School of Electronics and Computer Science

INFO6005 - Application Development Report

By Adam Costello, Michael Elkins, Jonathan Harrison, Samantha Kanza, Pratik Patel & Craig Saunders

Abstract

1 Introduction

OptiCaff is the new innovative way to maximise productivity during university hours. Opticaff can calculate the optimum times to consume caffeine by analysing a users caffeine content and their timetable for the day. This is however not all it can do; not only will it notify a user when its time for them to consume their next caffeinated beverage, it will suggest what type of drink would be most effective (tea, coffee, energy drinks). It will then inform the user of the nearest place in the University of Southampton that sells that product. Opticaff also features a competative element where users can compete with each other to appear on the leaderboard listing the most productive users.

1.1 Project Problem

Modern life is both busy and stressful and consuming caffeine has become a popular way of managing this for many people [14]. Despite its usefulness, the fact remains that caffeine is still a drug and can have adverse effects if misused [16]. Using caffeine effectively is a difficult skill to master as it would require logging the caffeine content of each beverage in addition to consumption time and then performing complex calculations to ascertain the optimum time for the next dosage. Unfortunately even performing this complex process would not necessarily be enough to manage caffeine levels effectively as they would not take into account daily tasks where obtaining caffeine was infeasible (e.g during a lecture). Opticaff aims to solve this problem by monitoring a users caffeine intake in addition to their timetabled activities so that consumption suggestions work in harmony with the users routine. In addition to this it also points the user to the closest place to purchase their desired beverage to minimise their detour.

1.2 Project Audience

The initial target audience of Opticaff are people who visit the University of Southampton Campus on a regular basis. The shops and vending machines that are suggested by Opticaff are all either University of Southampton affiliated or in close proximity to the campus and are therefore regularly accessible to frequent visitors of the University.

1.3 Project Goals

The goals of this project are to produce a prototype with the following functionality:

1. Obtain and use linked data detailing caffeine sources within the University of Southampton.
2. Integrate and use University of Southampton timetable data using Sussed
3. Monitor users caffeine intake and predict the optimum time for the next consumption.
4. Have the notion of a leaderboard to rank users “productvity” based on how well they maintain their caffeine level within the optimum range.
5. Approximate the users position and locate the closest appropriate caffeine sources to them.

1.4 Project Scope

This is a prototype application to showcase the main features and to illustrate what the full application will look like. The prototype produced will implement the key features of consuming caffeine, calculations based on a users timetable, directions to the nearest appropriate caffeine source, and ranking on the leaderboard. This application will use researched averages for the caffeine content of drinks as opposed to storing individual data for each one and will make assumptions for the average man and woman in terms of optimum caffeine intake. In addition the application will be able to add a users calendar data for the aforementioned calculations.

2 Project Conceptualisation & Monetisation

This section details an outline of the ideas that were conceived and the justifications for the final idea coupled with its monetisation potential.

2.1 Ideas

It was decided from the early stages that the application would use the University of Southamptons open linked data (see section **??** relating to its internal organisations. The focus of this data would be the points of service (e.g cafes, vending machines) that sold caffeine in and around the University. After establishing the data sets a number of ideas of how to best uitlise this data were brought to the forefront. In order to establish the most valuable idea, each potential solution was weighed against certain aspects:

- **Uniqueness Factor:** Was it a new novel idea and if so how?
- **Monetisation Potential:** Was there the potential to make money out of it?
- **User Interest:** Would it capture users on a long term basis as opposed to just on a novelty?

2.1.1 Caffeine Finder

An initial idea was a caffeine finder application that allowed the user to find the nearest, available, appropriate (e.g Student or Staff) caffeine to them at any given time.

- **Uniqueness Factor:** As stated in the competitors research section (section 3.1.5 there are apps similar to this in the market currently. It still has a slight unique factor in that it includes all of the University affiliated points of service including vending machines as well as nearby external stores. In addition it also covers a range of caffeinated products as opposed to focusing on tea and coffee. Nonetheless there are still similar applications out that perform similar functions.
- **Monetisation Potential:** There would be the potential to ask specific caffeine selling stores to invest in return for putting them at the top of the applications recommendations.
- **User Interest:** This idea doesn't have a great potential for capturing users interest on a long term basis. It has a novelty factor of showing the users where and when they can purchase caffeine, but runs the risk of loosing interest after the users have used it enough to retain any useful information.

2.1.2 Justification for Rejection

Overall this application was considered a decent idea. It wasn't greatly unique but still appealed to both a wide and specialised market in that it would be of use to everyone who frequented the University and it covered all the points of service in a nearby range. However, its main downfall was its lack of ability to maintain a user interest, as without that potential it would be infeasible to expect external parties to invest in it.

2.1.3 Caffeine Notifications

Another idea was to make a very simple application that essentially buzzed and notified the user whenever they walked past somewhere that sold caffeine.

- **Uniqueness Factor:** This was debatably more unique than a finder application as it would specifically alert the user to the presence of caffeine as opposed to waiting for them to search for it.

- **Monetisation Potential:** This idea didn't have a great deal of monetisation potential. If it's going to buzz whenever it passes a caffeine location then there was no feasible way to promote specific locations over others. The only way this could have been monetised would have been to charge for the app. However despite it's potential usefulness it still seemed unlikely that users would actually pay for the service when realistically they could just pay attention and get to know the locations and achieve the same result.

