



Bournemouth  
University

## FACULTY OF SCIENCE & TECHNOLOGY

BSc (Hons) Software Engineering  
May 2018

Univent: Paperless Solution for Distributing Flyers

by

Derrick Aryeetey Feehi

Faculty of Science & Technology  
Department of Computing and Informatics  
Final Year Project

# Abstract

Events are mostly published via social media; however, the main method is by distributing flyers on campus and stand advertisement. These leaflets are disregarded as they always end up in the bin, they also lack long term impact as many may read them and discard or misplace them, ending up with nothing to remind them of the event.

Bournemouth University currently has more than 17000 students enrolled, most of these students participate and organise events every single day. Events such as seminars, talks, parties, sports etc.

There is a need for an event application to help people publish events and keep track with any upcoming events.

The app described and analysed in this thesis will address this need and assist event organisers or those interested in going to such event. This app will aid publishing the event on a platform where all student on faculty will have access to. The app will also provide attendees with all relevant information pertaining to the events they are interested in.

The app stores users name of all those going to the event and creates a total number of those that have accepted the event, so the organisers are able to prepare accordingly. The fact that other students viewing the event can see who is attending, if students were undecided as to whether to attend, seeing their friends are attending might convince them in going.

Mobile Event apps are very common now, due to the increase of smartphones, therefore this app is very important as it provide easy access to all information that are usually required regarding an event.

This thesis describes the design, implementation of an Android app with a server side and describes the structure and what methodology and language is needed. In addition, it will also describe how this app communicates with the server database through the API and any potential security risk. Currently this app will be for Android platform only. This app will be communicating constantly with a database server, MYSQL through an API written in PHP running on a webhost.

By providing all the benefit, and useful features of the application, it should show how essential it is to have a unique mobile platform for universities where events can be posted, minimizing effort and maximizing benefit to all attendees and event planners.

# **Dissertation Declaration**

I agree that, should the University wish to retain it for reference purposes, a copy of my dissertation may be held by Bournemouth University normally for a period of 3 academic years. I understand that once the retention period has expired my dissertation will be destroyed.

## **Confidentiality**

I confirm that this dissertation does not contain information of a commercial or confidential nature or include personal information other than that which would normally be in the public domain unless the relevant permissions have been obtained. In particular any information which identifies a particular individual's religious or political beliefs, information relating to their health, ethnicity, criminal history or sex life has been anonymised unless permission has been granted for its publication from the person to whom it relates.

## **Copyright**

The copyright for this dissertation remains with me.

## **Requests for Information**

I agree that this dissertation may be made available as the result of a request for information under the Freedom of Information Act.

### **Signed:**

Name: Derrick Aryeetey Feehi

Date: 10 May 2018

Programme: BSc (Hons) Software Engineering

# **Original Work Declaration**

This dissertation and the project that it is based on are my own work, except where stated, in accordance with University regulations.

**Signed:**

# Acknowledgements

First and foremost, to my supervisor Deniz Centinkaya, for the support and the constant feedback.

I would also like to thank my incomparable aunties, Deborah Quartey and Magdalene Quartey for being there when needed. My sincere appreciation to my family who have been a major support throughout the entire project.

Finally I would like to thank those people who also helped me to make the project possible; Omolola Fagbule, Joshua Tubana, Gernot Liebchen and Raian Ali.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background & Context . . . . .	2
1.2	Problem Definition . . . . .	2
1.3	Solution . . . . .	3
1.3.1	Existing solution . . . . .	3
1.3.2	Proposed solution . . . . .	3
1.4	The Challenge . . . . .	4
1.5	Target Audience . . . . .	4
1.6	Project Aims . . . . .	5
1.7	Project Objective . . . . .	5
1.8	Risk . . . . .	5
1.9	Achievement . . . . .	8
1.10	Android vs iOS . . . . .	8
1.11	Success Criteria . . . . .	8
<b>2</b>	<b>Methodology</b>	<b>10</b>
2.1	Overview . . . . .	10
2.2	Agile . . . . .	10
2.2.1	XP . . . . .	10
2.2.2	Scrum . . . . .	11
2.3	Cowboy . . . . .	11
2.4	How Agile Helped Cowboy . . . . .	12
2.5	Project Plan . . . . .	12
<b>3</b>	<b>Requirements &amp; Analysis</b>	<b>13</b>
3.1	Overview . . . . .	13
3.2	Requirement & Analysis . . . . .	13
3.3	Application Vision . . . . .	15
<b>4</b>	<b>Background Study</b>	<b>16</b>
4.1	Overview . . . . .	16

4.1.1 OS . . . . .	16
4.1.2 Database . . . . .	16
4.2 Development Environment . . . . .	17
4.2.1 JDK . . . . .	17
4.2.2 SDK . . . . .	17
4.2.3 Android Emulator . . . . .	17
4.2.4 Graphic Layout . . . . .	17
4.2.5 IDE . . . . .	18
4.3 Android Framework and API . . . . .	18
4.4 Developing Problem . . . . .	18
<b>5 Literature Review</b>	<b>19</b>
5.1 Applying Gamification to Univent . . . . .	21
5.2 Related Work . . . . .	22
5.2.1 Fever . . . . .	22
5.2.2 EventBrite . . . . .	22
5.3 Summary . . . . .	23
<b>6 Univent Design</b>	<b>24</b>
6.1 Overview . . . . .	24
6.2 Android Application Design . . . . .	24
6.3 Initial Design . . . . .	26
6.4 Database Design . . . . .	27
6.4.1 Security . . . . .	27
6.4.2 General Architecture . . . . .	28
<b>7 Implementation</b>	<b>29</b>
7.1 Overview . . . . .	29
7.2 Login & Register . . . . .	30
7.3 Implementation of REST API . . . . .	31
7.3.1 Sending Request to the Server . . . . .	33
7.4 Sending Image to Server . . . . .	36
7.5 Testing . . . . .	36
7.5.1 System Testing . . . . .	36
7.5.2 Black Box Testing . . . . .	36
7.6 Publishing to Play Store . . . . .	36
<b>8 Evaluation</b>	<b>37</b>
8.1 Overview . . . . .	37
8.1.1 Requirements Evaluation . . . . .	38

8.2 Success Criteria . . . . .	39
<b>9 Conclusion</b>	<b>40</b>
9.1 Summary . . . . .	40
9.2 Future Works . . . . .	40
<b>Appendices</b>	<b>46</b>
A Screen Design . . . . .	47
B Android vs IOS . . . . .	53
C JSON Data Format . . . . .	54
D Test Suit . . . . .	55
E Android Framework & API . . . . .	60
E.1 Application Component . . . . .	60
E.2 Main Application Component . . . . .	60
E.3 Additional Components . . . . .	61
F Sending Image to Server . . . . .	62
G Project Proposal . . . . .	64
H Gantchart . . . . .	69
I Project Ethics . . . . .	70
J Disk Content . . . . .	76

# List of Figures

1.1	Risk Diagram	6
2.1	Agile Lifecycle	11
4.1	Android Emulator	17
4.2	Andriod UI Design Editor	18
6.1	Define a TextView in XML	24
6.2	Define a ImageView in XML	25
6.3	Defining RecyclerView	25
6.4	Home Screen	26
6.5	Database Design	27
6.6	Rest API	28
7.1	Logic Flow of Screen	29
7.2	Fab Menu	30
7.3	Declaring an Intent	30
7.4	Declaring SharedPreferences	31
7.5	SharedPreferences key value pair data	31
7.6	HTTP Methods	32
7.7	Internet Permission	33
7.8	URL to get all Events in the Database	33
7.9	URL to get all Events in the Database	33
7.10	Check if Connection is OK	34
7.11	Status Code	34
7.12	Status Code	34
7.13	Parsing JSON	35
7.14	PHP Page	35
1	Initial Home Screen Design	47
2	Final Home Screen Design	47
3	Initial Register & Login Screen Design	48

4	Final Register & Login Screen Design . . . . .	48
5	Menu Clicked Screen . . . . .	49
6	Add Event Screen . . . . .	49
7	Search Event Screen . . . . .	50
8	Profile Screen . . . . .	50
9	Upcoming, My Events & Choose Interest Screen . . . . .	51
10	Email Verification . . . . .	51
11	Account Verification Success . . . . .	51
12	Account Verification Error . . . . .	52
13	OS Global Market Share, (Data collected from Statista) . . . . .	53
14	JSON Format . . . . .	54
15	Activity Lifecycle . . . . .	61
16	Glide Usage . . . . .	62

# List of Tables

1.1	Likelihood and Impact . . . . .	6
1.2	Risk Analysis . . . . .	8
3.1	List of Requirements . . . . .	14
7.1	AsynTask Methods . . . . .	32
8.1	Survey . . . . .	37
8.2	User Comments . . . . .	38
8.3	Requirements Evaluation . . . . .	39

# **Chapter 1**

## **Introduction**

We live in a world where mobile development dominates, a world in which mobile phones are always with us to enhance our everyday lives, to give information, simplify tasks and provide entertainment. With the increase of mobile application called apps, individuals are seeking information on the go, this need allows developers to tailor make apps targeting exact needs of the users. Correlating this, Google play store and the iOS Store has more than 5 million apps (data from statista).

According to Statista 197 billion of mobile apps were downloaded in 2017, furthermore, 2.53 billion people use a smart phone, a report from Ofcom, confirms 55% of mobile owners uses their mobile phone to visit social networking sites or apps. Keeping this statistic in mind and with the increase of use of mobile application, it is not surprising to see an increase in the creation of app designed purposely for events.

Event apps offer information attendees needs, such as schedule, maps, date, time and more. Attendees download these apps so they can access information about any event they are interested in.

If we go back about 10 years ago, think how events where published, what comes to mind is a flyer. That old flyer, not designed properly, with no or less information printed on it and with all that typo(as once printed the organizer has no way of amending it). Yes exactly, those tedious flyers, that once received you will probably not even read and just bin.

Furthermore, if you are interested in that event, you will be handed that large paper with all critical information in it, address, date, event information and more, hoping it will not get lost in a mass of other papers.

Point is, mobile app illustrate the advancement society has made when it comes to tech, and for now this research is based on the development of user friendly Android based application named Univent. This event app will aid users in finding any upcoming events on campus and help event organizers to have a unique platform to publish their events and activities. This thesis will describe the design, development and evaluation of an app that will be used as a platform to publish any upcoming event in the University,

helping people to interactively engage in activities and events going on in campus.

## 1.1 Background & Context

Based on observation, students find it quite difficult to advertise upcoming events in University, with the only way being distribution of flyers around and most of these ending up in bins. It is quite difficult as well if the target is to reach as many students as possible within a short space of time. Flyers normally lack long term impact as many students may read the leaflet and then discard them. Most students that receive flyers will look at it once and if not interested will bin the leaflet as it's not worth keeping. Therefore, they might not have anything to remind them of the event or service being advertised.

Furthermore, we live in a generation which likes to get information about activities on campus very easily, without carrying around flyers of any sort.

Moreover, the use of flyers affects the target, in that, to reach as many people as possible, for instance there might be University students who are interested in a particular event, but since they were not there at that moment of distributing the flyer they will not have the opportunity to know about the event. This will mean the individual will be unable to participate and they will not gain the benefit that event offered, this miss out on the opportunity of meeting new people for example.

## 1.2 Problem Definition

Whilst the 21st century digital world offers modern techniques, that can reduce cost, unfortunately flyers and leaflet distribution is still popular in Bournemouth University(BU). This form of advertisement, also known as leafleting, though may have quite an advantage because it's easy to read, as the design of a leaflet tend to have larger letters and a limited number of words, as a fact they are designed predominantly to attract attention, not to go into too much detail.

**Limited Target:** the number of people reached is limited as this depends on the number of flyers that can physically be delivered.

Flyers have a trend to become obsolete, as people often order large quantities because higher quantity equals lower cost and there is a tendency of printing more material than needed for this reason. As events or services being promoted on the leaflets don't last forever, it is quite important to distribute all brochure before the end date because discarding the remaining leaflet is a waste of money and paper.

Another main issue is the mistakes that can easily be made in print work, and when flyers are already printed out, this will turn into an extra cost to amend.

In our current era of tech, flyers is not the best way of publishing an event. It includes a long process to create a simple leaflet, finding words which are more compelling to your

message, have a design, print and distribute the flyers. This process takes a long time and a lot of money if the leaflet needs to look professional, which consequently means working with a design company, photographer or a graphic artist. This often ends up being an arduous and expensive process.

Another disadvantage is that they are most often dismissed by people as “mere sheet of paper” according to an article at brochuremonsters.com. Complacency viewed by individuals as annoying, for those who don’t want to chat and don’t have time to stop and listen, to what is being advertised as they will rather do it in their own time. One might also glance at it for a second, but if it doesn’t spark any immediate interest it will end up in the nearest bin.

“This is very alarming as often it will end up as waste, and the main reason is that people are receiving leaflet they don’t want to receive”, said Ryan O’Quin, membership & development with the Ecology Action Centre in Halifax. Reducing the use of this form of advertisement will help the environment through the reduction of paper waste.

## 1.3 Solution

### 1.3.1 Existing solution

Though there is a vast market for event applications with the purpose of organising social functions and activities, there is always scope for more innovative ideas.

This thesis will seek to examine several existing event applications in section 5.2.

### 1.3.2 Proposed solution

This project aims to create a paperless solution where information is always available and easy to access. With this solution students will rely on mobile application, allowing them to read or post any ongoing or upcoming events (club nights, parties, football matches, university events, external speakers, etc.).

Let’s consider the amount of time it takes to prepare flyers, create a design, decide on content, print and distribute, all of this is time consuming, and if you want the leaflet to look professional, this means hiring a marketing or design firm.

Flyers can be labour intensive and costly to produce and distribute especially if targeting a large audience and it’s very difficult sometimes to build a connection with a target who are always on the go. The number of people reached by flyers are very limited as this is highly dependant on the number of flyers you can distribute to the audience. It is also quite an arduous process, from design, printing and finding the perfect location to reach as many people as possible.

Another disadvantage is the finality of print compared to an app, once a poster is printed it is more difficult to make corrections or adaptions and it is therefore less flexible

in its adaptability when compared to a software presentation that can be modified any time.

This is where a paperless solution for flyers has a powerful advantage. This minimizes your expenses, as there is no need to pay for printing and there is no need to worry about losing information, which can be accessed everywhere.

The application aims to provide the latest event on users mobile phone with all features designed to work online to give the latest information to users, and improve the event experience in every aspect: it is designed to build stronger connection for attendee satisfaction, a comprehensive event search to look for any interested event, have a customized event dashboard displaying only events of interest and more.

This application not only provides easy to access information to attendees, it helps organizers plan events, create the event, publish it and keep track of the number of attendees.

By providing all the benefit, and useful features of the application, it should show how essential it is to have a unique mobile platform for universities where events can be posted, minimizing effort and maximizing benefit to all attendees and event planners.

## 1.4 The Challenge

With the number of prospective students who use smartphone steadily increasing, along with internet usage, Universities are improving student experience and enabling them to see exactly what's happening at the university and feel part of a wider community through use of tech. BU has already embraced social media and new technologies to engage its students through channels like, Facebook, Twitter, BU app, etc.

The embracement of tech is further illustrated with BU own app, which enables students to find numerous information from checking their timetable to downloading lectures and seminars.

Infact it will be brilliant for this app the author has developed to be taken on board by the university and incorporated in the BU mobile app, as often integrating other apps on an internal system will be a challenge but as Doug Poole, digital media officer at Southampton University stated, -"it is worth persevering and joining these systems up as that is when the user will experience genuinely useful functionality".

## 1.5 Target Audience

The main users of this mobile application will be everyone on the faculty. This is mainly designed to aid students and everyone at university to be aware of any ongoing activities or events at the university and help event organizers to have an easy platform to publish any activity they have organized. The app has a register and login feature which only

accepts university emails, validates the user login details and checks a university email domain. This is one of the main feature of the app as it focuses solely on the University events and any other activities posted by students, this makes the app unique as all the similar mobile applications are targeted at a broader range of audience.

