Project Definition

ISYS20181: Practical project management

Group Eii:

Ashley Thompson (N0569351)

Mateusz Januszewski (N0504690)

Nirmoldeep Bajwa (N0564293)

Charalambos Foti (N0387309)

Table of Contents

[Project Introduction 3](#_Toc440277339)

[Aim 4](#_Toc440277340)

[Objectives 4](#_Toc440277341)

[Functional Requirements 5](#_Toc440277342)

[Project Management 6](#_Toc440277343)

[Team Responsibilities 7](#_Toc440277344)

[Team Responsibilities 8](#_Toc440277345)

[Sources of Information and Resources 9](#_Toc440277346)

[Risk Assessment 10](#_Toc440277347)

[Professional, Legal, Social and Ethical issues 11](#_Toc440277348)

[Project Plan, Milestones and Timescale 12](#_Toc440277349)

[References 12](#_Toc440277350)

[Appendix A 13](#_Toc440277351)

[Appendix B 16](#_Toc440277352)

# Project Introduction

The idea for the team project was born out of necessity, when we realized that it was difficult for us all to schedule weekly meetings without having to spend a large amount of time discussing our timetables for the week. Whilst we knew of some applications that helped schedule meetings around timetables, none of us could get those applications to work well on our mobile phones. With smart phones becoming increasingly powerful and more and more students relying on them instead of either laptops or personal computers, we concluded that a mobile application available to students would help address the problems presented by scheduling better than computer based solutions.

The application would initially be targeted at students and faculty of the Nottingham Trent University. With the University timetable already somewhat integrated with mobile calendars, it would be fitting to build an app that would automatically also add scheduled meetings to that timetable, so that people would not accidentally miss out on scheduled meetings or deadlines. By working with the NTU as opposed to another customer we can also assure that we get a wide range of opinions about what our app should do; the feedback opportunities from such a wide range of potential users outweigh, in our opinion, the advantages gained by working with a single customer. By working with the University as our target group we can ensure that:

* We can easily relate to our potential users, being students ourselves. This will help us both understand and implement the feedback received. We will also be more invested into the project as potential users.
* We have easy access to near instant feedback. Working with a singular remote customer can mean waiting back to hear from the user on what features he does and does not like. By working with the University as our customer we can ensure quicker feedback.
* We can test the application more easily. With a wide body of students available, we can ensure that each test is done more quickly and more fairly (Due to collecting a range of opinions instead of just one) than when working with a single customer.
* It is easier to integrate our application with the existing system if we have technical experts on using and maintaining the system already present on the Campus. We are also all users of the existing NTU systems so we do not need to spend as much time evaluating the existing system.

In the research conducted for applications that already existed and offered similar features we found a few, with the main front runner being the service offered by Microsoft Outlook. The application, whilst primarily dealing with emails, also offered users the opportunity to schedule meetings and integrated it with pre-existing timetables. However, we also found some problems with the application:

1. We could not schedule meetings from a mobile platform. The phones we used operated on Android and iOS and we were not able to set up meetings on mobile easily. This is a problem when considering the previously mentioned increased reliance on mobile phones.
2. When we scheduled a meeting, it was not added to our calendars. The only feedback we got about the meeting was a short email.
3. There were no alerts on our mobile phones that the meeting is happening.

Our idea would aim at addressing all three of these key issues, as well as adding a lot more functionality in the form of meeting priorities. We will also address the mobile phone issue by targeting the application onto a mobile platform. Due to the majority of us having Android phones, as well as the programmers having some experience working with Android already, we will use the Android Operating System as our platform of choice.

The potential market for such an application is very broad. Whilst the application is aimed at students of the University initially, it can be used outside out of the University if integrated with Android Calendars. Android OS is by far the biggest mobile operating system platform, and also showing steady growth since 2012 (From 69.3% in 2012 to 82.8% in Quarter 2 of 2015), so it will offer a large market for potential users. Reference 1

# Aim

To design and implement a mobile-based calendar and timetable application that streamlines the organization and collaboration process within large groups of people.