- **User Interest:** This application could easily hold user interest as its a very simple concept and doesn't require any effort on their part apart from keeping the app running in the background.

2.1.4 Justification for Rejection

This application had benefits in its simplicity but with its lack of monetisation potential it was unsuitable for this project.

2.1.5 Caffeine Productivity with a Competative Edge - Opticaff

The final idea expanded on our inital idea of a caffeine finder application but with an additional function to tie the user interest in. Opticaff is an application that allows users to input their daily timetable and based on user input of caffeine consumption and personal details, calculates when they need to consume their next caffeinated beverage. It uses your calendar for the day to determine if you have any events left (e.g a lecture) and if so it calculates when you should consume caffeine and what strength it should be to maintain optimum caffeine levels throughout. However, this is not the only functionality of the application. It also has the concept of competing with your peers to maintain the optimum caffeine levels with a leaderboard to showcase the most productive users.

- **Uniqueness Factor:** This application had a unique factor in that it had more functionality than any of the competitors mentioned in section 3.1.5. There are applications in the market that monitor caffeine consumption, applications that locate caffeine, and competative applications. This application combines all three for a truly unique product.

- **Monetisation Potential:** This idea retained the potential to be promoted to owners of the caffeine points of service for investment. It could also potentially be pitched to be sponsered in a research function to investigate the effects of monitoring caffeine intake on productivity.

- **User Interest:** This idea has by far the best user interest potential out of the three. It's not just a novelty application as people consume caffeine every day, and based on their different calendars (which for uni students and lecturers will differ by day) and external factors such as amount of sleep the night before their caffeine consumption will differ each day.

2.1.6 Justification for Acceptance

Opticaff was chosen as the final idea as it fulfilled all three of the criterion listed above, and fulfilled all of them in a superior fashion. It has the unique factor not through its individual ideas but through the combination of these into a multifunctional app that not only allows you to monitor your caffeine consumption, but to locate places that sell caffeine and tie the two together to optimise your caffeine intake. In addition to this it has the competative element which aids in capturing and retaining user interest as well as boosting potential monetisation potential.

3 Background Research & Analysis

3.1 Market Research & Analysis

This section details the market research analysed for Opticaff. The Coffee market was researched as were the different mobile development platforms; which were analysed to determine the most suitable one to use for this prototype application. Finally the potential competitors to Opticaff were detailed and analysed to see if any of them pose a genuine threat against it.

3.1.1 Coffee Research & Analysis

Coffee consumption in the United Kingdom has steadily increased over the past decade. In particular the past five years been seen an explosive increase, there are several theories as to why this is the case. Firstly, instant coffee shops have become more common on our high streets. Companies such as Starbucks™and Costa™have been opening more stores as more citizens have been buying instant coffee; this doesn't show any signs of slowing down either as Starbucks have recently announced 300 new stores to be opened over the next five years [20].

Secondly, these brands have contributed to the newfound 'coolness' that is associated with coffee [23]. Lastly, there is evidence that the economic climate has played a part in coffee's rise. Also known as the 'lipstick effect', Britons have been unable to afford expensive treats for themselves so they have been spending on cheaper treats, a good example of which is coffee [21]. This research is important to OptiCaff because it shows that the coffee industry is on the rise, and it would not make business sense for us to invest in a declining industry.

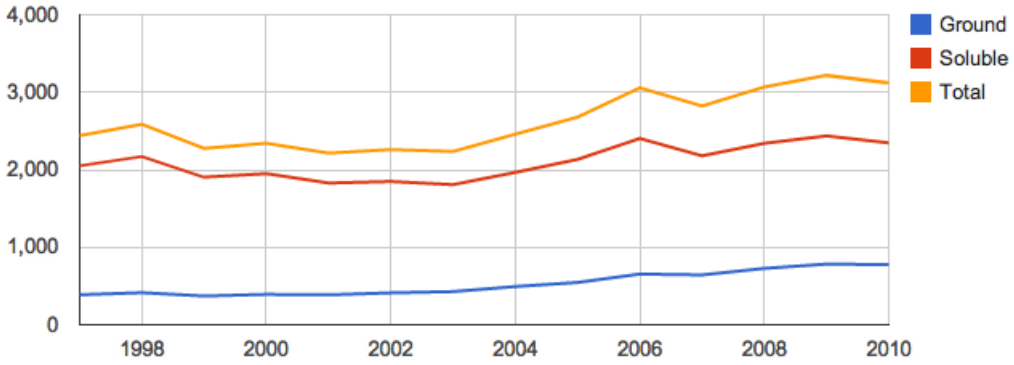

Figure 1: Coffee Statistics 2010 [**?**]

3.1.2 Mobile Platform Research

OptiCaff 's purpose lent itself heavily to being a mobile application, and as such that sparked the debate of what platform it should be developed for. There are various mobile operating systems that Opticaff could be deployed on [**?**]:

- Google's Android
- Apple's iOS
- Blackberry's RIM
- Windows Mobile

Based on the Smartphone Operating System statistics from June 2011 Android and Apple are dominating the market at present. Therefore these were the two platforms that were considered.