## 1.6 Project Aims

The application aims are to solve the problem mentioned above. To achieve this, the project must inspire and convince users away from printing out flyers or brochures and make it easier for students to post any upcoming events and reach as many audience as possible through the use of the app. Lecturers can post important events (external speaker etc.) and the University can post any important upcoming events. This project will make it much easier for people to connect and make each other aware of any events ongoing or upcoming events, through easy accessibility on their mobile phones. All these features are designed to work online to give the latest information to users and improve the event experience in every aspect.

## 1.7 Project Objective

The thesis objective should formulate a means to providing a solution to the problem, this following objective have been set out to fulfil this aim:

- To elicit requirements from observation
- Provide a solution that helps users to keep track of activities in the faculty and aid them in publishing any new ones.
- Design front end system that will show JSON data retrieved from server.
- App must work without delays and have smooth transaction
- Final project must work without issues, ensuring user safety

## 1.8 Risk

Risk in software project incorporates various factors or conditions that may represent an issue for the completion of the project. Risk can be divided into two component; the likelihood the risk will occur and the impact of the risk.

Managing risk includes identifying any potential risk, verify the impact on the functionality and come up with a strategy to control it.

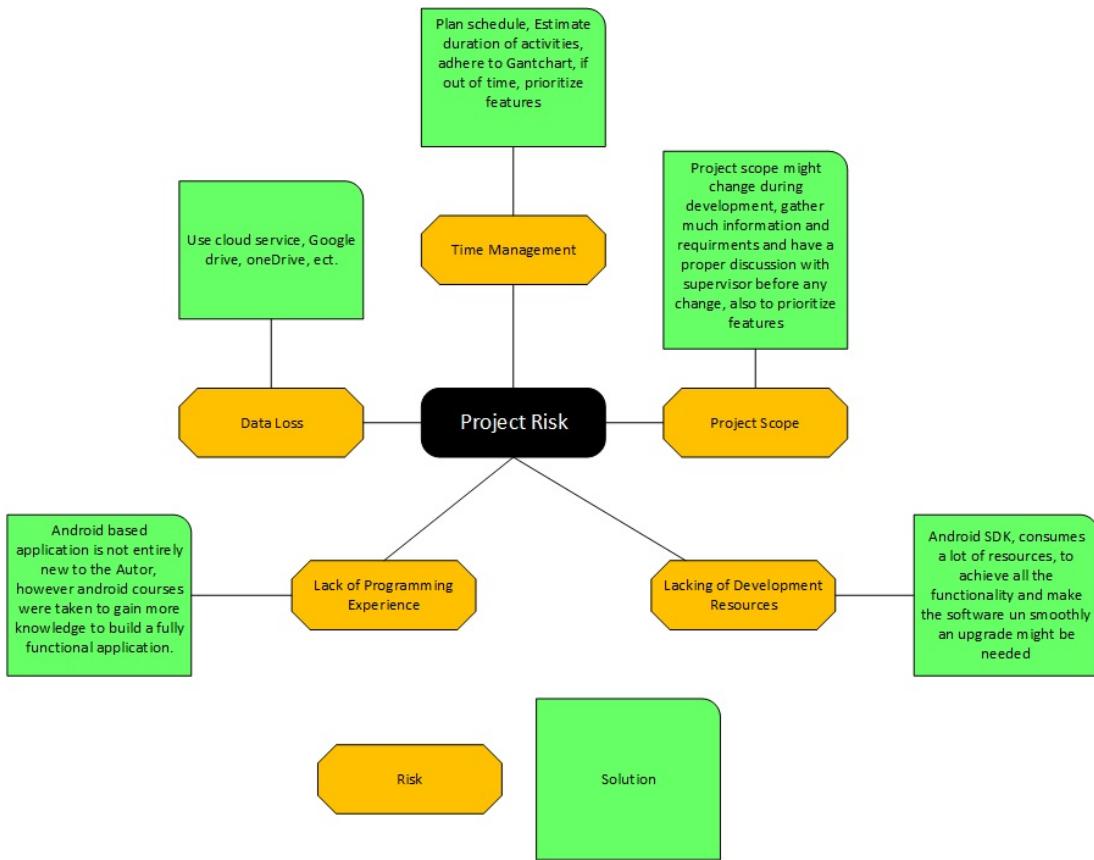


Figure 1.1: Risk Diagram

Likelihood	Impact
P = Possible	M = Major
U = Unlikely	Me = Medium
L = Likely	Mi = Minor
V = Very Likely	N = No Impact

Table 1.1: Likelihood and Impact

Risk ID	Risk	Impact	Likelihood	Solution
1	Project Scope	Me	L	Project scope might change during development, gather much information and requirements and have a proper discussion with supervisor before any change, also to prioritize features
2	Time Management	Mi	V	Plan schedule, Estimate duration of activities, adhere to Gantchart, if out of time, prioritize features
3	Lacking of Development Resources	M	P	Android SDK, consumes a lot of resources, to achieve all the functionality and make the software un smoothly an upgrade might be needed

4	Lack of Programming Experience	Me	L	Android based application is not entirely new to the author, however android courses were taken to gain more knowledge to build a fully functional application.
5	Data Loss	M	V	Use cloud service, Google drive, oneDrive, ect.

Table 1.2: Risk Analysis

## 1.9 Achievement

A mobile android application has been developed successfully and ready to go on the market. All requirements are met, and it has been tested by both client and developer and all necessary tests have been executed. The project accomplishes the following objectives:

- All functionality is integrated and working efficiently
- Multiple actions can be performed on the app
- App is published and can be downloaded from the Google Play Store Market

## 1.10 Android vs iOS

This section explains the main operating systems on the market, those being Android and iOS, see Appendix B

## 1.11 Success Criteria

The measure of success can be judged on whether the project as a whole offers a solution to the aforementioned problem. The app's functionality will be assessed throughout

the entire software development lifecycle, in the form of debugging and running the application on a mobile device. Meeting with potential clients will provide feedback that will aid in verifying most of the functionalities which may well lead to changes in the features and appearance of each component. The main feature of the app is to provide support in publishing and keeping track of activities in the faculty, therefore the app will be uploaded on the play store, to enable users to download and leave comments regarding their experience of using the app which will allow collation of what feedback that can be utilised to gauge the overall success of the app and allow the developer to improve any areas of concerns relating to the main function of the app.

# **Chapter 2**

## **Methodology**

### **2.1 Overview**

User requirement changes continuously and relying on a methodology helped organize the development into phases, providing a more flexible development. This chapter will detail the approach that will be used in the project, with the aim of outlining how each one will lead to achieving the objectives.

### **2.2 Agile**

“Agile methodologies are purported to imbue flexibility in software development projects, thereby enabling software development teams to perform more effectively”, Maruping et al. (2009). They share the property of iterative incremental development that tackles requirement changes quickly, satisfy customer and produce quality products, Hamed and Abushama (2013). Many Agile methodologies has been introduced, of these, XP and scrum are the most popular.

#### **2.2.1 XP**

Extreme Programming is a software development methodology designed to improve the quality of software and its ability to properly adapt to the changing needs of the consumer, Powell-Morse (2017). Similar to other Agile Methods of development, XP aims to provide iterative and frequent small releases throughout the project, allowing both team members and customers to examine and review the project’s progress throughout the entire System Development Life Cycle(SDLC).

Extreme Programming amends a software project in five essential ways; communication, simplicity, feedback, respect, and courage. Extreme Programmers get feedback by testing their software from day one and distribute the system to the customers as early as possible, implementing changes as suggested by the customer.

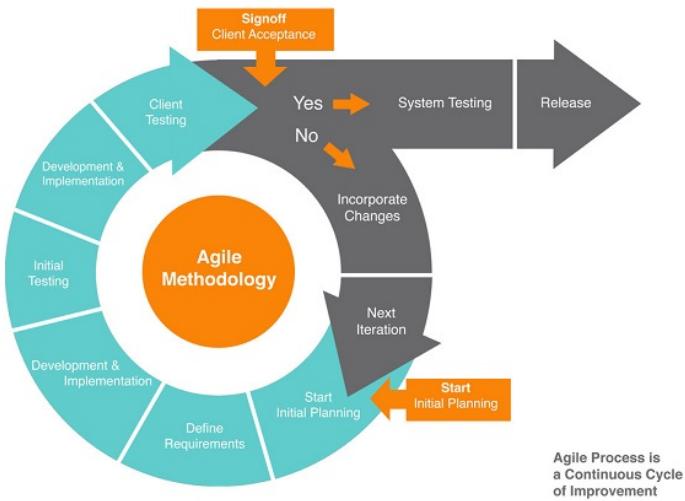


Figure 2.1: Agile Lifecycle

### 2.2.2 Scrum

Scrum is a subset of Agile and one of the most popular process frameworks for implementing Agile. It is an iterative software development model used to manage complex software and product development. Fixed-length iterations, called sprints lasting one to two weeks long, allow the team to ship software on a regular cadence. At the end of each sprint, stakeholders and team members meet to plan next steps, smartsheet (2016).

Scrum follows a set of rules; a scrum master has to be appointed, he/she normally conducts scrum meetings and makes decision. Scrum also relies heavily on backlog(list of features, bug fixes and improvement). These meetings aims to get developers to be more productive with an appropriate amount of supervision to handle issues that may arise.

### 2.3 Cowboy

Most methodologies focusses on groups of people to collaborate efficaciously to indite a software project, what about solo programmers? Cowboy was designed to fill this void.

Cowboy is an agile system that benefits from agile methodologies, customer centered approach and helps the programmer to stay focused while meeting customer needs. It is an iterative approach, meaning it adds features and fixes bugs of previous cycle.

Finally, cowboy suggest that the artefact should be kept simple and adequate, encapsulate the core ideas, which are subject to change(or even elimination) as requirement change, so not much time should be lost perfecting them, Hollar (2006).

According to Hollar, developers who had relied on this methodology includes; Alan

Turing, viewed as the precursor to the modern programmer, Bill Gates, essentially when inditing MS Basic.

## 2.4 How Agile Helped Cowboy

Cowboy relies on having a customer(or at least someone knowledgeable to represent the client interest). In this thesis it will be the supervisor of the project and a potential client interested in the development of the artefact. Their feedback should weigh somehow in deciding the order in which to add the features. The developer has autonomy over the development process and the authority to refuse a feature suggested by the client, which if pursued may take away vital time.

During development, cowboy suggests developers should target the huge problem first then move on to minor issues, code refactoring should be done during each cycle.

The combination with the agile methodology mentioned above, helped in the successful development of the application.

Deniz Centinkaya, agreed to act as the supervisor of this project and it was planned to hold weekly meetings, and continuous communication through email was relied on as well.

## 2.5 Project Plan

In order to have a successful project, a Gantchart has been developed. The chart breaks down the project into sub projects and provides the start and the end date of that particular sub project, refer to appendix H.

# **Chapter 3**

## **Requirements & Analysis**

### **3.1 Overview**

Requirements are essential to software development, they define the property and the functionality of the software. This chapter looks at the requirement elicitation and method used for prioritisation of the requirements. It also mentions what are the main issue faced during an android app development.

### **3.2 Requirement & Analysis**

Since this thesis has no client (stakeholder) to gather requirements from, the main objective of this projects was to solve a problem the author recognized, in the Bournemouth University, refer to section 1.2. Requirement was mainly defined based on observation and thorough research. Similar applications on the market (refer to section 5.2) were studied and from the result requirement were derived. In addition requirements gathered was discussed into detail with the project supervisor.

The beginning of the project was mainly spent observing how events and activities are organized on campus, trying to gain more knowledge on the method used and the effects in using such method. Thereafter further research was done on similar applications on the market, that can potentially aid in solving the issue. All of this lead to deriving the requirement needed to build the application.

After gathering the requirements the Numerical Assignment(Grouping) method was used to prioritize the requirements. Numerical assignment is the most common prioritization technique, this approach is based on grouping the requirements into priority groups, the number of groups can be different but normally its three (critical, standard, optional).

Listed below are the essential functional requirements of the mobile application.

Group	Requirements
Critical	The application should have a register and login screen
Critical	The application should integrate a validation for email, only universities email are accepted
Critical	The application should provide specific information such as event location, event start and end date, description, title.
Standard	The application should provide information of all attendees interested in an activity or event
standard	The application should have a “Share” feature. This will open any interested app installed on the users phone so they can share the event with friends
Standard	The application should provide a “Send Email” feature. This will open any email client installed on users phone, so they can email the event host for further information regarding the event
Standard	The application should have a “Map Feature”. This will direct the user to the Maps application with the event location already pre entered.
Standard	The application should have an “Add to Calendar” feature. This will open any calender application on the users phone and it will allow one to save the even on the calender creating a future reminder.
Standard	The application should allow the user to choose category they are most interested in and based on that information display all types of event or activities in those categories
Optional	The application should save the user login detail, so user has to login just once.
Standard	The application should show the location on a map directly on the screen of the app

Table 3.1: List of Requirements

### 3.3 Application Vision

The vision gives an idea of the functionality of the application. The purpose for this application is, that an individual who sees an activity or event, can get all the relevant information available on that event. Any other user with Android devices can participate, and also see who is already participating in that particular event. In order to do so, the user has to press on the button RSVP on the app, to be added to the attendee list and this information will be available to anyone who has access to the app but to maintain privacy only the nickname(username) of the attendee is shown. The target people for this app are users with Android devices who are currently enrolled in the university of Bournemouth.

The mobile app enables the user accomplish two main things: view existing event or activities that other users have created and create a new Event. To be able to add an event, the user must enter various information such as, event name, start date, start time, end date, end time, location, description, event type and an image(optional). On an extra view users can see only events based on their interest. The app offers the option to choose which category of event the user is interested in, which creates a second view with a personalized view of event recommendation based on the information provided. To be able to add an interest, the user has to go to the profile page and click on interest and then choose as many categories as they want. All information are stored to a database. If the internet connection is not enabled, the application will notify user that a connection is needed to continue, (unfortunately this feature was not included due to lack of time, it will be considered in future works). To upload the information to a database, the client must synchronize with the server, then the server will query the database. A user who wants to search through all events, can go to the menu, click on search, which allows them to search by event name. The application connects automatically with the database, and displays all events in order of date. The application provides the option to save any upcoming events the user is going to attend and also to show any event created by the user. The app has extra features to enhance the experience of the user, such as; they can share events with friends by clicking on the share button, they can also add the event to the calender by clicking on add to calendar feature, finally the app can direct the user to the maps app, providing the location of the event. To show the direction on the map, the users device has to have a location based application or have a browser installed. User also has the opportunity to email the event host directly by clicking on the send email button on the event information.

# **Chapter 4**

## **Background Study**

### **4.1 Overview**

This section will show other areas which required further research to aid in making a concrete decision as to which technology will be used for the development of the project.

#### **4.1.1 OS**

The majority of the apps on our mobile phones are native apps, meaning they are written in languages that the platform accepts, for example swift is used to write iOS apps and java is used to write Android apps while c# is used for most Microsoft phone apps. Native apps are very responsive and also offer the most reliable experience to user. On the other hand an hybrid app is more similar to a web app but is installed like a native app. They are normally built with javascript, HTML, and CSS and runs in something called webview, a browser within the app. Performance wise, however, it's inferior compared to native apps.

The project was developed as a native app, using the Android OS, this was chosen based on the fact the author has knowledge of the platform and it is the OS with the largest share on the market, mobiloud (no date).

#### **4.1.2 Database**

MYSQL was chosen as the database for this project, mainly for its scalability and its compatibility with Java and PHP language. For this reason phpMyAdmin was used aswell as a development tool to handle database management.

