework for the backend of the project. I have no experience with using frameworks on the backend so this will be a first, like most things I'll have to do to finish this app. I also submitted the analysis and design document. Couldn't have been more confused, there were no examples for how to do it. But I've been assured I can change it before final submission.

My Reflection

All in all I feel the month has gone well. The last document has been submitted so I can focus entirely on coding the app now. Beginning the process is tricky as it's tough to know where to start,

Intended Changes

I don't think I would change much about what I did this month.

Supervisor Meetings

I spoke over the phone with my supervisor today. He gave me some helpful tips and even set me up to meet Michael Bradford, who is very well versed in backend frameworks, to point me in the right direction.

Reflective Journal December

Student name: Sean D Arcy X11371436

Programme: BSc in Computing

Month: December

My Achievements

The month of December consisted of a mixture of studying for exams and enjoying the Christmas festivities. Therefore my software project got put on the back burner until the exams were finished.

My Reflection

Due to the impending exams, the project was delayed to such an extent that even this monthly Journal is late. In hindsight I should have dedicated some time to the project instead of focusing solely on exams.

Intended Changes

I would not ignore the project for exams.

Reflective Journal January

Student name: Sean D Arcy X11371436

Programme: BSc in Computing

Month: January

My Achievements

Since completing the exams the software project has been my main focus. I have been working on getting the login and register system functioning in cloud 9 ("Sdarcy06"), with the usage of the slim framework. It will be linked to a SQL database to store the user's login credentials. I have also been focusing on completing all documents.

My Reflection

This month has been productive but has highlighted the fact that some of the things I intend to include in my project may be quite difficult, given my lack of experience with some of the technologies to be used.

Intended Changes

I am happy with the progress of this month and therefore would not change much about it.

Supervisor Meetings

I have emailed my supervisor about meeting before the mid-point presentations and am awaiting a reply

Reflective Journal February

Student name: Sean D'Arcy x11371436

Programme: BSc in Computing

Month: February

My Achievements

This month marks the start of actual functional code in the app. With the documents and mid point presentation completed, the emphasis is now fully on fulfilling the requirements. I now have the pages of the app linking together and have begun work on having items being added to the users bucket list.

My Reflection

Android Studio is a nightmare to set up. I had not anticipated having to spend so much time getting the emulator running properly. But the wheels are turning and I feel good about where I am in the project.

Intended Changes

Get Android Studio set up earlier. It wasn't as easy as expected.

Supervisor Meetings

I met with Paul to get some feed back on the mid point presentation and received some good advice on what to do now.

Reflective Journal March

Student name: Sean D'Arcy

Programme: BSc in Computing

Month: March

My Achievements

This month has been flooded with CA's that need attention from every module, so I couldn't do too much on the software project. Having said that, I have completed the login and register, and I have all the pages linked together. The homepage design has been completed and I have started the 'my bucket' page.

My Reflection

It was a busy month so I don't feel I could have done things differently.

Intended Changes

In this next month we have our exams which will take a lot of the focus. But I hope to continue with my project and have it close to completion by the end of april.