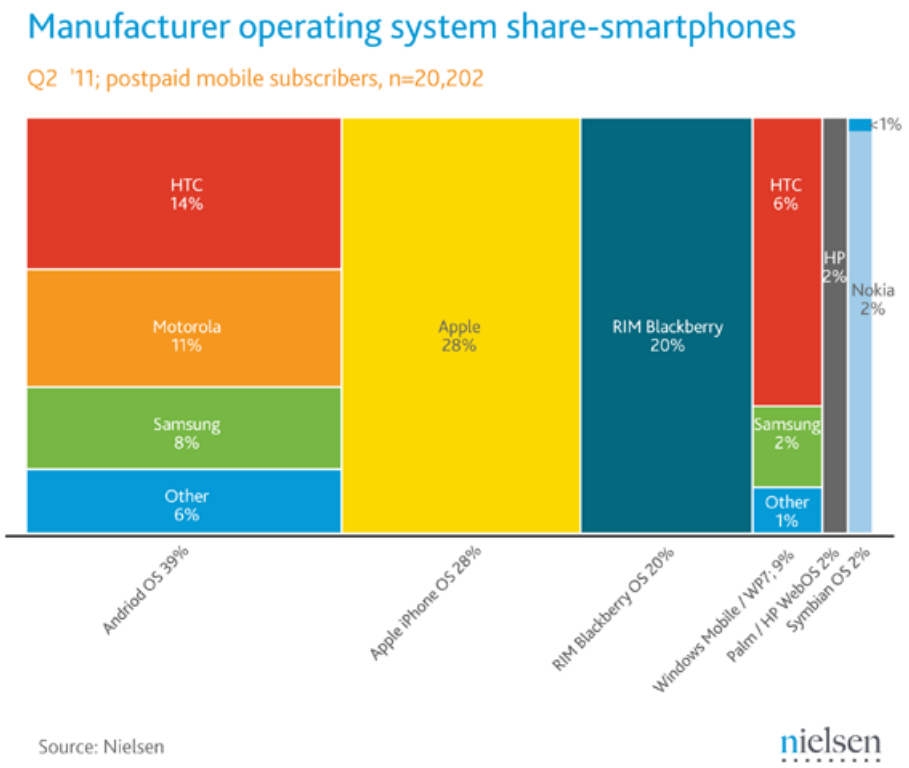

Figure 2: Manafacture Operating system share-smartphones June 2011 [9]

The following elements were considered in regards to which platform to use: language, development requirements and popularity of the apps for that operating system.

Programming Language

- iOS requires applications to be written in Objective C [**?**].
- The Android Operating System requires applications to be written in Java [5].

Development Requirements

- The only IDE available for developing iPhone applications is Xcode and there is a requirement to have Apple's iOS SDK installed [**?**]. Both of these are only available to OS X users and therefore Apple Computers.
- Android doesn't require any special hardware to develop an application. The Android SDK is freely available and it's recommended development enviroment Eclipse [17] is also free.

Application Popularity

- Apple's App Store is currently the more popular at 25 billion downloads [**?**]
- Google Play (Android's market place) was reported to have reached 10 billion downloads in December 2011 [**?**]

It was decided that for the prototype application, Android would be the simpler option for the following reasons:

- Android uses Java. All members of the team are experienced in Java development one member has experience in Android Development.

- Android can be developed on any platform which is useful as the group uses a combination of OSX, Linux and Windows.

- Google Play may be less popular than Apple's app store, however it still holds a large user base and it was decided that ease of development was a higher priority for a prototype application.

3.1.3 Gamification Research & Analysis

A common issue for new apps is user retention; a method of increasing this is Gamification [8]. Gamification is the practise of adding game-like elements to something that is not already a game, e.g. a to-do list [7]. There are ways in which gamification can be applied to OptiCaff.

1. Use of achievements or awards. Achievements are used to recognise when a player has fulfilled certain conditions when playing a game. An example of how this could be used with OptiCaff would be giving an achievement to a user that has stayed in the optimum caffeine range for 3 hours.

2. Use of leaderboards. A leaderboard would show the users that are the best at using OptiCaff in a specified way, an example of this would be show the users that stay in the optimum range for the longest time.

Based on this research, it was decided that OptiCaff would use leaderboards as a method of gamification, provided that it is done in a safe way. This is because leaderboards can link all of the users of OptiCaff turning it into a multiplayer game. Obviously as with any game there would be the potential to cheat by pressing the button without actually consuming the caffeine, but this still doesn't detract from the fun element. One of the dangers of gamification is extreme behaviour, OptiCaff will not reward behaviour that is potentially dangerous, e.g. rewarding the user that has the highest caffeine intake.

3.1.4 Monetisation Research & Analysis

It is more challenging to make money from Android Applications as opposed to Apple iOS Applications according to a report from Distimo, an app store analysis company [15]. The report suggests several methods to maximise the money earned, these assume that the app is high enough quality to be sold.

1. 80% of paid applications have been bought less than 100 times.

2. In-app advertising can either make a lot of money or very little, it depends on the amount of users an app has.

3. Finally, it is common on app stores/marketplaces to have a paid version of an app that is also free. There is no difference in the function of the app, though a paid version would not display advertisements. This would give the user the choice to use the OptiCaff free or paid version though there would be income for the team either way.

Based upon these three points it has been decided that OptiCaff would be developed as a free and paid version that contains advertising in the free version. The next logical step after gaining a significant user base would be to pitch the application to the owners of the caffeine vendors for funding in exchange for favouring their points of service within the application.

3.1.5 Competitors Research & Analysis

This section details both the direct and indirect competitors to Opticaff and ascertains if any of them pose a substantial threat.