## 4.2 Development Environment

Relying on tools to develop an app is very useful as it helps accelerate the development. For Android there are various tools which this chapter will focus on describing. To be able to build the application a development environment, Java Development Kit(JDK) and Android SDK are required. To develop with the SDK, Google offers a bundle, the bundle comes with an IDE and the Android SDK.

### 4.2.1 JDK

Java Development Kit is the essence of any java application. As defined by the Technopedia Dictionary, “the JDK is a software development environment used for developing Java applications and applets. It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development”.

### 4.2.2 SDK

Android Software Development Kit provides a selection of tools and libraries required to build Android apps to ensure the process goes as smoothly as possible and the SDK is used to get it to run on an Android device and access unique features of the OS.

### 4.2.3 Android Emulator

The emulator allows to run the app without having to install the app on an actual Android device. It emulates most of the functionality of a real Android device, apart from for the GPS module.

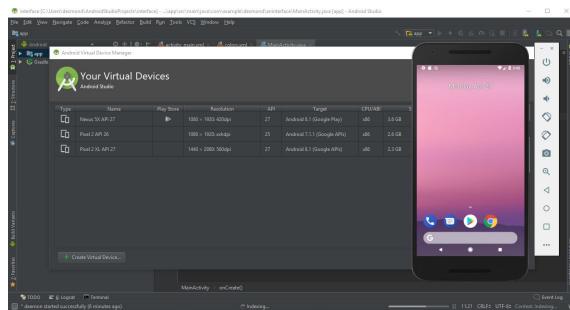
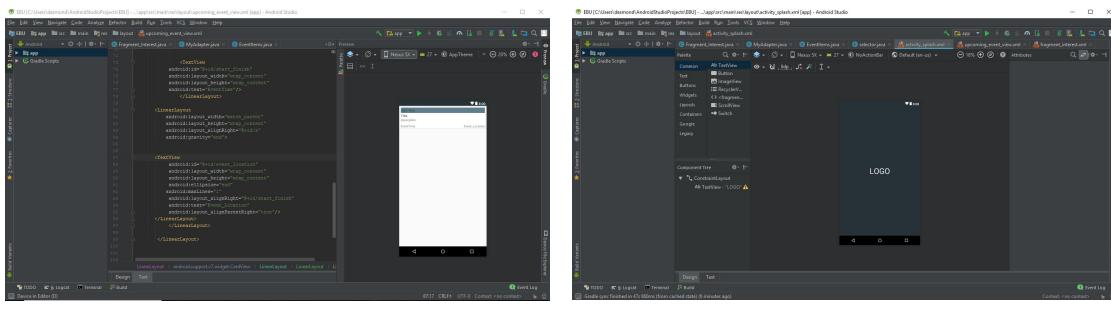


Figure 4.1: Android Emulator

### 4.2.4 Graphic Layout

The Layout Editor from the SDK allows developers to create user interfaces(UI) with an editor(Drag and Drop) or by writing XML Code. These editors support many features for



(a) XML Text View Editor

(b) XML Design View Editor

Figure 4.2: Andriod UI Design Editor

rapid UI development and both are easy to use and it is directly included into Android Studio. The screen of the editor is shown in figure 4.2.

#### 4.2.5 IDE

According to TechTarget Integrated Development Environment(IDE) is “a software suit that consolidates the basic tool developers needs to write and test software. An IDE consists of a code editor, a compiler or interpreter and a debugger that the developer can access through a single graphical user interface”. For the purpose of this project Android studio was used as its the official IDE for Android development.

### 4.3 Android Framework and API

In order to build the artefact, it is essential for the author to understand the Android framework and API, this information is found in Appendix E.

### 4.4 Developing Problem

Dealing with diverse platform is one of the most challenging aspect in app development. Mobile development is moving towards fragmentation rather than unification.

**Fragmentation:** Considering Android devices, most of them comes with different screen resolution. Different devices exist with different properties such as CPU speed and memory. According to a study made by Mona, Ali and Phillip in their article “Real Challenges in Mobile app Development”; 76% of their survey participants see the existence of multiple platform as a challenge for developing mobile apps, while 23% believe it is an opportunity for technology advances that drive innovation. The development process cannot leverage knowledge from a platform to another. Also, when an application is created for various platforms, the developers must treat the development independently and check that consistency is kept across each one.

## **Chapter 5**

### **Literature Review**

The concept of paperless has been around for some time now and has long been the ambition for most organisations and individuals, however it's yet to be realised. Paper has become an integral part of our daily lives; people learnt how to properly organize their tasks, previous generations have used it to pass their wisdom to the current generation, many individuals cannot imagine a life without paper. However with the rapid growth of Information Technology(IT), many things are changing. Nowadays most people want immediate or instant services, they want access to information at any given time and place. As such the pedestal that paper used to rest on appears to be becoming obsolete, with an upward trend in the number of organizations trying to reduce the use of paper.

Isaeva and Yoon (2016) published an article on paperless Universities, they mention that despite the progression of IT, being paperless is still a dream for many companies and the case looks more critical when it comes to Universities - "the biggest paper consumer organization". They posed a pressing question, "Is it possible to make a campus paperless, or will it forever remain as a dream?". Nowadays universities around the world have adopted computer-based systems. However, the management systems of universities are still paper based. The article goes on to mention some reasons for going paperless as the following;

- Reduce the amount of paper used;
- Reduce the amount of time spent searching and retrieving documents;
- Reduce the amount of duplicate data;
- Decrease the physical space allocated to file storage;
- Increase flexibility of documents' use and security;
- Increase efficiency of university administration processes;
- help to save natural resources;

According to an article published on the Yale University website, in 2011, the University consumed 211,033 reams of paper - “that’s enough paper, if laid end to end, will stretch three-quarters of the way around the earth”. Using paper doesn’t only take up natural resources, it’s also very expensive, (i.e, printing, storage, disposing, recycling and processing). Recognising this, Yale University’s departments and schools initiated different steps to go green; not only are they benefiting the environment but gained financially by shifting to digital solutions. For example: The office of Student Employment saved \$100,000 by replacing paper timesheets to electronic timetables. The department of Finance and Business Operations managed to save around \$60,000 when they started publishing their annual reports online, Ngim (2013).

The above showcases the reason in going paperless in going paperless: Cost saving, improving data sharing, avoiding duplicate. Many organisations are trying to implement paperless, however, it is not an easy task. Although many people are comfortable with technology there are still reservations when new developments are made, this is unsurprising as most fear the unknown. A paperless world is a huge leap into the unknown. However this may be managed in two ways: Graphic User Interface(GUI) and Gamification.

As mentioned in Isaeva and Yoon (2016) article, a crucial factor in going paperless is simplified GUI. Developers tend to go overboard with the amount of features implemented and end up having a cramped system with lots of redundant information. As a result, users cannot highlight what is most important and become overwhelmed. Thus, developers should focus on making easy and concise GUI to make it more user-friendly for the target audience.

**Gamification** - is a comparatively new term that is used to describe the use of game elements and game design technique in non game context. The aim of this method is to induce a certain behaviour in people by relying on game mechanics to improve motivation and involvement in a specific task. According to the self Determination Theory written by Deci and Ryan (2008), people have extrinsic and intrinsic motivation to perform a task. Extrinsic is about the rewards and prizes, on the other hand intrinsic is related to the task performed because its fun and people like it. Websites like stackoverflow uses gamification to try and keep user motivated to perform a certain task. This system is normally driven by points, for example on stackoverflow, an online programming community, rewards users with points and badges after they have posted a number of questions or replied to questions posted by other users. The most popular elements are leaderboard and badges,

Nowadays, the newer generations make exhaustive use of digital technologies. They like to learn relying on technology, it’s not surprising that, the use of game elements in a non-game context seems a choice to be used in education, it has been used as an inspiration of engagement, motivation to boost learning by providing an environment that

supports cooperation, competition, feedback and reward, Kaplan and Haenlein (2010).

Hamari et al. (2014) made a research about “Does Gamification Work?”, based on his assumption, gamification is seen in three main component. First is the motivation effect, second is the psychological outcomes and third is behavioral outcomes. Ways to entice users to keep using a gamification product is to make them feel that the activities are worthy. Applying gamification have to go through a clear arrangement to have an impact on users. The game elements has to be selected cautiously because every game element has it's own function. A bad gamification design can have an harmful effect on the system, Hanus and Fox (2015).

According to a survey made by Uskov and Sekar (2014), in which they interviewed 18 volunteer students to collect their feedback about the impact of badges and leaderboards during a course experience and they also interviewed six students which were randomly selected for interviews about their perceptions about the course and the use of game elements. Their results shows that, the concept of badges had a greater impact on the motivation of students than the leaderboards. Students had shown more interest in gaining badges and saw them a social reward and secondary objectives to strive for in the course.

## 5.1 Applying Gamification to Univent

Gamification can be used as a means of promoting rewards for completing tasks. In the learning environment, students can be rewarded for taking the initiative to improve their soft skills. In this way, some of the discrepancies in personal efforts that are often present in student project work are reduced. The same technique can be applied to Univent, allowing individuals to publish an event and rewards up front often goes a long way to ensuring that all individuals are working towards paperless. Moreover, integrating a gamification system according to which, when a user achieves a goal it triggers a reward system based on gaining points. In this way, the user's interactions with the app will be more focused on going paperless. The reward system based on points is aimed to measure the paperless success of a host, the number will increase, and decrease based on the reaction of their events, so the points will change depending on the upvote or downvote that were made on the host's feed. The same effect on the points was determined by how many paperless events a user decides to host (host will be able to tick a box to declare they are going 100% paperless for that event, meaning the will only use Univent to advertise plus they will not use paper base advert such as posters). The more point and upvotes a host gets can get them more ad space, meaning that each event they post will be of high priority and will be added at the top of the event list and will further be added to the recommendation section on the app, potentially increasing the number of views.

## 5.2 Related Work

This section will explore the functionality and the utility of existing solution that will be viewed as competitors.

### 5.2.1 Fever

Fever is an event discovery and booking app for iOS and Android. It provides information on the best events in major holiday destinations in Spain with expansion to major international cities like London, New York. Fever inspires users to find out about what these cities has to offer, from music festivals to fashion and restaurants.

The app also looks at each user's top three interests, their activity, and interactions with other Fever users to offer a better recommendation engine.

Fever is more a location based app, meaning the app uses the users location in order to find which events to promote. Although it's not targeted at the locals as they may already be aware of most of the services suggested, conversely it is likely to be of tremendous interest to people who are new to the city and will like to discover the best restaurants. Furthermore Fever only publishes event that they have personally selected and is not open to public posting. However, if someone outside of Fever wants to promote on their app they can fill out a form, but this is a long process to advertise on the app. The app is only currently available in: New York, London, Paris, Madrid, Barcelona, Seville, Valencia, Bilbao and Malaga.

### 5.2.2 EventBrite

This application allows the user to find what's trending nearby at a wide variety of venues. They can keep up to date with any upcoming event like concerts, festivals, holiday events and networking events. Eventbrite enables users to buy tickets directly on the app stored on the mobile phone for convenience access. Users can also store their credit and debit card details for faster payment. The application can find things to do based on what the user is into, where they want to go or when they want to go out. This will provide a recommendation of events for the user, based on all these information. The application includes a share feature where users can share events with friends and vice versa and the app is available in German, Spanish, French, Italian, Dutch, Portuguese, and Swedish.

Overall the Eventbrite app has similar features to what is described in the application vision, see section 3.3 and even more features and currently more than 5 million downloads according to the play store. One major drawback of this application, however, is that, the mobile app doesn't have the feature that allows to add events. The only way to add an event is to use their main website and it's more suited to larger enterprises who are looking for a more convenient, visual way to manage their events.

Currently Eventbrite is based on events worldwide, giving the user access to events in various parts of the world. Univent will still be unique, as at the moment there is no app that covers events or activities just for Universities (in this case the Bournemouth University) or any event related to the members enrolled on the faculty.

### **5.3 Summary**

Eventbrite and Fever are good examples of how Univent will work. They showcases the main features and how tasks can be handled. Furthermore, the review of both provides a better understanding of what features are required and gives the opportunity to identify any missing features that could be important to implement for an all round user experience.

# Chapter 6

## Univent Design

### 6.1 Overview

The app design is a very important phase of the development lifecycle, as it can have an impact whether or not the app is accepted by the users. This chapter presents the choices made for the design of the application.

### 6.2 Android Application Design

The user interface is designed using widgets. Android provides basic widgets such as, image view, textview, etc., which can be used to create the application. All these widgets are included in the Android SDK. Android gives users the possibility to create their own widgets, named custom widgets.

All the screens in the project are composed of various widgets, both basic and customized.

The home screen, includes widgets such as, RecyclerView, TextView, ImageView. The recycleview is populated with data directly from the database.

- **TextView** - A textview displays text to the user, normally displaying contextual information or the name of other elements on the screen. Figure 6.1, shows how to define a text view in the XML editor.

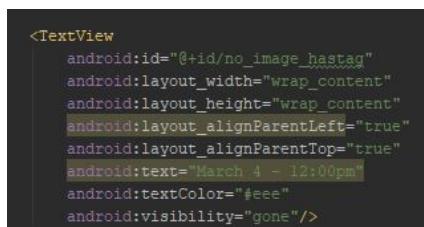


Figure 6.1: Define a TextView in XML

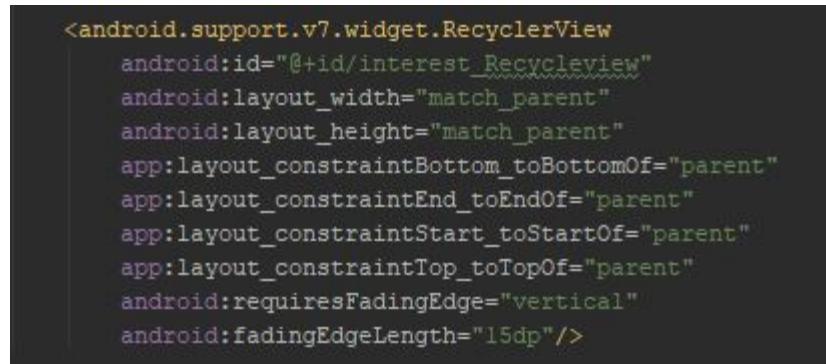
- **ImageView** - An ImageView is used to display image to the user from the resource file or from an external source, like the internet. The following snippet is an example of imageview, refer to figure 6.2



```
<ImageView  
    android:id="@+id/imageID"  
    android:layout_width="match_parent"  
    android:layout_height="300dp"  
    android:layout_alignParentLeft="true"  
    android:layout_alignParentStart="true"  
    android:layout_alignParentTop="true"  
    android:scaleType="fitXY" />
```

Figure 6.2: Define a ImageView in XML

- **RecycleView** - According to the official android website, “a Recycleview is a view group that displays a list of scrollable items”. The list items are automatically inserted to the list using an Adapter that pulls content from a source such as an array or database query and converts each item result into a view that’s placed into the list. Figure 6.3, shows how to declare a recycle view.



```
<android.support.v7.widget.RecyclerView  
    android:id="@+id/interest_Recyclerview"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    android:requiresFadingEdge="vertical"  
    android:fadingEdgeLength="15dp"/>
```

Figure 6.3: Defining RecycleView

Recycleviews are normally accompanied with cardveiws, these shows information inside cards, and its corners can be customized by the user. The main function of these cards is to act as the rows of the recyclevew.

- **ViewPager** - A viewPager is viewgroup that allows the user to swipe left or right to display a new screen. Its a more efficient and user friendly way of displaying screens to users, refer to Appendix A.

### 6.3 Initial Design

Before the final design of the app was created, various designs were made in attempt to give a better UI experience to the user, refer to Appendix A. All the designs were made with the Android XML editor to see exactly how it will look on a mobile device. Using the XML editor made designing simple, meaning no need of wireframe designs before recreating them for the app, which saved a lot of time. All the widgets mentioned above in section 6.2, were used to complete the design. Extensive research on similar applications, section 5.2, provided inspiration for the final look of the app and the meeting with the client also provided ideas for new features to implement. The app was further redesigned to be more user friendly, with two new screens created, Discover and Interest(named as "for you" in the app), using viewpager, as detailed in 6.2, so users can swipe left or right to move between screens as shown below in figure 6.4.