# Objectives

* The user-interface design of the application must be user-friendly, and preferably emulate the native calendar applications on their respective operating systems.
* There will be a testing period shortly before the application is complete. This will consist of a peer review group large enough to test the application adequately.
* The application must be able to communicate with other users of the application, whether that be using real-time synchronization or push/pull from server to client.
* User input should be minimal and therefore much of the applications processes will ideally be automated. For example, the user will not have to enter their timetable and events manually, instead the application should be given access to this information, provided it is stored locally on the device.
* The application will allow the user to easily enter their own timetable and calendar events, as well as modify, delete and hide items of their choosing.
* Design a fully functional application, as well as the relevant supporting documents by each deadline as detailed in Appendix B.

# Functional Requirements

Requirements we must have

* Functionality to ask users for permission to access their timetable
* Users should be able to compare timetables with other users of the application
* Option to synchronize the timetables of all group members
* SQL Database used to store all timetable/calendar information
* Ability to hide certain timetable events from other users
* Be able to set re-occurring meetings, such as bi-weekly, annual, semi-annual etc.
* An easy to use interface, which is equal in aesthetics and usability to that of the native timetabling applications.

Requirements we would like to have

* Group chat functionality, with the ability to share snippets of timetables and discuss work.
* Cross-platform functionality, such as Android, iOS and Windows phones.
* Utilization the universities IT finder, to find rooms containing an adequate amount of free computers
* A feedback form for users to send their thoughts about the application

Ambitious requirements

* A basic file-sharing system incorporated into the possible group chat feature
* A campus map visualizing agreed meeting places – could be similar to the existing NTU student application map

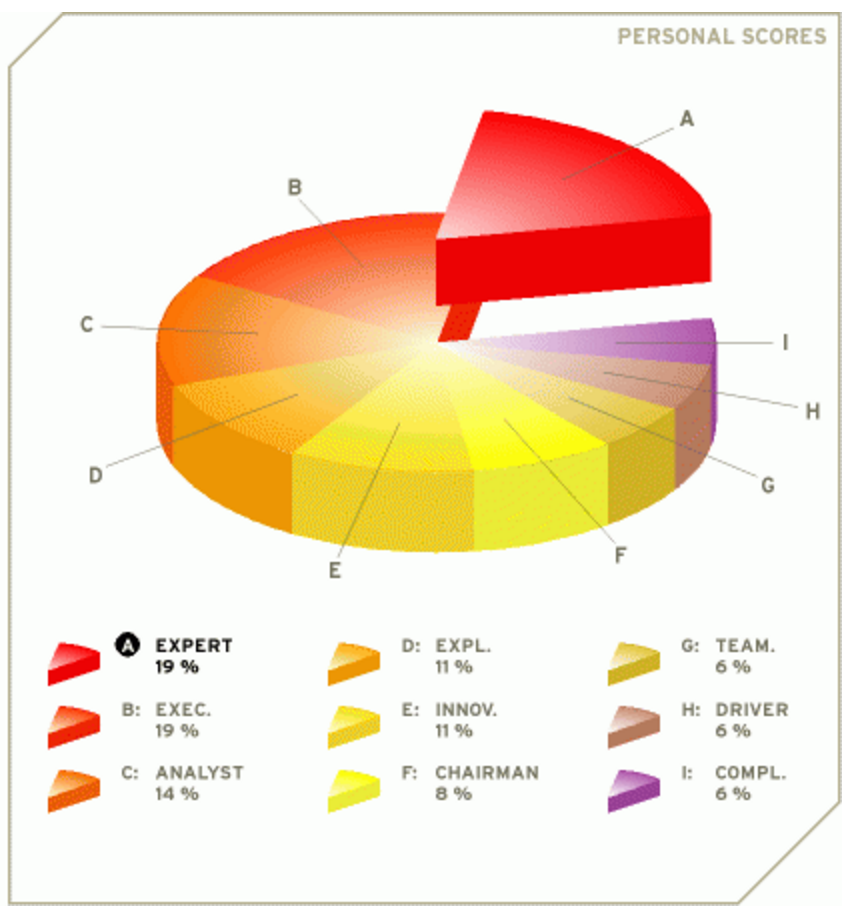
# Project Management

During the first meeting that took place, the group decided that based on past experiences with group work, it would be sensible to hold at least two meetings a week with the possibility of extra when deadlines are approaching or when issues arise. Here we discussed things that had gone well in past groups, and things that had not and came up with a structure that would reflect these thoughts. We also delegated roles and assigned work that each member of the group will carry out. Below are the decisions we all agreed upon;