	Caffeine Finder	Caffeine Zone	Caffeine Data	Opticaff
Does this app allow you to locate caffeine sources?	✔	✘	✔	✔
Does this app direct you to caffeine sources?	✔	✘	✘	✔
Does this app help you to manage caffeine content?	✘	✔	✘	✔
Does this app help you to manage caffeine content in relation to your day's activities?	✘	✘	✘	✔

Table 1: Table of Opticaff's Competitors

Caffeine Finder is a BlackBerry application that offers some of the same functionality that OptiCaff will provide; this makes Caffeine Finder a direct competitor of OptiCaff. Caffeine Finder directs a user to the nearest restaurant or café, give the address and even display reviews of the destination if available. There are however several negative points regarding this application:

- The chosen platform was the BlackBerry which has a small screen compared to Android phones and iPhones.
- The application doesn't inform the user when the optimum time to have a coffee or other caffeinated drink is.
- A user may already be tired before they think to check Caffeine Finder which is something OptiCaff will try and prevent.
- The application was released in 2005 and has not been updated regularly since that time, this is shown by reports that it is not fully compatible with newer operating systems.

Caffeine Zone 2 Lite is a free iPhone application that tracks the amount of caffeine in the body, OptiCaff will also have caffeine tracking ability and alerts. This makes Caffeine Zone 2 Lite a direct competitor, although OptiCaff offers a superior service for the following reasons:

- OptiCaff offers a complete solution, Caffeine Zone 2 Lite only tells the user when they should have caffeine, it makes no effort to tell the user where they can get a caffeinated drink.
- The alerts generated do not consider the user's schedule, OptiCaff will look at the user's calendar to see if they require an earlier warning for caffeine so as to accomodate their schedule.
- Caffeine Zone 2 Lite is focussed on being an educational tool about caffeine use this is in contrast to OptiCaff which will prioritise providing a service.

One of the issues of using open data is that the data itself can be considered an indirect competitor of OptiCaff. It could be possible for another product to be created that uses the same data set, this means that OptiCaff could have more potential competitors than it would if it used closed data. OptiCaff could not be replicated by a competitor only using the same open data however, this is because there will be caffeine level prediction and notification features implemented.

The table below summarises the competative edge of Opticaff, illustrating how it combines the key features of its competitors into a superior all encompassing service:

3.2 Application Research & Analysis

This section details the research and analysis for the data within Opticaff. The calendar data that will be used for the application is detailed, as are the caffeine calculations that will be used in the prototype.

3.2.1 Calendar Research & Analysis

OptiCaff's objectives include the use of university timetables to schedule caffeine level notifications. Sussed is Southampton university's student portal that displays a student's timetable, OptiCaff will need to access the timetable through Sussed. This is currently the only way to obtain a student's timetable given that its not freely available as open data. Another Southampton university produced application, iSoton, has been able to do this process showing that it is possible. Further investigation however has shown that this is not a trivial process.

An alternative solution is to use Google Calendar which can be easily integrated into Android given that they are both Google Products [1]. This has the disadvantage however that it requires the user to either have or set up a Google Calendar detailing their timetable information.

Overall given the prototype nature of this application and ability to use Google Calendar in a much simpler fashion it was decided that Opticaff would use Google Calendar.

subsubsectionCaffeine Research & Analysis In order to provide the caffeine management element of this application, the different levels of caffeine that appeared in beverages and its affect on human beings needed to be researched. This section details the caffeine levels and decay rate that has been used in Opticaff.

Caffeine Levels in Products
Given the vast range of different cafffeinated products and the limited time to produce a prototype application, it was decided that the products displayed by OptiCaff would be grouped into four different types of drink, and each type would be allocated an average caffeine content. Below is a table showing these totals, which were obtained these sources [18] [19] [22].

Drink Category	**Average Caffeine Content (mg)**
Tea	40
Coffee	54
Energy Drinks	80
Soft Drinks	34.5

Caffeine Decay Levels
In addition to calculating the level of caffeine obtained from a specific product, it was also important to work out the optimum caffeine levels and how long it would take the caffeine to "decay" within the body so that the next dosage time could be predicted. For the purposes of the prototype, Opticaff uses the same optimum caffeine levels as it's competitor Caffeine Zone 2 (detailed in Section 3.1.5) uses which are between 200 and 400mg [26]. The half life of caffeine ranges between 2.5 and 4.5 hours [27] [25] and to simplify matters 4 was chosen as the number to use in Opticaff and was calculated using the half life formula detailed here: [28].

4 Design & User Interface

This section details the design considerations that were made prior to constructing the user interface; it demonstrates the four main components of the user interface and the issues encountered during creation.

4.1 Design Considerations

When implementing the user interface, we wanted to ensure that the design of the application was consistent with the Android 4.0 design ideology. To achieve this, we referenced the Android design and style guide which outlines some of the key aspects of an Android 4.0 application, and some of Google's design decisions made throughout the OS, which should be implemented and considered within our application. This included decisions such as universal behaviour of the back button, how preferences should be handled within the application using a separate fragment or activity, and interaction and navigation of the different subactivities of our application.

Later versions of Android make use of an Actionbar which is implemented across all Android 4.0 applications as a way of providing a consistent method for navigation. This has therefore been included within the application with a main logo and icons which can be used to navigate between the different viewpoints.