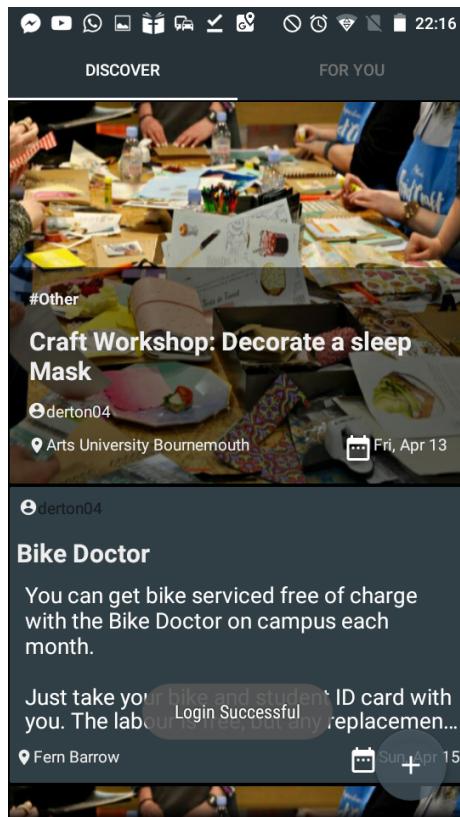


Figure 6.4: Home Screen

The discover screen shows all the upcoming events with no filter, while the "for you" screen shows only recommended events based on the user interest. In order to achieve this a new feature was to be added, as a way for the user to choose what they were into. A snippet of the Interest screen can be found in Appendix A.

## 6.4 Database Design

Figure 6.5, describes the structure of the database. The event table contains all the information of the event, the user table has all the user information and finally the GoingTable, contains the ID of the users an the specific event ID the wish to attend.

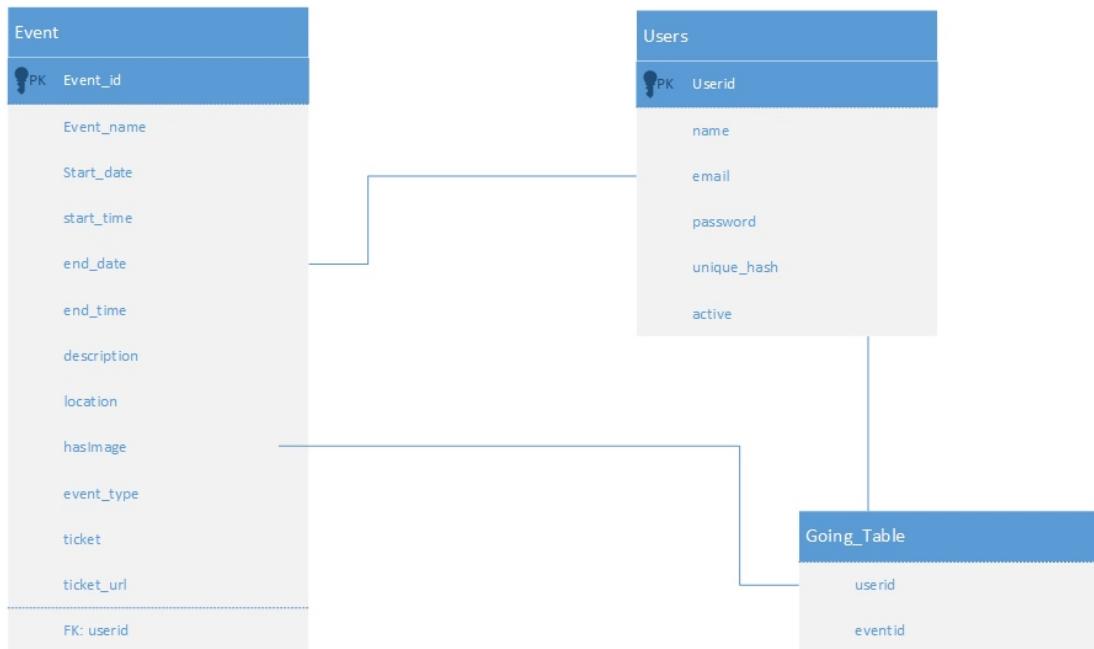


Figure 6.5: Database Design

### **6.4.1 Security**

Considering the importance of data it is not a surprise for attackers to target the data these are containing. In this section various challenges in database security are discussed.

- **Blank Password, blank email:** In order to avoid this issue, a password and email validation has been put in place, which will notify the user if any of the fields are left blank.
  - **Secure credential:** User passwords are not stored directly onto the database, but are encrypted using a hash key. The hash key is then stored in the database, whenever the password is called it is then decoded.
  - **Excessive privileges:** Granting Users (or applications) database privileges that exceed the requirements of their task, these privileges may be abused for malicious actions.

- **Weak Authentication:** Weak authentication allows attackers to take the identity of database users. A counter measure for this issue will be to Implement a two-factor authentication.

#### 6.4.2 General Architecture

An Android device with the Univent application already installed communicates with the web service using a RESTful API. The application sends HTTP requests with GET/POST method headers and receives formatted JSON responses. The API is written in PHP and handles querying the MYSQL database. Figure 6.6, shows the system architecture.

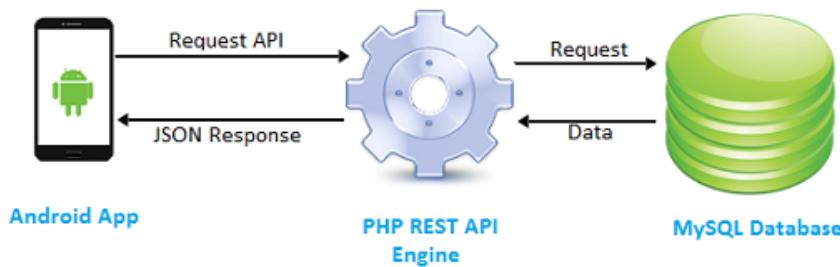


Figure 6.6: Rest API

# Chapter 7

## Implementation

### 7.1 Overview

This section will describe how the features were implemented to develop the final artefact, the system will be developed using Android studio.

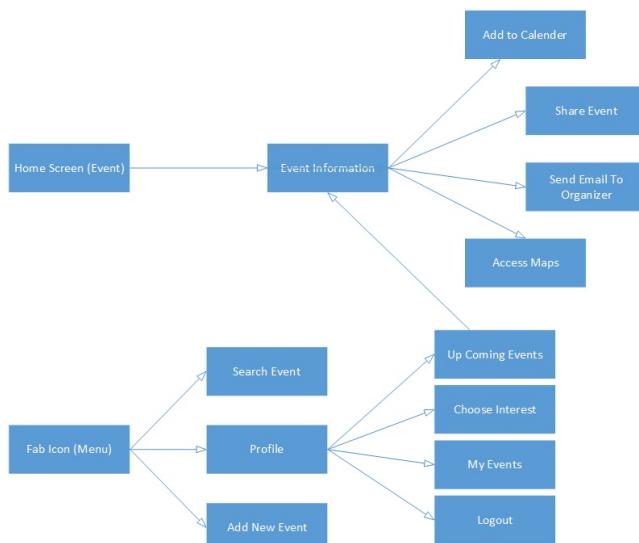
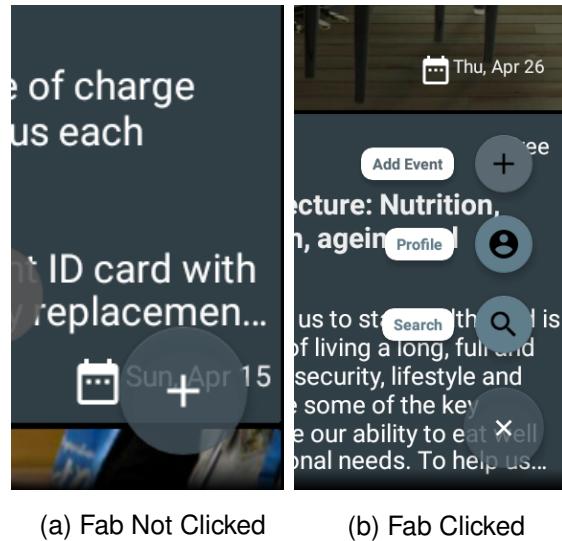


Figure 7.1: Logic Flow of Screen

The figure above 7.1 shows the relationship between each activity(screen), how to move between screens. Both screens, upcoming event and homescreen link back to event information, because this option is available from both activities, meaning the user can access event information from both. Add to calender, share event, send email to organizer and access maps are all behind event information, this means the logic behind each of those activities has to be implemented in the event information activity. Fab icon, refer to figure 7.2, is a navigation menu that links to search event, profile and add new event.



(a) Fab Not Clicked      (b) Fab Clicked

Figure 7.2: Fab Menu

## 7.2 Login & Register

The login.xml has the function to display the views to the user while the login activity has the function to handle all the java logic in the background. This activity will ask the user to login, user then has to select their university, enter university email and password, this information is then checked against the user database; if the user has already got an account and the password is correct the user will be logged in successfully. This activity also checks if the user account is verified or not and displays the appropriate message. The user can move from the login activity to the Register activity by clicking on the textview, "not registered". Refer to appendix A, for details on screen design.

Activities relies on intent, refer to Appendix E.3, to move from one activity to the other, the figure below 7.3 shows an example of how to move from the login activity to the register activity. Intent also allows data passage between activities; for the purposes of this app data such as user email, user ID, was passed through activities to maintain traceability. In order to enhance the user experience, shared preferences APIs was used to store

```
Intent NotRegistered = new Intent( packageContext: Login.this, Register.class);
startActivity(NotRegistered);
finish();
```

Figure 7.3: Declaring an Intent

user login details on their phone(internal memory of the phone). Shared preferences permits saving and retrieving key value pair data to be used to save primitive data type such as Boolean, string, int, long and float.

**MODE\_PRIVATE** – this is the most used mode of sharedPreferences, It is a default

mode, which means that when any preference file is created it will only be accessible from the application.

Calling the edit() function of SharedPreferences class which returns Editor class object allows to save data into sharePreferences, refer to figure 7.4

```
sharedPreferences = getSharedPreferences( name: "SaveLoginDetails", MODE_PRIVATE);
editor = sharedPreferences.edit();
editor.putString("email", Email);
editor.putString("password", Password);
editor.commit();
```

Figure 7.4: Declaring SharePreferences

Values stored in shared preferences can be called using SharedPreferences object by calling different primitive type function starting with get plus the Primitive type name, refer to figure 7.5

```
sharedPreferences = getSharedPreferences( name: "SaveLoginDetails", MODE_PRIVATE);
String option = sharedPreferences.getString( key: "option", defaultValue: null);
String username = sharedPreferences.getString( key: "email", defaultValue: null);
String password = sharedPreferences.getString( key: "password", defaultValue: null);
```

Figure 7.5: SharePreferences key value pair data

The **register activity** allows the user to create an account if they are not previously registered. It prompts the user to enter their details, university name, nickname, email and password. Afterwards, before registering the user, it verifies if the user does not have a previous account, if so it will displays a message. Otherwise it creates an account and sends a verification email to the users email address to verify their account, refer to appendix A.

Once the user is registered and logged in, the app receives all the event data from the server and displays it on the home screen.

### 7.3 Implementation of REST API

REST stands for "Representational State Transfer" REST architecture will be used to build the client/server applications. It's simple to implement REST as it basically works on HTTP protocol, refer to figure 7.6 below.

Since Android does not have a dedicated library for implementing A REST API, it provides a number of pre-implemented solutions that can be used for the implementation. According to Bedyński (2011), the main issue however, is how to design an action flow between components of the system from the presentation layer (Activities) down to the network and local memory operations. He also mentions that there is nothing like the best way of implementing a RESTful API client on Android platform because each solution can,

<b>GET</b>	To fetch a resource
<b>POST</b>	To create a new resource
<b>PUT</b>	To update existing resource
<b>DELETE</b>	To delete a resource

Figure 7.6: HTTP Methods

and should be modified according to the unique requirements of the protocol, however, there is one pattern which is definitely not advised, that is to, avoid running RESTful methods directly from the UI thread.

- this would cause an ANR (Application Not Responding error).
- slow down the application and make it less responsive

Android however provides a solution, `AsyncTask`, which is a high-level concurrent construct. `AsyncTask` can interact with the UI thread by updating the UI via event handlers. For example, the event handler `onPostExecute` executes after the task is finished, and can update the UI with the task results. A study by, Lin and Dig (2015), shows `AsyncTask` is the most used async construct in Android. However, it is designed for short-running tasks (i.e., less than one second) and if improperly used, can lead to memory leaks and lost results.

The basic methods used in an android `AsyncTask` class are defined in the table below:

Method	Function
<code>doInBackground()</code>	This method contains the code which will be executed in the background
<code>onPreExecute()</code>	This method contains the code which will be executed before the <code>doInBackground</code> method
<code>onProgressUpdate()</code>	This method receives progress updates from <code>doInBackground</code> method
<code>onPostExecute()</code>	This method is executed after <code>doInBackground</code> completes processing and the result from <code>doInBackground</code> is passed to this method

Table 7.1: AsynTask Methods

### 7.3.1 Sending Request to the Server

There are multiple networking libraries and classes available for Android that can send POST requests, however, the preferred method is through HttpURLConnection. This section will describe how to send request from Android client to the server;

**Giving Permission to Android:** Univent requires an internet connection in order to get information about the event and to retrieve data from the database. All required permissions must be declared in the AndroidManifest.xml file.

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Figure 7.7: Internet Permission

**Connecting to the Server using URL:** The URL object, refer to figure 7.8, contains all the necessary information to reach the destination resource. In this example, the url is pointing to the php file on the server, named get\_event.php. This file handles the query to get all events from the database. URLs are constructed as sets of information. Lets consider the URL:

**http://derton04.000webhostapp.com/login.php?parameter=value**

In the example above, http is the protocol, derton04.000webhostapp.com is the domain name, and login.php the path. Parameter a query parameter key (called key), value a query parameter value (named as value), and parameter=value a query parameter key-value pair (referred to as key-value pair).

```
String reg_url = "https://derton04.000webhostapp.com/login.php";
URL url = new URL(reg_url);
HttpURLConnection = (HttpURLConnection) url.openConnection();
```

Figure 7.8: URL to get all Events in the Database

In the next step, 7.9, the methods and properties of the request object are set. First, set the method as request method to be invoked as POST. The method setDoOutput, needs to be set to true before sending a post request. setDoOutput is not needed for GET requests.

```
httpURLConnection.setRequestMethod("POST");//meThod to transfer the data
httpURLConnection.setDoOutput(true);
OutputStream os = httpURLConnection.getOutputStream(); //allows us to send data in stream
BufferedWriter bufferedWriter = new BufferedWriter(new OutputStreamWriter(os, charsetName: "UTF-8"));
String data = URLEncoder.encode( $: "email", enc: "UTF-8") + "=" + URLEncoder.encode(email, enc: "UTF-8") + "&" +
URLEncoder.encode( $: "password", enc: "UTF-8") + "=" + URLEncoder.encode(password, enc: "UTF-8");
bufferedWriter.write(data);
bufferedWriter.flush();
bufferedWriter.close();
os.close();
```

Figure 7.9: URL to get all Events in the Database

This library also gives the possibility of including data in the request. The URLConnection class provides an OutputStream object as the mechanism for providing this data, for performances reason it is wrapped in a buffered writer. The parameters/data are then attached to the url, then encoded onto the outPutStream before being transmitted over the internet.