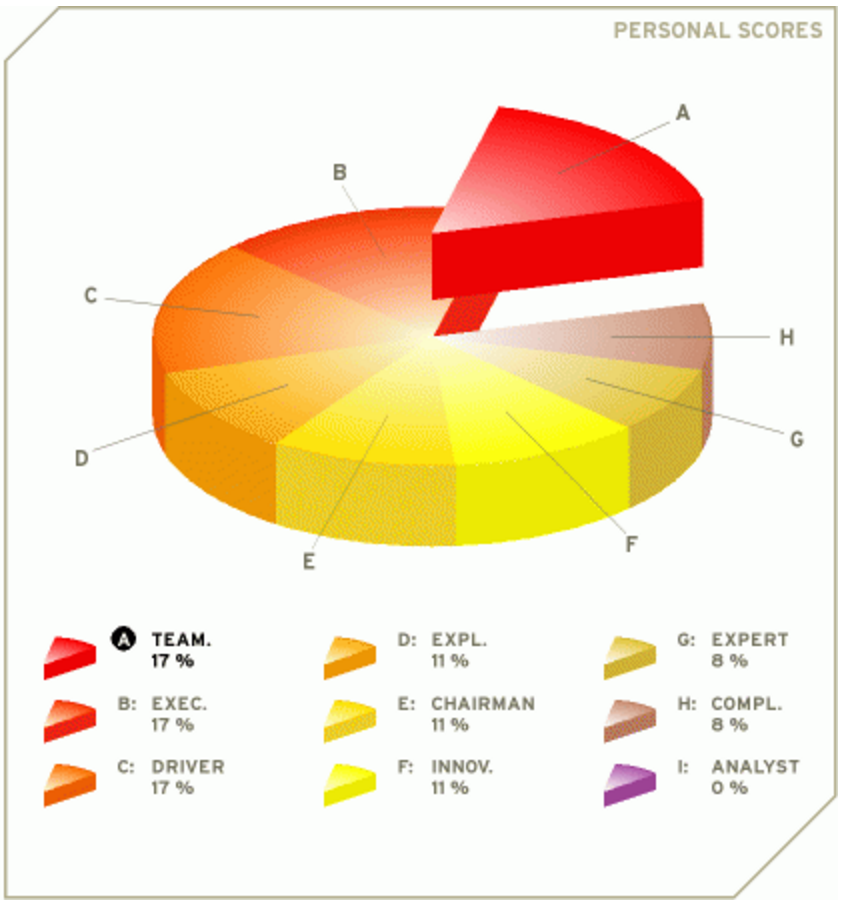
* Ashley Thompson will be the project leader, meaning he is responsible for arranging group meetings, booking study rooms, delegating work and should be the first person people arise any issues with.
* Mateusz Januszewski will act as backup team leader in extenuating circumstances.
* Nirmoldeep Bajwa will be in charge of recording the minutes and topics covered during the meetings.
* Ashley Thompson will take minutes in the absence of Nirmoldeep.
* The group will aim to hold at least two meetings of around 30 minutes each week. – These will act as a review of the group’s overall progress and should help to give direction and solve any individual issues the development team may be having.
* The first meeting will take on Monday mornings, with the second being every Wednesday afternoon.
* The group will use a Dropbox and/or similar online file sharing websites to store the project documents. This will help ensure all member of the team will have constant access to all aspects of the project. In addition to this, this should help to reduce data redundancy and corruption, therefore acting as a backup in-case of local file corruption/loss.
* Ashley Thompson will be responsible for the collection and compilation of all separate documents.