For the implementation of the UI, it was designed so there were different activities for the different viewpoints available. There was one main activity for the home screen, and then activities for the map, graph, leaderboard, and the settings were implemented using the newer preference fragment class, which would match the preferences theme of the system. We implemented the "Up" navigation option only on settings, as it is the only part of the application that is not a top level activity. Therefore all of the other activities make use of the back navigation instead, due to the lack of a hierarchy between the different viewpoints. Tapping on the logo in the top left corner will always take you back to the default home view.

There are certain key design principles that have been outlined by Google in their style guide, that we have tried to implement within the application. Making sure that the important information is easily accessible, and the most important options and features can be accessed from the main home screen, providing an overview of events and the next time they should take caffeine. In the app structure they stress how the main home view should not just be a navigation menu, but instead stating the importance of "making content the centerpiece of your start screen", which this satisfies.

4.2 Final Design

4.2.1 Action Bar

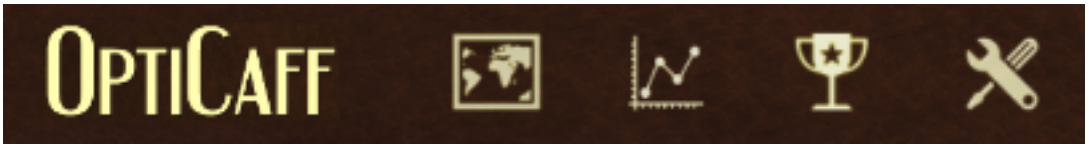

Figure 3: Action Bar

The action bar has used well known and explanatory symbols to illustrate each activity that the buttons relate to (e.g a map of the world for maps, and the well known settings wrench symbol). The colours used are contrasting so that the symbols show up clearly for the user.

4.2.2 Activities

Below is a figure detailing the main screens of each activity section in Opticaff.

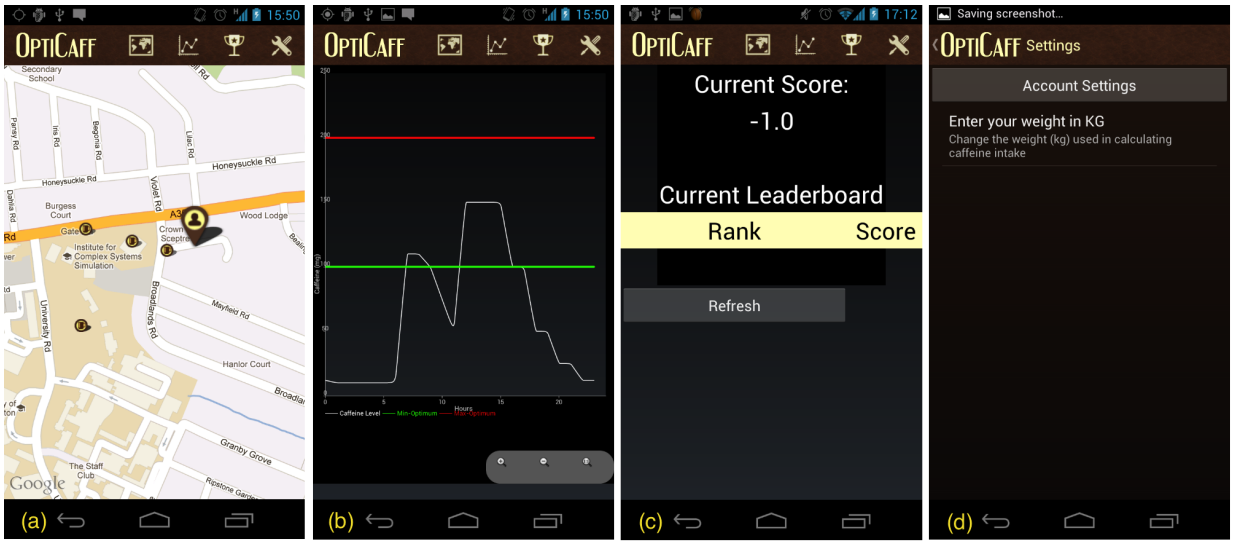

Figure 4: Activities: Maps, Graphs, Leaderboard & Settings

(a) Map Screen
This screen uses the well known pointers utilised in popular map applications such as google maps to illustrate the users location. The nearest caffeine vendors are shown in small brown circles around the map to give the user a visual idea of their proximity.

(b) Graph Screen
This screen shows the tracking of a users caffeine levels. This enables the user to easily see if they are staying within optimal levels. The colours red and green are used for dangerous and optimal levels respectively. This is an intuative design choice as red is often used to portray danger or stop signals wheras green is used for positive symbols such as in traffic lights.

(c) Leaderboard Screen
This screen allows users to see their current score and to compare it with those of their friends. Putting the users score at the top in large text allows them to immediately assess whether they are achieveing a good score through the use of positive and negative symbols.

(d) Settings Screen
This screen allows users to change their settings. It is currently very simplistic as the prototype has minimal settings. However it provides a basic explanation for the setting to aid the user in why they would edit it.

4.3 Issues with User Interface

When developing the UI, we were initially implementing the different activities in fragments, with one main container activity managing all of them. Unfortunately this caused issues when we implemented the map, as due to some incompatibilities between the MapActivity class in android and the new Fragment classes, there were problems when trying to contain what was viewed as an activity, within a fragment. This meant that we implemented the different parts of the application as different activities.