The snippet below, figure 7.10 checks if the connection was successful and if the data has been transmitted.

```
int response_code = HttpURLConnection.getResponseCode();

// Check if successful connection made
if (response_code == HttpURLConnection.HTTP_OK) {
```

Figure 7.10: Check if Connection is OK

The figure below, 7.11 shows the status code of an http connection.

```
HttpURLConnection.HTTP_OK          // HTTP Status-Code 200: OK
HttpURLConnection.HTTP_NOT_FOUND  // HTTP Status-Code 404: Not Found
HttpURLConnection.HTTP_NOT_IMPLEMENTED // HTTP Status-Code 501: Not Implemented
```

Figure 7.11: Status Code

After receiving the Http status code OK, the URL will return some values that are directly coming from the database. Using inputStream and the String builder class, will fetch this data(JSON format) and store it in the variable, in this case result.

```
// Read data sent from server
InputStream input = HttpURLConnection.getInputStream();
BufferedReader reader = new BufferedReader(new InputStreamReader(input));
StringBuilder stringBuilder = new StringBuilder();
String line;

while ((line = reader.readLine()) != null) {
    stringBuilder.append(line + "\n");
}
input.close();

String result = stringBuilder.toString();
```

Figure 7.12: Status Code

Figure 7.13, shows how the data is parsed using JSON parsing technique inside android application and then set into recyclerview. To access the database, it requires an

```

JSONArray jsonArray = new JSONObject(result).getJSONArray( name: "response");
// JSONArray jsonArray = new JSONArray(result);
JSONObject jsonObject = null;
int id;
//Data = new String[j jsonArray.length()];
for (int i = 0; i < jsonArray.length(); i++) {

    jsonObject = jsonArray.getJSONObject(i);
    userID = String.valueOf(jsonObject.getString( name: "userid"));
    userName = jsonObject.getString( name: "name");
    userEmail = jsonObject.getString( name: "Email");
    error = jsonObject.getString( name: "error");
    unique_hash = jsonObject.getString( name: "unique_hash");
    message = jsonObject.getString( name: "message");
}

```

Figure 7.13: Parsing JSON

interface, first, it must receive JSON data, add data and update the database, secondly, it must deliver JSON data containing events in the database. A PHP scripts is used to provide this interface, below is a complete example of connecting an android application with MYSQL database via PHP script. It creates a basic application that retrieves data from MYSQL database using GET and POST method.

The PHP page given below takes parameters(userid) by POST method, and displays all event in the database created by that userID. The result will be displayed in JSON format. The PHP file is responsible for handling the communication with the database,

```

/public_html/get_my_event.php
1  <?php
2
3  require "init.php";
4
5
6  $u(userID) = $_POST["userID"];
7
8  | $query = mysqli_query($connection, "SELECT * FROM Event WHERE userID = '$u(userID)'");
9
10 | if($query)
11 | {
12 |   while($row = mysqli_fetch_array($query))
13 |   {
14 |     $flag[] = $row;
15 |   }
16 |
17 |   print(json_encode(array("response"=>$flag)));
18 |
19 | }
20 mysqli_close($connection);
21 ?
22 ?>

```

Figure 7.14: PHP Page

to insert, update, retrieve and delete data. To do so, the query is embedded in the PHP file which is then capable of establishing the connection with the MySQL database and execute the required query.

**JSON - JSON (JavaScript Object Notation)** is a lightweight text-data interchange format used to transmit data in form of objects(structured data: key value pairs) over the internet. JSON also supports string, Boolean, number, array and null formats for storing data. After the database operation is done, the server will save the result as JSON format, and send it to the requesting client, refer to Appendix C for an example of JSON data format.

## 7.4 Sending Image to Server

This section explains how the images of each event were uploaded to the server using the Volley library, see Appendix F.

## 7.5 Testing

To ensure the system was tested thoroughly a test suit was designed. Test results are shown in Appendix D.

### 7.5.1 System Testing

The purpose of system testing is to test Univent against the functional requirements declared in section 3. Appendix D detail the system testing.

### 7.5.2 Black Box Testing

Android app must be tested on the emulators, but it is important to test the app in the real world. The app has been constantly been tested from initial installation to daily use of app as per the end users point of view. This method is more powerful to come to know if any issues become visible in Android apps, as issues will surface in day to day use.

After each new functionality added, a thorough testing is done by the author, the supervisor and some experimental users. This method resulted in being very effective as some defects, with varying severity, were identified, for example, the send email button in the event information page, was meant to send an email to the event host, instead it was sending the email to the actual user. Another behaviour identified that was very crucial, was the login validation, user were allowed to register with a blank password, therefore they could login without typing in the password.

## 7.6 Publishing to Play Store

After all testing has been completed, the application was published on the Google Play Store. The app was released on the Android Market under the name Univent and it will be free for all users.

# **Chapter 8**

## **Evaluation**

### **8.1 Overview**

As usability is quite crucial for the success of the application, a random selection of users were offered to try the application. Once they had tested the app they were asked to leave a review on the play store and fill out a survey. Below is the outcome of the survey and some of the reviews.

Question	Score
How would you rate the mobile app?	Weighted average is 4.75 out of 5 stars
I found the system really easy to use	75% Strongly agree, 25% Agree
I would need assistance to be able to use the app.	50% neither agree or disagree, 25% disagree and the other 25% strongly disagree
How visually appealing is the app?	50% Extremely appealing, 25% Very appealing and the other 25% Somewhat appealing
I think I would use the app frequently	75% agree and 25% strongly agree
How likely is it that you would recommend the app to a friend or colleague?	50% Extremely likely and 50% very likely
Do you have any other comments about how we can improve the app?	see table 8.2

Table 8.1: Survey

User	Comments
User 1	Good app, allows you to view events from Bournemouth University, allows you to add your own events, simple to use nice layout
User 2	Very good app, simple user friendly interface
User3	Create a version compatible with apple iPhones
User 4	Some usability issues when you verify your email address, I had to re enter Bournemouth University before i could sign in, I didnt know I had to do that without guidance
User 5	Pretty good app. Perfect for checking available BU uni events - convenient use. Layout is nice and simple. Good simple app

Table 8.2: User Comments

For general feedback, refer to table 8.1 and 8.2. The screen design received very good feedback; the participants noticed how the screens were appealing and they also noticed how useful the artefact could be.

### 8.1.1 Requirements Evaluation

In terms of functionality all the requirements mentioned in chapter 3 have been met.

Requirements	Met? Yes/No
The application should have a register and login screen	Yes
The application should integrate a validation for email, only Universities email are accepted	Yes
The application should provide specific information such as event location, event start and end date, description, title.	Yes
The application should provide information of all attendees interested in an activity or event	Yes

The application should have a “Share” feature. This will open any related app installed on the users phone so they can share the event with friends	Yes
The application should provide a “Send Email” feature. This will open any email client installed on users phone, so they can email the event host for further information regarding the event	Yes
The application should have a “Map Feature”. This will direct the user to the Maps application with the event location already pre entered.	Yes
The application should have an “Add to Calendar” feature. This will open any calendar application on the users phone and it will allow one to save the event on the calendar creating a future reminder.	Yes
The application should allow the user to choose the category they are most interested in and based on that information display all types of event or activities in those categories	Yes
The application should save the user login details, so user is only required to login once.	Yes
The application should show the location on a map directly on the screen of the app	Yes

Table 8.3: Requirements Evaluation

## 8.2 Success Criteria

The aim of this thesis was to provide support in publishing and keeping track with activities in the faculty. The success criteria set in section 1.11, has been successfully met and all requirements have been implemented.

The app was built using Android studio, the Android framework and API were discussed in chapter 4, that knowledge aid in implementing the app successfully.

Overall this project was successful, all feedback regarding the system received from various users indicates users were able to use the app successfully and were able to create an event as well as keep track of any other activities posted on the app.

# **Chapter 9**

## **Conclusion**

### **9.1 Summary**

The artefact delivers exactly what it was built for, which was to build an android app that encourages users to use the app as a platform to share events. A server side was created as a way to communicate between server and client, on the client side HttpURLConnection and Volley were used to retrieve and send data to the server. Both solutions were adopted because of their efficiency and simplicity in being implemented, ignoring other existing solutions like okhttp and retrofit.

The app allows users to publish their own events and view any other upcoming events on the campus. This makes publishing very easy for users, and individuals can find all information needed about an event on the go.

The feedback received from users was very positive, and majority of them will definitely consider using the App in their daily life.

This project will contribute in going paperless. Which will ensure cost savings for the University and will enable efficient communication and collaboration among individuals. Univent will definitely make an impact. For Universities to be efficient they have to adopt similar solutions which align with environmental friendliness.

The App met all requirements, however during testing and evaluation, further ideas were introduced, which will be implemented depending on future funding.

### **9.2 Future Works**

Building an app, is a very complex task and time consuming process. Apps need to be improved and updated constantly to entice users. Even if the main requirement were met and the app has been published, there are always room for improvement.

As mentioned in section 1.4, integrating the mobile application with the University, will depict that the app is useful and can make a change to how events and activities are

published and accessed in the university. This is definitely a challenge, however it will be a great achievement if it was possible to work alongside the university.

Future implementation could be:

**Database Security** - Most of the information of the user is stored in the backend databases. One of the main vulnerabilities is SQL (Structured Query Language) injection attack. SQL injection attack is one of the main vulnerabilities and prevalent database attacks. The attacker, by exploiting the application and database she/he can get unauthorized access to the database and cause harm. A solution to this problem will be to use prepare statement in the server side before executing the query.

**Include Gamification** - Including a gamification system to the app might motivate users to use the mobile app to publish events and go greener, refer to section 5.1 about gamification.

**Push Notification** - push notifications will remind users about any upcoming event they are interested in and also remind them of any update about any event they are attending. One good side of having push notification is that it reminds users of your app and improving the chances of the app remaining installed on their devices. Google Cloud Messaging(GCM), is a free service used to send push notifications to users and ensures notifications are delivered securely and reliably.

**Follow Friends** - Users should be able to follow friends and also get event notification on what their friends are interested in.

**Keep User Updated** - Users should be notified when they decide to attend an event; a list of all essential information about that event should be sent to the users email and the users should be kept updated about any changes on the event. For example if the event is cancelled for any reason the user should be notified about the cancellation.

**Extend to iOS** - When most of the features have been implemented on the android platform, it is essential to have a stable app on one platform before implementing it on another, to avoid any drawbacks, the app can finally be extended to the iOS platform to reach more users.

**Word Count: 9940**

# Bibliography

- d. A. Souza, M. R., Constantino, K. F., Veado, L. F. and Figueiredo, E. M. L., 2017. Gamification in software engineering education: An empirical study. *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE T)*, 276–284. [Accessed on 05/05/2018].
- Araújo, I. and Carvalho, A. A., 2017. Empowering teachers to apply gamification. *2017 International Symposium on Computers in Education (SIIE)*, 1–5. [Accessed on 05/05/2018].
- Beal, V., no date. What is api - application program interface? webopedia. <https://www.webopedia.com/TERM/A/API.html>. [Accessed on 23/04/2018].
- Bedyński, P., 2011. Andood – an android application. [Accessed on 22/04/2018].
- Deci, E. L. and Ryan, R. M., 2008. Self-determination theory: A macrotheory of human motivation, development, and health. *Canadian psychology/Psychologie canadienne*, 49 (3), 182. [Accessed on 05/05/2018].
- Deterding, S., Sicart, M., Nacke, L., O'Hara, K. and Dixon, D., 2011. Gamification. using game-design elements in non-gaming contexts. *CHI'11 extended abstracts on human factors in computing systems*, ACM, 2425–2428. [Accessed on 05/05/2018].
- D'silva, K., Vanajakshi, J., Manjunath, K. N. and Prabhu, S., 2017. An effective method for preventing sql injection attack and session hijacking. *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 697–701. [Accessed on 05/05/2018].
- Fitzgerald, B., Hartnett, G. and Conboy, K., 2006. Customising agile methods to software practices at intel shannon. *European Journal of Information Systems*, 15 (2), 200–213. [Accessed on 22/04/2018].
- Fundamentals, A., no date. Application fundamentals — android developers. <https://developer.android.com/guide/components/fundamentals.html>. [Accessed on 24/04/2018].

- Goyal, G., Singh, K. and Ramkumar, K. R., 2017. A detailed analysis of data consistency concepts in data exchange formats (json xml). *2017 International Conference on Computing, Communication and Automation (ICCCA)*, 72–77. [Accessed on 22/04/2018].
- Hamari, J., Koivisto, J. and Sarsa, H., 2014. Does gamification work?—a literature review of empirical studies on gamification. *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, IEEE, 3025–3034. [Accessed on 05/05/2018].
- Hamed, A. M. M. and Abushama, H., 2013. Popular agile approaches in software development: Review and analysis. *2013 INTERNATIONAL CONFERENCE ON COMPUTING, ELECTRICAL AND ELECTRONIC ENGINEERING (ICCEEE)*, 160–166. [Accessed on 22/04/2018].
- Hanus, M. D. and Fox, J., 2015. Assessing the effects of gamification in the classroom: A longitudinal study on intrinsic motivation, social comparison, satisfaction, effort, and academic performance. *Computers & Education*, 80, 152–161. [Accessed on 05/05/2018].
- Hollar, A. B., 2006. Cowboy: An agile programming methodology for a solo programmer. [Accessed on 04/22/2018].
- Houston, D., 1997. Is paperless painless? *Proceedings of the 25th Annual ACM SIGUCCS Conference on User Services: Are You Ready?*, New York, NY, USA: ACM, SIGUCCS '97, 141–145. URL <http://doi.acm.org/10.1145/266064.266111>, [Accessed on 01/05/2018].
- How....., L., no date. Learn how to test your android application? <http://www.softwaretestingclass.com/learn-how-to-test-your-android-application/>. [Accessed on 29/04/2018].
- Isaeva, M. and Yoon, H. Y., 2016. Paperless university x2014; how we can make it work? *2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET)*, 1–8. [Accessed on 01/05/2018].
- Jonnalagadda, S., 1913. *Android application for library resource access*. Ph.D. thesis, Sciences. [Accessed on 22/04/2018].
- Kaplan, A. M. and Haenlein, M., 2010. Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53 (1), 59–68. [Accessed on 05/05/2018].
- Lee, G. and Xia, W., 2005. The ability of information systems development project teams to respond to business and technology changes: a study of flexibility measures. *European Journal of Information Systems*, 14 (1), 75–92. [Accessed on 22/04/2018].

- Lin, Y. and Dig, D., 2015. Refactorings for android asynchronous programming. *2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 836–841. [Accessed on 22/04/2018].
- Malik, M. and Patel, T., 2016. Database security-attacks and control methods. *International Journal of Information Sciences and Techniques (IJIST)*, 6 (1/2). [Accessed on 07/05/2018].
- Maruping, L. M., Venkatesh, V. and Agarwal, R., 2009. A control theory perspective on agile methodology use and changing user requirements. *Information Systems Research*, 20 (3), 377–399. [Accessed on 22/04/2018].
- Maxim, B. R., Brunvand, S. and Decker, A., 2017. Use of role-play and gamification in a software project course. *2017 IEEE Frontiers in Education Conference (FIE)*, 1–5. [Accessed on 05/05/2018].
- Mehdi, M., 2013. Timespan-museum without walls mobile application. *School of Computer Science, University of Glasgow*. [Accessed on 22/04/2018].
- mobiloud, no date. Web, hybrid or native apps? what's the difference? <https://www.mobiloud.com/blog/native-web-or-hybrid-apps/>. [Accessed on 23/04/2018].
- Ngim, R., 2013. Yale going paperless to save money time and trees, monday morning news, may 20, 2013, volume 7, no. 19. <https://its.yale.edu/news/mmn/v7n19>. [Accessed on 01/05/2018].
- Noprianto, Soewito, B., Isa, S. M., Iskandar, K., Gaol, F. L. and Kosala, R., 2017. Server for sqlite database: Multithreaded http server with synchronized database access and json data-interchange. *2017 19th International Conference on Advanced Communication Technology (ICACT)*, 786–790. [Accessed on 22/04/2018].
- Novac, O. C., Novac, M., Gordan, C., Berczes, T. and Bujdosó, G., 2017. Comparative study of google android, apple ios and microsoft windows phone mobile operating systems. *Engineering of Modern Electric Systems (EMES), 2017 14th International Conference on*, IEEE, 154–159. [Accessed on 22/04/2018].
- Powell-Morse, A., 2017. Extreme programming: What is it and how do you use it? <https://airbrake.io/blog/sdlc/extreme-programming>. [Accessed on 22/04/2018].
- Rapoport, M., Suter, P., Wittern, E., Lhotak, O. and Dolby, J., 2017. Who you gonna call? analyzing web requests in android applications. *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, 80–90. [Accessed on 22/04/2018].