5 Implementation

5.1 Overview

Our system has been implemented using GAE (Google App Engine) and Android to construct an App Engine Connected Android Application. The justification for using Android is detailed in Section **??**. GAE was used for the backend of our system for two reasons, firstly because members of the team had prior experience in using this system. Secondly because support for this project is provided by Eclipse; which generates a starter project with the following implementations [10]:

- Handling communication between the service and application
- Registering and unregistering with the service
- Authentication and user accounts
- Transferring of data between the application and datastore to allow use of the same classes.

5.2 Architecture

The overall architecture of such an application is given by the diagram below:

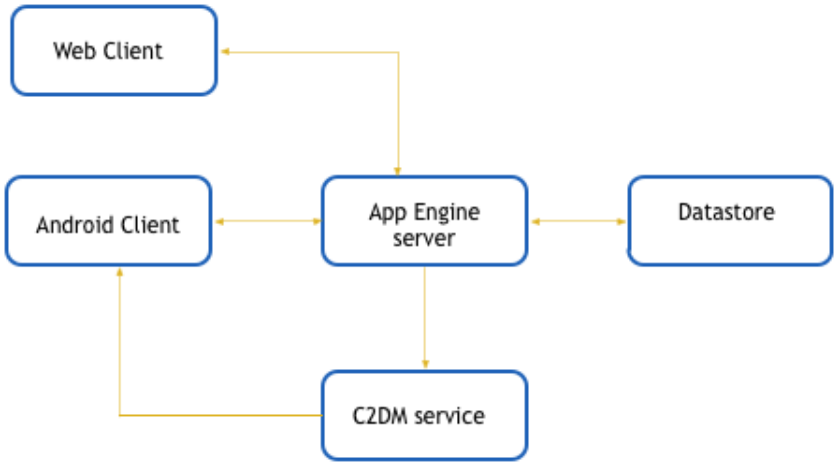

Figure 5: Architecture Diagram [10]

From this architecture the following modules have been implemented:

- The **Android** application.
- The **GAE** including the **Datastore**.
- A basic implementation of the **Web Client** [13] (which could be expanded as part of the future work, see Section **??**.

The C2DM (Cloud to Device Messaging) service which facilitates sending messages to Android devices from the cloud system is an additional service provided by Google [13].

5.3 Google App Engine

The GAE backend was implemented in Java and is split into a number of components: the Datastore and Remote Procedure Call (RPC) service

5.3.1 Datastore

The first component is the database, which is implemented using GAE's implementation of Java Data Objects (JDO). The objects that have been created to store information in the system are detailed in the following table:

Name	Purpose	Example
CaffeineSource	To model a location where caffeine products are sold.	Avenue Cafe.
CaffeineProduct	To model a unique caffeine product containing information such as name, type and milligrams of caffeine in product.	Red Bull Can
CaffeineSourceProduct	To model a unique caffeine product being sold at a location extending the referenced CaffeineProduct to include pricing information at this location.	Red Bull Can at Avenue Cafe.
OpeningTime	To model a opening time for a location.	Monday's opening hours are between 8:00 and 19:00 for the period 00:00:00 16/4/12 to 23:59:59 15/6/12.
LeaderboardScore	To model a user's score for leaderboard purposes.	

Table 2: Table detailing System Objects

Additionally other objects may exist in the database such as DeviceInfo and C2DMConfig which are automatically added by the system to allow Cloud to Device Messaging (C2DM) which hasn't been created by the team.

All of the location information used in Opticaff is retrieved using SPARQL queries on the University of Southampton's SPARQL endpoint [12]. The datastore is evaluated daily to ascertain if it needs re-populating. This is done through the use of a scheduled task (cron job) and HTTPServlet which checks if the information currently stored is valid (e.g location opening times haven't expired) and if so performs the necessary dropping and adding of information to ensure the data is kept up-to-date.

Create, Read, Update and Delete (CRUD) operations have also been created to handle the tasks involved in the cron job service as well as the RPC service which is detailed in the next section.

The University of Southampton's data is stored is so that the application doesn't have to perform SPARQL queries on the fly and process them on every user request (e.g nearest caffeine source locations). Additionally the use of a JDO datastore is because it allows both the Android application and datastore to use the same Java objects and provides the means to transfer and use data easily.

5.3.2 RPC Service

The RPC service acts as an interface to the datastore for the Android Application and Web Client; communicating with this service is done using Google Accounts which handles the authentication process and user accounts meaning that to use our system it's necessary to have a Google account furthermore it means we don't have to store user account information in our datastore.

This service contains a number of RPC methods which perform query actions on the datastore to provide Datastore objects information such as top 5 players on our leaderboards and nearest caffeine source locations and information for those sources.

5.4 Support and Libraries Used

Libraries and sources of information which were used to construct this project are listed below: Google App Engine Documentation Android Documentation Jena Library To perform SPARQL queries and process results in GAE. (http://incubator.apache.org/jena/index.html) How to read calendar on phone in Android. http://stackoverflow.com/questions/7859005/how-to-read-and-edit-android-calendar-events-using-the-new-android-4-0-ice-cream How to create alarm events in Android. http://justcallmebrian.com/?p=129 How to create Progress Dialog in Android. http://www.helloandroid.com/tutorials/using-threads-and-progressdialog

5.5 Implementation Problems

During development a few problems were encountered:

- When running the Local App Engine server, an issue occurs when using a new version of java due to the removal of a getDefaultTimezone method. A solution was given in [3] and this was followed successfully.
- When testing the GAE Connected Android Project; the application was unable to obtain the debug url for the Local GAE. A solution was found here [2] and followed.