- Rohilla, S. and Mittal, P. K., 2013. Database security: Threats and challenges. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3 (5). [Accessed on 06/05/2018].
- Rouse, M., 2016. What is integrated development environment (ide)? - definition from whatis.com. <https://searchsoftwarequality.techtarget.com/definition/integrated-development-environment>. [Accessed on 23/04/2018].
- Sitorus, M. B., Ferdiana, R. and Adji, T. B., 2017. Designing gamification framework to support social media application based on game elements and cutting-edge technology. *2017 International Conference on Electrical Engineering and Computer Science (ICECOS)*, 125–130. [Accessed on 05/05/2018].
- smartsheet, 2016. Full comparison: Agile vs scrum vs waterfall vs kanban. <https://www.smartsheet.com/agile-vs-scrum-vs-waterfall-vs-kanban>. [Accessed on 22/04/2018].
- Stallings, W., 2001. Operating systems: Internals and design principles, prentice hall, 2000, isbn 0-13-031999-6 it is available at williamstallings. com/os4e. html. [Accessed on 22/04/2018].
- Techopedia, no date. What is java development kit (jdk)? - definition from techopedia. <https://www.techopedia.com/definition/5594/java-development-kit-jdk>. [Accessed on 23/04/2018].
- Uskov, V. and Sekar, B., 2014. Gamification of software engineering curriculum. *Frontiers in Education Conference (FIE), 2014 IEEE*, IEEE, 1–8. [Accessed on 05/05/2018].
- Yusoff, A. M. and Sidhu, M. S., 2001. Paperless in the electronic era: millennium dream and reality. *Computer Supported Cooperative Work in Design, The Sixth International Conference on, 2001*, IEEE, 536–541. [Accessed on 01/05/2018].
- Ziesemer, A., Müller, L. and Silveira, M., 2013. Gamification aware: Users perception about game elements on non-game context. *Proceedings of the 12th Brazilian Symposium on Human Factors in Computing Systems*, Porto Alegre, Brazil, Brazil: Brazilian Computer Society, IHC '13, 276–279. URL <http://dl.acm.org.libezproxy.bournemouth.ac.uk/citation.cfm?id=2577101.2577164>, [Accessed on 05/05/2018].

# **Appendices**

## A Screen Design

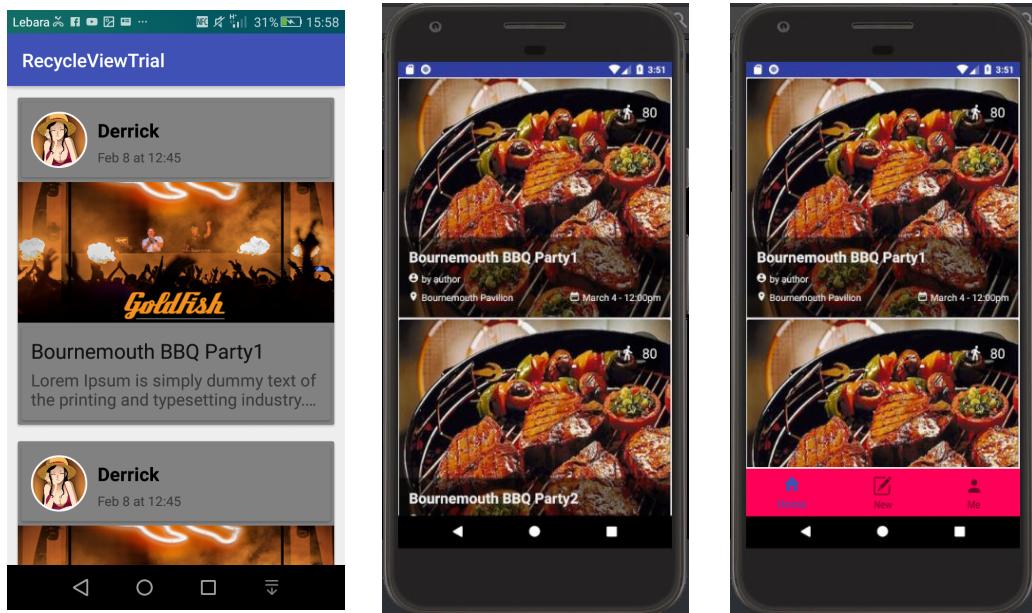


Figure 1: Initial Home Screen Design

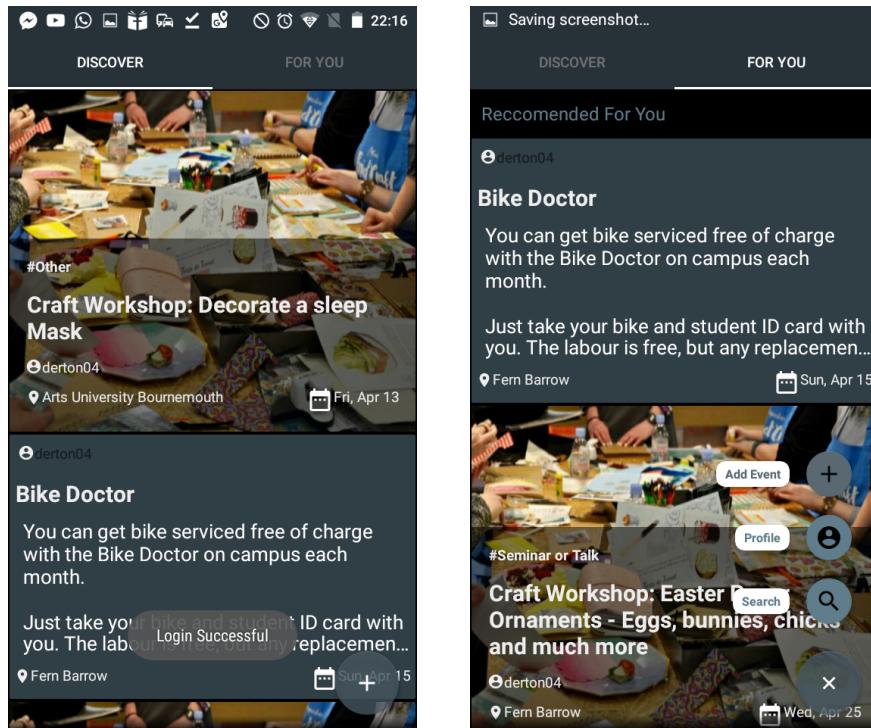


Figure 2: Final Home Screen Design

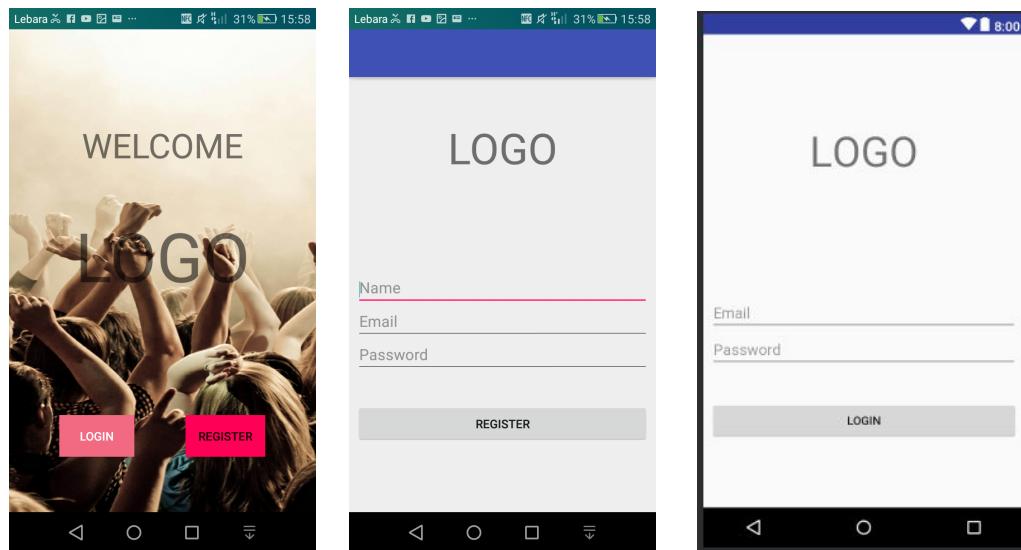


Figure 3: Initial Register & Login Screen Design

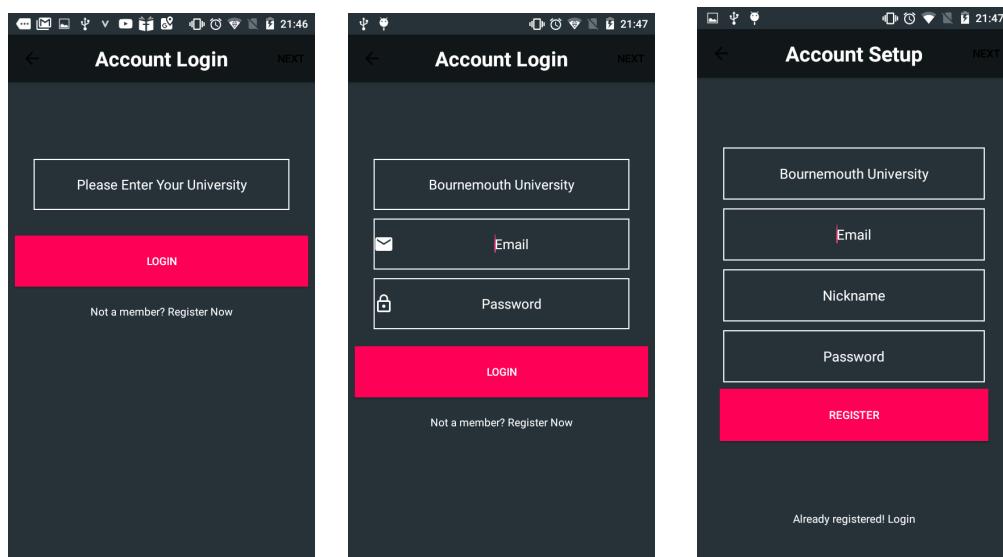


Figure 4: Final Register & Login Screen Design



Figure 5: Menu Clicked Screen

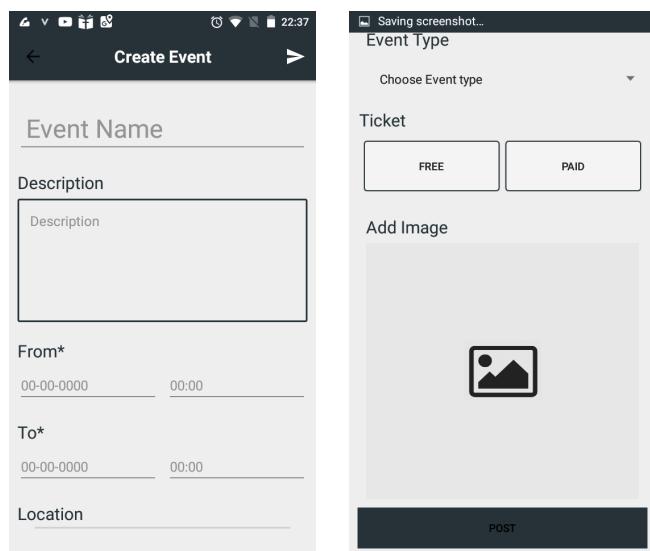


Figure 6: Add Event Screen

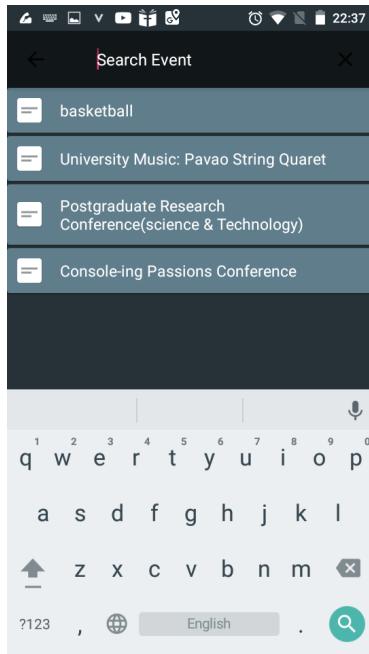


Figure 7: Search Event Screen

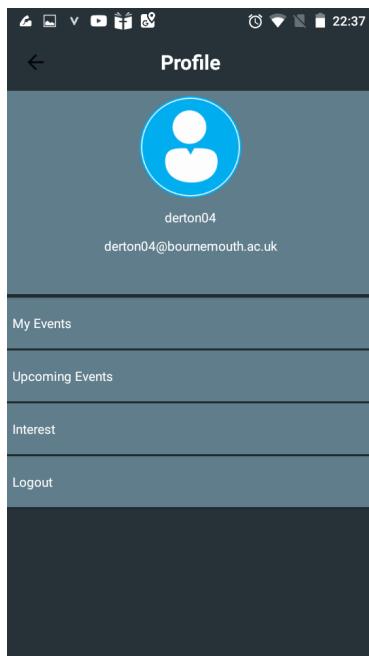


Figure 8: Profile Screen

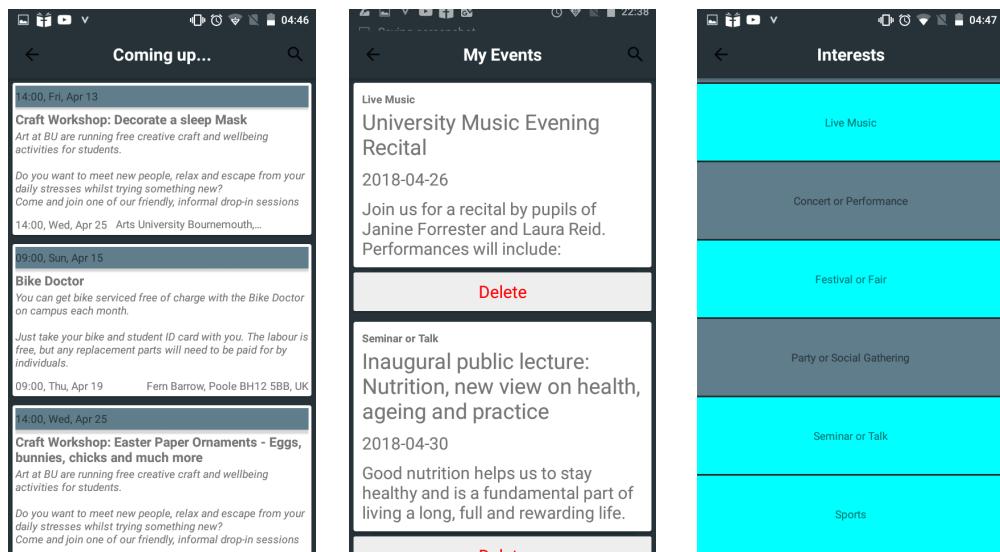


Figure 9: Upcoming, My Events & Choose Interest Screen

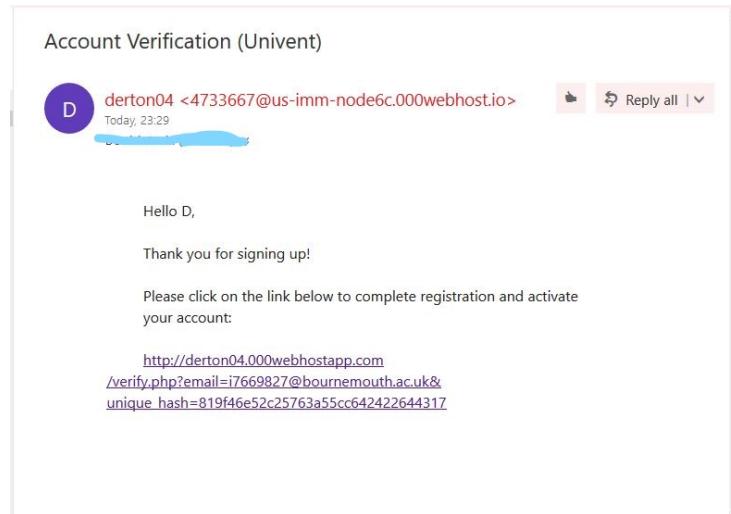


Figure 10: Email Verification

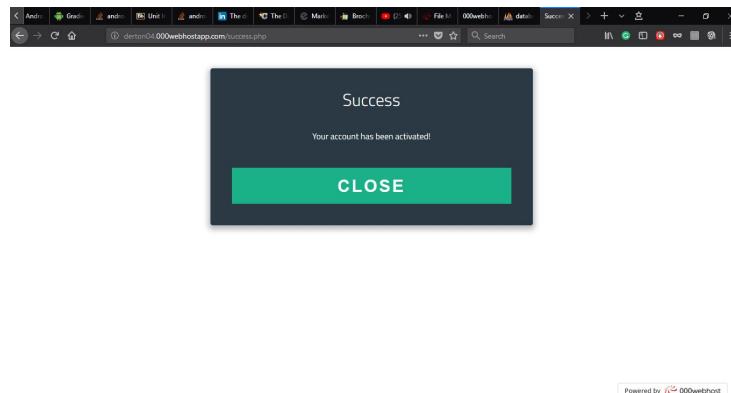


Figure 11: Account Verification Success

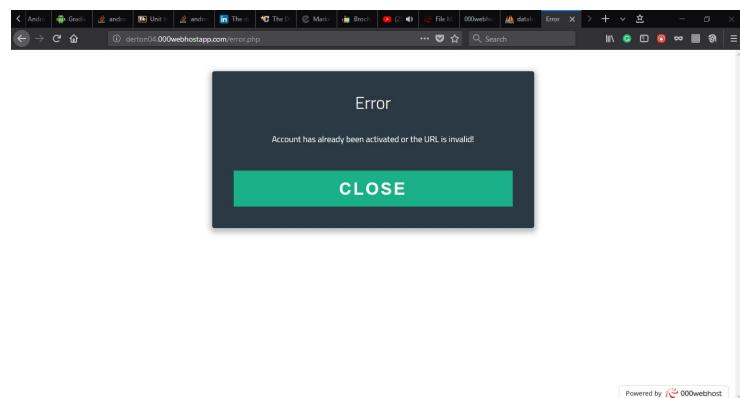


Figure 12: Account Verification Error

## B Android vs iOS

The operating system(OS) is a program that controls the execution of application and acts as an interface between a computer user and the computer hardware. The OS has these main objectives; first it has to make a computing system easy to use, secondly it has to execute programs and facilitate troubleshooting for user; and third it has to efficiently use the hardware of the computing system; Stallings (2001) cited in Novac et al. (2017).

With the introduction of iOS and Android by Apple and Google, the way of thinking about operating system completely changed. Android is by far the leading mobile platform, currently holding 86% of the market shares against iOS holding of 12% - 20%.

One of the major differences between these two is the devices they run on. Android, since based on linux, runs on a variety of devices, while iOS can only run on Apple manufactured devices.

The Android OS, is an open source platform, meaning it has the ability to run third party tools, so users can add more functionality enhance performance and add more features to it, while Apple is more restricted, even though both operating systems and final products are maintained and developed by the same company.

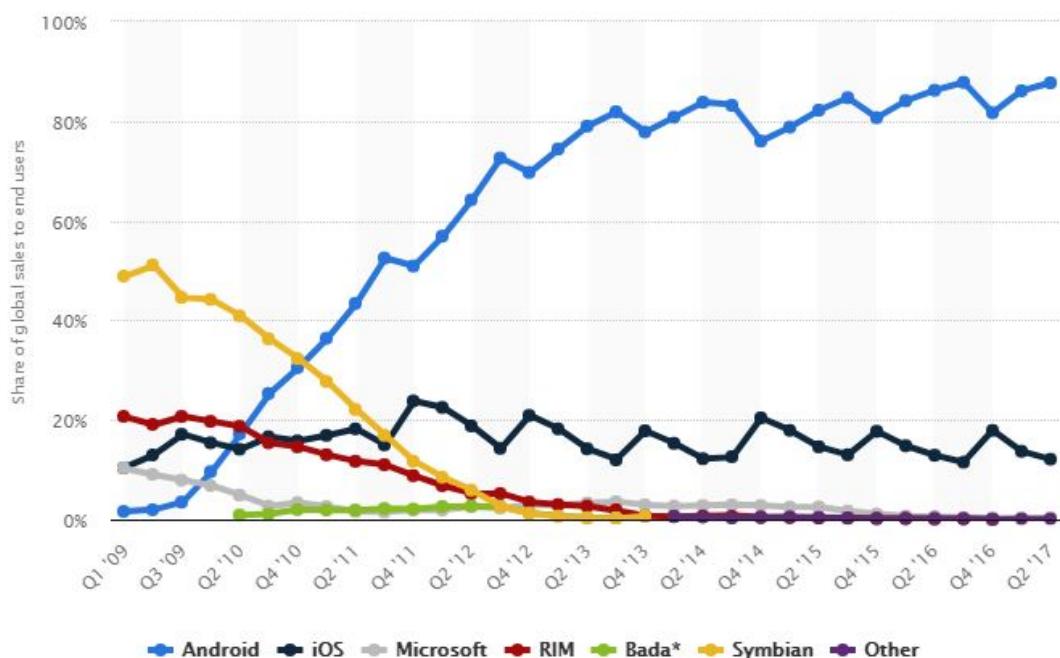


Figure 13: OS Global Market Share, (Data collected from Statista)

The idea behind this project is to build an Android mobile application which will help people in Universities to have easy access to events activities on the faculty. Android platform was chosen, as the leading OS on the market.

## C JSON Data Format

```
{"response": [{"eventid": "172",
  "event_name": "basketball",
  "start_date": "2018-05-03",
  "start_time": "01:00:00",
  "end_date": "2018-05-06",
  "end_time": "20:00:00",
  "Description": "",
  "location": "",
  "hasImage": "1",
  "event_type": "Sports",
  "ticket": "Free",
  "ticket_url": "null",
  "name": "name",
  "Email": "email@bournemouth.ac.uk"}]
```

Figure 14: JSON Format

## D Test Suit

Test NO	Purpose	Test Method	Expected Result	Actual Result	Outcome
1	To check if user Registration works correctly	User should be able to register	User Details should be added to the User database	As expected	Passed
2	To check if email validation works correctly	Enter a non-Bournemouth university email	Appropriate error message should show N.B Email domain must be of the chosen faculty (e.g. University of Bournemouth, Email must be 123456@bournemouth.ac.uk)	As expected	Passed
3	To check if user login works correctly	Click login, with all fields entered	User details should be retrieved from the database and verified accordingly	As expected	Passed
4	Check if user login details are saved	Requires test above, to be passed. Close app reopen app	App should take user to home screen, without asking for login details	As expected	Passed
5	Check if event information screen works correctly	Press on any event showing on the home screen	Event information screen should open with all related information of the event.	As expected	Passed
6	Check if number of people attending works correctly	Open even information screen	Number of attendees should display	As expected	

7	Check if attendee screen works correctly	Pressing on attendee button	Attendee screen should display with the actual names of attendees.	As expected	Passed
8	Check if add to calendar button works correctly	Press add to calendar button on event information screen	Open calendar on the user's phone with all event information pre-inserted. N.B If user has more than one Calendar app, a screen with all the apps should open and user must choose.	As expected	Passed
9	Check open maps button works correctly	Press open maps button on the event information screen	Maps app should open with the event location pre-set. N.B If user has more than one Map app, a screen with all the apps should open and user must choose.	As expected	Passed
10	Check if send email button works correctly	Press send email button on information screen	Email app on the phone should open with the event host email as entered as the receiver. N.B If user has more than one Email app, a screen with all the apps should open and user must choose.	As expected	Passed

111	Check if get event ticket works correctly	Press on get ticket N.B Get ticket button only appears, if the event is paid, otherwise, if the event is free, it will not appear.	Website to purchase event ticket should open in the browser. N.B If user has more than one browser app, a screen with all the apps should open and user must choose.	As expected	Passed
121	Check if suggested event works correctly	Press on any event from the suggesting screen at the bottom of the event information screen	Should open event information screen of that selected event	As expected	Passed
13	Check if moving from discover screen to for you screen works correctly	Swipe right and left	Swiping right from the discover screen should open the for you screen, swiping left from the for you screen should open the discover page	As expected	Passed
14	Check if the menu button works correctly	Press on the menu button on the home screens	Menu button should expand showing 3 other options, Search, Add Event and Profile	As expected	Passed
151	Check if profile option works correctly	Press on profile on the menu	Should open profile screen	As expected	Passed

16	Check if Add Event option works	Press on Add Event on the menu	Should open Add event screen	As expected	Passed
17	Check if Search option works correctly	Press on Search option on the menu	Should open the search screen	As expected	Passed
18	Check if validation of empty field works on the add event screen	Press on Post Event, without inserting any details	Appropriate message should display informing user to enter all required fields	As expected	Passed
19	Check if post event button works correctly	Enter all required field and press on post event button	Should add event to database and appropriate message should display	As expected	Passed
20	Check if interest screen works correctly	Click on interest on the profile screen	Interest screen should open	As expected	Passed
21	Check if event recommendation works correctly	Click on a category on interest screen, go back to recommendation screen	Recommendation screen should show event in the category selected	As expected	Passed
22	Check if option delete event on my event screen works properly	Click on delete button	Should delete the selected event from the database	As expected	Passed

23	Check if logout button works correctly	Press on logout button	User should be log out and redirected to login screen	As expected	Passed
----	--	------------------------	---	-------------	--------

## E Android Framework & API

Android is built on a framework and an API. According to webopedia an API is “an application program interface (API), set of routines, protocols, and tools for building software applications. Basically, an API specifies how software components should interact. Additionally, APIs are used when programming graphical user interface (GUI) components. A good API makes it easier to develop a program by providing all the building blocks. A programmer then puts the blocks together”.

A framework provides an interface for Android Apps to access the system resources. As an example, LocationManager is presented in this layer to support Android apps retrieving GPS coordinates of a device. It consists of tools for designing UIs like buttons, text fields, and system tools like intents.

### E.1 Application Component

According to the Android developers website, application components are the essential building blocks of an Android app. Each component is an entry point through which the system or a user can enter your app. These components are loosely coupled by the application manifest file `AndroidManifest.xml` that describes each component of the application and how they interact.

There are four different types of app components:

- Activities.
- Services.
- Broadcast receivers.
- Content providers.

Each type serves a distinct purpose and has a distinct lifecycle that defines how the component is created and destroyed. The following sections describe the four types of App components.

### E.2 Main Application Component

- **Activity** - This is the visible part of the Android application, it manages the views that define what the screen looks like and how the user can interact with it. The activity has a specific life cycle as shown in figure 15. When an activity is launched, it is brought up on the activity stack, and made visible. Oncreate, OnStart, OnResume will be called in the process. The activity from which the new activity is launched will be signalled with Onpause or OnStop. Lastly, one activity can be used to call another; a call to finish() on the activity removes it from the stack, this

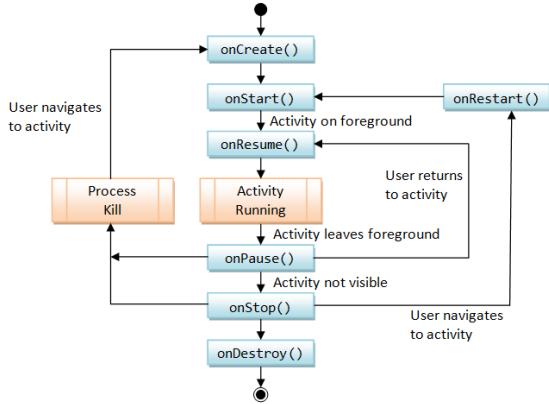


Figure 15: Activity Lifecycle

is done in combination with `startActivity`, so the new activity will replace the current one instead of going on top of it.

- **Service** - is a running component, that doesn't have an interface but has a connection with the activity. It is a component that runs in the background to perform complicated and long tasks, for example, playing music and or while continually checking for new data.
- **Broadcast Receivers** - Broadcast receivers, simply handles communication between the system and the application. For example, a broadcast announcing that the screen has turned off or the battery is low. Apps can also initiate broadcasts, for example, to let other apps know that data has been downloaded to the device and is available for them to use.
- **Content Provider** - Content provider handles data sharing among applications on request or to save some data. Such requests are dealt with by methods in the content provider class.

### E.3 Additional Components

- **Intent** - Most of the activities are activated by an asynchronous message called an intent. Intent manages interactions between application components across the whole system. They are like messengers that request an action from other components, whether the component belongs to the app or another. An interesting example is the Map application that handles Intents request to display specific GEO location.

Intent mechanism makes it easy to reuse different system components across the platform. They can focus on really innovative and unique functionality and simply

add ('attach') new components like maps or navigation to their applications which makes them even more compelling and useful, Bedyński (2011).

- **Manifest** - Every Android application has its own AndroidManifest.xml file in the root directory. This file lists all the activities and defines the properties of the application, for example the permission to the Android platform to access the internet.

## F Sending Image to Server

This section will explain how the event images were uploaded to the server using the Volley library. Volley is a networking library developed by Google, easy to use, implement and memory efficient. It makes networking faster and easier for apps. The volley library has the features like automatic scheduling of network request, request prioritisation, cancel/block a request. Volley uses cache to improve the performance by saving memory and bandwidth of remote server, for example, when user request the same data, instead of calling again from the server, Volley will directly load it from cache, saving resource and improving user experience. To upload the image to the server, it has to be encoded as a base64 encoded string in android. Uploading images to server in android is time consuming process and as well as the code is also very complex. The image is converted into Base64 string format and sent to server using volley network library and the server side will decode that image through the PHP file and finally save the image to the folder in the server(to easy fetch the image from the server, the event title is used as the image name on the server, this process of adding the image is done synchronously while uploading a new event to the database.). Uploading image as base64 reduces the time of uploading and complexity of the code.

The image is fetched from the server using the glide library. According to Github, “Glide is an Image Loader Library for Android developed by bumptech and is a library that is recommended by Google. It has been used in many Google open source projects including Google I/O 2014 official application. It provides animated GIF support and handles image loading/caching.”

Each image is called synchronously, while the events are being fetched from the database, as to increase performance, so the image will be displayed together with each event. The snippet below shows how to fetch an image from the server. In the snippet

```
Glide.with(context).load("https://derton04.000webhostapp.com/MyImages/"  
+event.getEvent_name() + ".png").into(holder.eventImage);
```

Figure 16: Glide Usage

above, https is the protocol, derton04.000webhostapp.com , the domain name, and My-Images the path where the image file is saved and event.getEvent\_name() + "png", gets

the event name plus the file format png, into(holder.eventImage) is the imageView that displays the image.