6 Project Management & Tools

6.1 Project Management

This section details how this project was managed, the roles of the different group members and the methodologies that were used.

6.1.1 Team Roles

The following roles were outlined for this project:

- **O**rganiser - Oversees project and time management
- **D**eveloper - Develops the application
- **U**sability Expert - Manages User Interface and HCI (Human Computer Interaction)
- **R**esearcher - Researches background information
- **P**resenter - Presents pitch & manages presentation

This table details the roles each member of the team undertook:

Team Member	Roles	Description
Adam Costello	**R, P**	Adam coordinated with Craig on the Presentation section of this project, and worked with Sami on the research section focusing on the market research.
Mike Elkins	**U, D**	Mike coordinated with Pratik to design and build the User Interface section of the application.
Jonathan Harrison	**O, D**	Jonathan was in charge of organising the team and making sure tasks were completed in a timely manner. He also worked with Sami on the backend element of the application concentrating on the calendar and scheduling the system to import the necessary data.
Sami Kanza	**R, D**	Sami coordinated with Jonathan on the backend of the application, concentrating on the queries to retrieve the data, she also worked with Adam on the research section, focusing on the caffeine research.
Pratik Patel	**U, D**	Pratik worked with Mike to design and build the User Interface section of the application.
Craig Saunders	**P, D**	Craig worked on the location and directing element of the application in addition to working with Adam on the Presentation section.

6.1.2 Team Organisation

The team decided that bi-weekly meetings would be appropriate for this project. In each meeting the progress made between then and the previous meeting was stated, any problems that needed solving were raised and then the tasks for the next meeting were divided up. The nature of these meetings were similar to a SCRUM [24] daily standup, but were obviously performed less frequently than that.

6.2 Group Methodologies

The team decided to implement several agile-based technologies to aid with the application development process. The development areas have been broken down into individual tasks (stories)

to simplify the process. An iterative development process was also adopted, with the base of the application built first, followed by a gradual development of its features.

Pair Programming was also an agile technique that the team used. The application development was split into three areas: Database development and SPARQL querying, the map interface and GPS positioning / directions, and the overall user interface. Each section had two group members assigned to it, and each pair worked together to share their skillset and therefore produce a superior result.

6.3 Tools & Techniques

This section details the tools used by the team to aid with their project management and development.

6.3.1 Version Control

There were several version control mechanisms we could have used. Git and SVN were both considered, with potential to store the code on UGForge or Googlecode or Github etc. In the end the group chose to use SVN Googlecode. SVN was chosen primarily due to the fact that Eclipse was being used as the main development environment, and there is an SVN plugin for Eclipse that makes the subversioning process easier to integrate with the development process. Googlecode was chosen because the application that is being developed is for Android, and therefore it was felt that Android's creators Google would be the most sensible place to store the application.

6.3.2 Data Source

OptiCaff used the Open Data Service from the University of Southampton [4] to retrieve the relevant information for the application. This service provides open linked data about some of the administrative information regarding the university. It also provides a SPARQL Endpoint [12] (a service which facilitates users querying a knowledge base using the SPARQL query language) [11]. OptiCaff utilises this with a few specialist queries, and combined with user preferences can provide the user with a wide selection of caffeine choices around campus.

6.3.3 Development Tools

Given that OptiCaff was to be an Android application. The development tools needed to facilitate its production were the Android SDK tools [5]. The recommended development environment suggested by Android was to use Eclipse [17] with the ADT (Android Developer Tools) plugin [6]. Given the groups overall familiarity with Eclipse and it's additional useful plugins for version control (see section 6.3.1) this was used as our IDE of choice.

6.3.4 Communication Tools

The Team used Google Docs to share documentation and task lists, and Facebook to communicate via group chat.

7 Future Work

This application has a great potential for future work. This section details some of the ways in which the application could be extended/improved:

7.1 Better Calendar Integration

This system currently uses Google Calendar, in the future it would be useful to extend this to use different calendar such as university timetables so that certain events such as lectures don't need to be input manually to Google Calendar.

7.2 Adding Favourite Locations

Adding the notion of favourite products and locations would enable the user to customise the caffeine suggestions that they recieve from the application. For instance if the user only liked tea and coffee but not energy drinks then that would be taken into account for the predictions and they might be advised to consume caffeine more regularly as they favour drinks with a lesser caffeine content.

7.3 Accurate Caffeine Levels Per Product

For the purposes of the prototype Opticaff only used average values for the four categories of caffeinated beverages: (coffee, tea, soft drinks and energy drinks) and assumed an average size for each beverage. For a final application each listed product would have it's appropriate caffeine content listed.

7.4 Advanced Leaderboard Functionality

Currently within the prototype there is a leaderboard for the optimum caffeine levels. In the finished application it would add a better competative edge to add additional functionality such as a history of scores, mapping caffeine levels to a specific event (e.g I was at an optimum caffeine level during this lecture). It would also encourage competativeness if there was the notion of adding friends so that users could compete directly against their peers. There could also be the notion of groups such as course groups or module groups to see who maintained the optimum caffeine levels within a specific group.

7.5 Web Interface

Having a web interface that users could also use to update their information or to view leaderboard statistics could increase the amount of people using Opticaff as there would be multiple ways they could use it.

7.6 Dynamic Stock Reflection

If caffeine points of service were to start to release (either publicly or privately just to Opticaff) stock levels of products dynamically then Opticaff could take this information into account and only direct users to places that had the desired beverage in stock.

7.7 Health Warnings

Despite its useful traits, Caffeine is still a drug and like all drugs needs to be managed carefully. Opticaff aims to aid the users in managing their caffeine content and keeping it at an optimum level. However, it would be worth adding health warnings about the risks of overdosing on caffeine and to make it clear that Opticaff promotes responsible useage both for health and legal reasons.

7.8 Adaptation to other Universities

Given that this system uses Google Calendar to verify the users daily activity, and that the system has been built to import data from a set of SPARQL about the caffeine locations; adapting this application for multiple universities wouldn't take very long. The users would continue to add their timetable data in as before, and a new set of queries would be built for that university. This would enable Opticaff to be pitched to various university establishments for minimum additional development time.

7.9 Adaption to Specific Coffee Chains

Based on similar principles as the idea above, this application could be adapted to a specific Coffee Chain such as Costa or Starbucks if they gave Opticaff access to their location and product data. This would then enable the app to be used by anyone who was a fan of caffeine or indeed these specific stores as large chains such as these have branches all over the country.

8 Evaluation

8.1 Product Evaluation

In regards to achieveing the projects aims, four out of the five goals set out in the initial planning stages were completely met, and one of them was partially met. The table belows details this:

No	Requirement	Met	Description
1	Obtain data regarding caffeine sources in and around the University of Southampton.	Yes	All points of service that are listed as providing caffeine are used within our system.
2	Store University Timetable data using Sussed	Partially	As discussed earlier in section 3.2.1 accessing the timetable data from Sussed wasn't a simple task so Google Calendar was used instead. This does mean that the users have to input their own timetable data, however it does meet the requirement of using calendar data.
3	Make calculations about user caffeine intake and make predictions for the optimum time to consume next.	Yes	Opticaff works out the caffeine decay rate (based on the calculations specified in section **??**) and based on the users timetabled events makes a prediction for the best time to consume the next caffeinated beverage.
4	Have the notion of a leaderboard to rank users "productvity" based on maintaining their optimum caffeine levels.	Yes	A basic leaderboard has been implemented that ranks users based on their caffeine consumption.
5	Ascertain the users position and locate the closest appropriate caffeine sources to them.	Yes	The long/lat values of all the caffeine points of service are stored by Opticaff. By using GPS the users location can be determined also and then they can be directed to the appropriate destination.

This shows the the project was very successfull in meeting its aims and the group was happy with the final product and the features it implemented.

8.2 Team Evaluation

The group feel that they worked well together as a team. Everyone participated and fulfilled the two roles they were assigned at the beginning of the project. The Bi-weekly meetings were kept to with additional sessions when necessary. The project met its requirements and was delivered on time for the deadline.

References

[1] http://android-developers.blogspot.co.uk/2011/10/ics-and-non-public-apis.html.

[2] http://code.google.com/p/google-plugin-for-eclipse/issues/detail?id=46.

[3] http://code.google.com/p/googleappengine/issues/detail?id=6928.

[4] http://data.southampton.ac.uk/.

[5] http://developer.android.com/sdk/index.html.

[6] http://developer.android.com/sdk/installing.html.

[7] http://edition.cnn.com/2010/tech/web/09/30/web.checkin.onetruefan/.

[8] http://enterprise-gamification.com/index.php/facts.

[9] http://mashable.com/2011/07/28/nielsen-mobile-stats-june-11/.

[10] https://developers.google.com/eclipse/docs/appengine_connected_android.

[11] http://semanticweb.org/wiki/sparql_endpoint.

[12] http://sparql.data.southampton.ac.uk/.

[13] https://rich2012cafe.appspot.com/.

[14] http://www.directlyfitness.com/2011/ultimate-caffeine-guide/.

[15] http://www.distimo.com/blog/2011_05_in-depth-view-on-download-volumes-in-the-google-android-market.

[16] http://www.drugfree.org/drug-guide/caffeine.

[17] http://www.eclipse.org/.

[18] http://www.energyfiend.com/caffeine-content/coca-cola-classic.

[19] http://www.food.gov.uk/science/surveillance/fsis2004branch/fsis5304.

[20] http://www.guardian.co.uk/business/2011/dec/01/starbucks-coffee-business-jobs.

[21] http://www.guardian.co.uk/business/2011/oct/18/costa-coffee-whitbread-premier-inns.

[22] http://www.livestrong.com/article/296806-energy-drink-nutritional-values/.

[23] http://www.passionforcoffee.com/background/overview/.

[24] http://www.scrum.org/.

[25] Actions of Caffeine in the Brain with Special Reference to Factors That Contribute to Its Widespread Use. *Pharmacological Reviews*, 51(1):83–133, March 1 1999.

[26] Best time for a coffee break? there's an app for that. *Penn State*, February 2012.

[27] J Arnaud. M. The Pharmacology of Caffeine. *Prog Drug Res*, 31:273–313, 1987.

[28] Wikipedia. Half-life — wikipedia, the free encyclopedia, 2012. [Online; accessed 23-April-2012].