## G Project Proposal

### BU Computing Programmes 2017-2018

#### Undergraduate Project Proposal Form

Please refer to the **Project Handbook Section 4** when completing this form

Degree Title:	Student's Name:
Software Engineering	Derrick Aryeetey Feehi
	Supervisor's Name:
	Deniz Cetinkaya
	Project Title/Area:
	Paperless Solution for Distributing Flyers

#### Section 1: Project Overview

##### **1.1. Problem definition - use one sentence to summarise the problem:**

Based on observation, students find it quiet hard to advertise any upcoming events in University, the only way is to distribute flyers around and most of these ends up in the bin, its quiet hard as well if the target is to reach as many students as possible.

##### **1.2 Background - please provide brief background information, e.g., client:**

Flyers can be labor intensive and costly to produce and distribute especially if the target is large and it's very hard sometimes to connect with the audience. The number of people reached by flyers are very limited, it depends on the amount of flyers you can actually distribute to the audience. The process is quiet long as well, from design, printing and finding the perfect location to reach as many people as possible.

##### **1.2. Aims and objectives – what are the aims and objectives of your project?**

This purpose of this project is to create a paperless solution, to prevent users from printing out flyers or brochures and make it easier for students to post any upcoming events and reach as many audience as possible. Lecturers can post any important events (external speaker etc...) and the University can post any important upcoming events. This project will make it much easier for people to connect and make each other aware of any events going on or upcoming events.

# BU Computing Programmes 2017-2018

## Section 2: Artefact

### **2.1: What is the artefact that you intend to produce?**

I intend to create a mobile application, where students will login with their University email address, then read or post any ongoing or upcoming events (club nights, parties, football matches, university events, external speakers, etc.....).

### **2.2 How is your artefact actionable (i.e., routes to exploitation in the technology domain)?**

The app will extend to all android mobile phones except from IOS. Based on research done, not many apps are out there on the market which can solve this issue. This app will be exclusively for BU students, so they can find it easier to reach other students, a way of connecting, meeting new people, making friends as well. A student can post an upcoming football match he wants to organize, others student seeing the post on the app can then decide to participate. This app being restricted to students only, is more reliable as it will reduce the probability of reaching out to complete strangers outside the university who might have bad intentions in mind.

# BU Computing Programmes 2017-2018

## Section 3: Evaluation

### 3.1 How are you going to evaluate your work?

This project is a mobile based application which will help students, lecturers share events among each other. Evaluation of the app will be based mostly on the user experience, early release/beta release of the app will be tested by students, where they will give a feedback on the performance of the app. Any feedback received will be then reviewed and all appropriate modification to the app will be applied.

### 3.2 Why is this project honours worthy?

This project reflects my ability to apply all the knowledge gained from my previous years in this course, programming has been essential throughout the entire course and all the knowledge and skills gained will be going into this project.

### 3.3 How does this project relate to your degree title outcomes?

This project is mainly software related, and it's also an idea that meets real needs and provides a quality and practical solution to solve it.

### 3.4 How does your project meet the BCS Undergraduate Project Requirements?

This project solves an ongoing issue, this idea will be creating a paperless solution for distributing/sharing flyers/events to every student.

### 3.5 What are the risks in this project and how are you going to manage them?

Risk	Lkelihood	Effect on Project	Risk Reduction Action
Time Constraint	H	Not able to complete project	Manage time, follow planned schedule
Computer Failure	L	Lose whole project	Make Backup
Not enough students for experiment (participants)	M	Not able to evaluate project properly	Early recruitment, different testing method
Interface might be low quality	M	Project might be of low quality	Continuous interaction with participants

# **BU Computing Programmes 2017-2018**

## **Section 4: References**

### **4.1 Please provide references if you have used any.**

[1] Bizfluent. 2017. The Pros & Cons of Advertising Fliers | Bizfluent. [ONLINE] Available at: <https://bizfluent.com/info-7747187-pros-cons-advertising-fliers.html>. [Accessed 15 October 2017].

[2] Print City Singapore. 2017. Pros and Cons of Printed Brochures vs. Online Brochures. [ONLINE] Available at: <https://www.printcity.com.sg/pros-and-cons-of-printed-brochures-vs-online-brochures/>. [Accessed 15 October 2017].

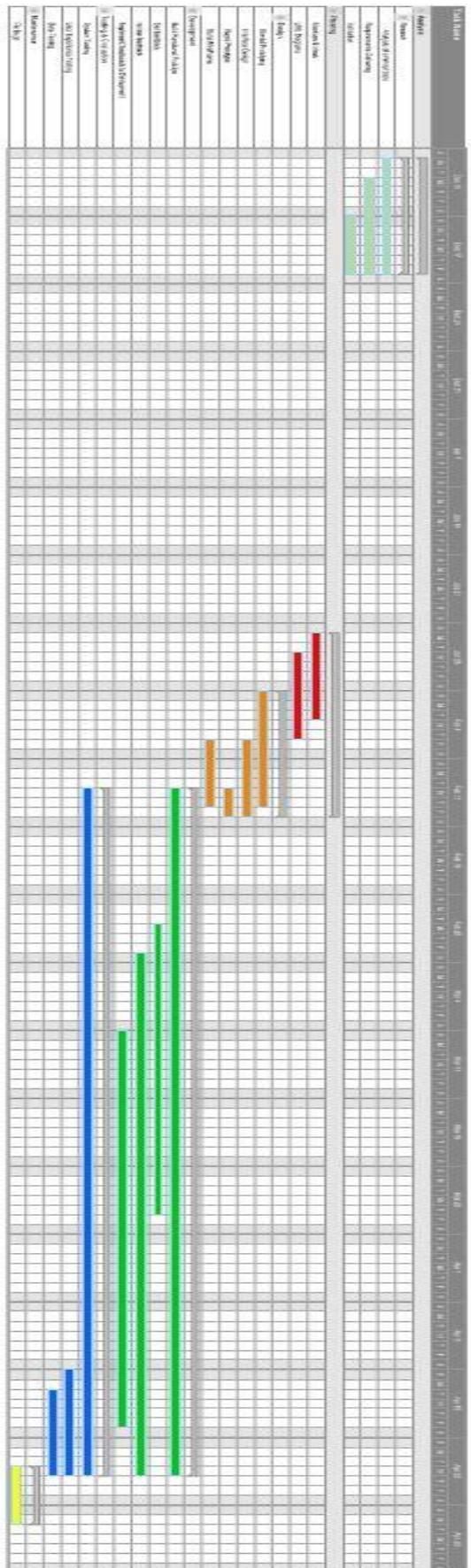
## **Section 5: Ethics (please delete as appropriate)**

**5.1 Have you submitted the ethics checklist to your supervisor?** Yes

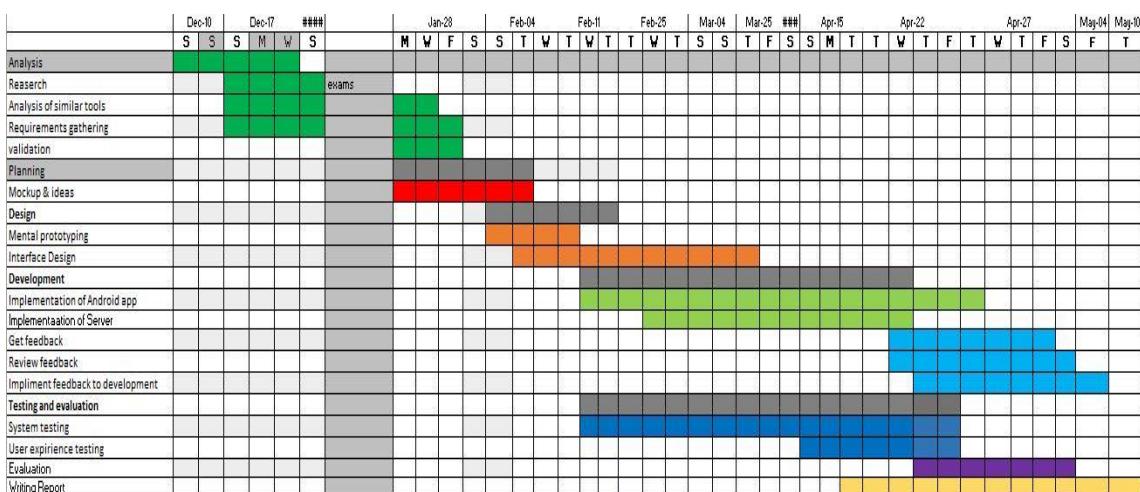
**5.2 Has the checklist been approved by your supervisor?** No

## **Section 6: Proposed Plan (please attach your Gantt chart below)**

**BU Computing Programmes 2017-2018**



## H Gantchart



## I Project Ethics



Bournemouth  
University

# Research Ethics Checklist

Adapted for the use by Department of Computing and  
Informatics ONLY

## 1. Student Details

Name	Derrick Aryeetey Feehi
School	Faculty of Science & Technology
Course	BSc Software Engineering
Have you received external funding to support this research project?	No
Please list any persons or institutions that you will be conducting joint research with, both internal to BU as well as external collaborators.	

## 2. Project Details

Title	Paperless Solution for Distributing Flyers
Proposed Start Date	29-January-2018
Proposed End Date	31-August-2018 – <b>Note this is not the submission deadline!</b>
Supervisor	Deniz Cetinkaya
<b>Summary (including detail on background methodology, sample, outcomes, etc.)</b>	
<p>Based on observation, students find it quiet hard to advertise any upcoming events in University, the only way is to distribute flyers around and most of these ends up in the bin and its quiet hard as well if the target is to reach as many students as possible. The purpose of this project is to create a paperless solution, to prevent users from printing out flyers or brochures and make it easier for students to post any upcoming events and reach as many audience as possible. Lecturers can post any important events (external speaker etc...) and the University can post any important upcoming events as well. This project will make it much easier for people to connect and make each other aware of any events going on or upcoming events.</p>	

# Research Ethics Checklist

Adapted for the use by Department of Computing and Informatics ONLY

## 3. External Ethics Review (Answer “Yes” go to 4, “No” go to 5)

Does your research require external review through the NHS National Research Ethics Service (NRES) or through another external Ethics Committee?	No
--	----

## 4. External Ethics Review Continued

Answered “Yes” to question 3 will conclude the BU Ethics Review so you do not need to answer the following questions. Note you will need to obtain external ethical approval before commencing your research.

## 5. Research Literature (Answer “Yes” go to 6, “No” go to 7)

Is your research solely literature based?	No
---	----

## 6. Research Literature Continued (Either answer will conclude the review)

Will you have access to personal data that allows you to identify individuals OR access to confidential corporate or company data (that is not covered by confidentiality terms within an agreement or by a separate confidentiality agreement)?	Yes
Describe how you will collect, manage and store the personal data (taking into consideration the Data Protection Act and the Data Protection Principles).	
I will collect personal data by gaining consent for collecting and processing the data from the individual and also provide enough information about the project for the participant to be able to give informed consent. Security of the data is very important so it will be held in a secure location, whether electronic or hard copy.	

# Research Ethics Checklist

Adapted for the use by Department of Computing and  
Informatics ONLY

## 7. Human Participants Part 1 (Answer “Yes” go to 8, “No” go to 12)

Will your research project involve interaction with human participants as primary sources of data (e.g. interview, observation, original survey)?	Yes
---	-----

## 8. Human Participants Part 2 (Answer any “Yes” go to 9)

Does your research specifically involve participants who are considered vulnerable (i.e. children, those with cognitive impairment, those in unequal relationships—such as your own students, prison inmates, etc.)?	No
Does the study involve participants age 16 or over who are unable to give informed consent (i.e. people with learning disabilities)? NOTE: All research that falls under the auspices of the Mental Capacity Act 2005 must be reviewed by NHS NRES.	No
Will the study require the co-operation of a gatekeeper for initial access to the groups or individuals to be recruited? (i.e. students at school, members of self-help group, residents of Nursing home?)	No
Will it be necessary for participants to take part in your study without their knowledge and consent at the time (i.e. covert observation of people in non-public places)?	No
Will the study involve discussion of sensitive topics (i.e. sexual activity, drug use, criminal activity)?	No

## 9. Human Participants Part 2 Continued

Describe how you will deal with the ethical issues with human participants?

# Research Ethics Checklist

Adapted for the use by Department of Computing and  
Informatics ONLY

## 10. Human Participants Part 3 (Answer any “Yes” go to 11, all “No” go to 12)

Could your research induce psychological stress or anxiety, cause harm or have negative consequences for the participant or researcher (beyond the risks encountered in normal life)?	No
Will your research involve prolonged or repetitive testing?	No
Will the research involve the collection of audio materials?	Yes
Will your research involve the collection of photographic or video materials?	No
Will financial or other inducements (other than reasonable expenses and compensation for time) be offered to participants?	No

## 11. Human Participants Part 3 Continued

Please explain below why your research project involves the above mentioned criteria (be sure to explain why the sensitive criterion is essential to your project's success). Give a summary of the ethical issues and any action that will be taken to address these. Explain how you will obtain informed consent (and from whom) and how you will inform the participant(s) about the research project (i.e. participant information sheet). A sample consent form and participant information sheet can be found on the Research Ethics website.

I will need to interview potential clients of the app, such as SUBU students who normally hand out flyers in University. Interview will also be recorded, so I can review information whenever it's needed. I will be gaining consent from the individual and also will have to provide enough information about the project for the participant to be able to give his consent and show interest. All information will be held in a secure location, whether electronic or hard copy.



# Research Ethics Checklist

Adapted for the use by Department of Computing and  
Informatics ONLY

## **12. Final Review**

<p><b>Will you have access to personal data that allows you to identify individuals OR access to confidential corporate or company data (that is not covered by confidentiality terms within an agreement or by a separate confidentiality agreement)?</b></p>	No
<p><b>Will your research take place outside the UK (including any and all stages of research: collection, storage, analysis, etc.)?</b></p>	No
<p><b>Please use the below text box to highlight any other ethical concerns or risks that may arise during your research that have not been covered in this form.</b></p>	

**Review Completion Date: 29-January-2018– Double click to change it!**

***The following section is to be filled by the supervisor only***

## **Supervisor's Review:**

**Please leave your comments:**

Adapted for the use by Department of Computing and  
Informatics ONLY

## Help

### Q3 External Ethics Review

If you choose “Yes”, it will **conclude your BU Ethics Review**. Please ensure you obtain external ethical approval before commencing your research. Contact the Dorset Research Consortium with any questions regarding NRES application.

### Q4 Research Literature

Please be careful when choosing “Yes” as this means your research will only use publicly available data (e.g., research paper, market reports) or permitted private/confidential data. This also means there will be no human participants involved in your project in any form (e.g., no client, no user testing, no interview, no focus group, no survey, no field study, no observation).

### Q5 Research Literature Continued

Choosing either “Yes” or “No” will end this review BUT you will need to fill in the following field.

### Q7 Human Participants Part 1

Note: please be careful when choosing “No” as this means there will be no human participants involved in your project in any form (e.g., no client, no user testing, no interview, no focus group, no survey, no field study, no observation).

#### Above minimal risk

The “Above minimal risk” will be automatically identified if you answered yes to one or more questions below. In that case, you **must be careful**. Consult your supervisor or project tutor if you need help. Alternatively, you may contact Sarah Bell, RKEO Research Governance and Ethics Adviser at [sarah.bell@bournemouth.ac.uk](mailto:sarah.bell@bournemouth.ac.uk).

- Does your research specifically involve participants who are considered vulnerable?
- Does the study involve participants age 16 or over who are unable to give informed consent?
- Will the study require the co-operation of a gatekeeper?
- Will it be necessary for participants to take part in your study without their knowledge and consent?
- Will the study involve discussion of sensitive topics?
- Could your research induce psychological stress or anxiety, cause harm or negative consequences for the participant?
- Will your research involve prolonged or repetitive testing?
- Will the research involve the collection of audio materials?
- Will the research involve the collection of photographic or video materials?
- Will financial or other inducements be offered to participants?
- Will your research take place outside the UK?

## J Disk Content

- Copy of thesis report in pdf
- Database Backup
- Android Project Files & Directory