# Team Responsibilities

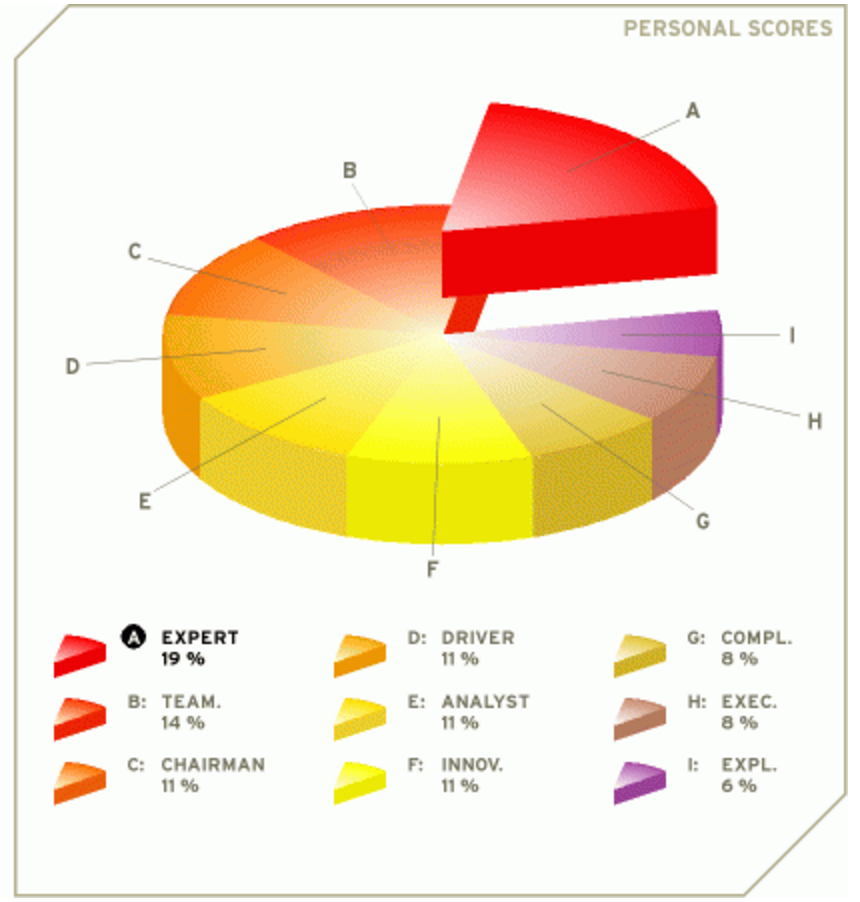
Ashley Thompson – Ashley’s team role test identified Ashley primarily as an executive. This is also sometimes referred to as the organizer within a group, meaning he should theoretically make a good leader. This is because he should be efficient, practical and systematic, as well as possessing the ability to turn ideas into concrete actions.



Mateusz Januszewski – Mateusz’s test highlighted that he is an expert, meaning he has the skills and expertise necessary for the specific tasks at hand. This is especially helpful as the group has identified Mateusz as the main programmer on the project. Mateusz’s test also showed that he will be strong and independent enough to overcome any programming issues that arise, as demonstrated by his high Executive percentage.



Nirmoldeep Bajwa – Nirmoldeep has been identified as a team player, acting as a mediator to the group. He is someone that should bring resolve to issues and disagreements, making sure everyone is happy with the outcome decided upon. He also has a high executive rating, showing that our group should be full of strong, independent works that should be able to take charge of their own work and ensure they are making progress.



Charalambos Foti – Charalambos has also been identified as an expert, showing that he will be comfortable in a role that requires concentration and effort in a particular, specific field. For example, Charalambos should be adept at programming, and as he also has a high team player rating, should act as a good aid to Mateusz.

# Team Responsibilities

**Ashley Thompson**

Ashley is appointed as the project manager/leader after we all agreed that he is driven and can successfully lead a team in the creation of the application. The team’s leader has to allocate the work and arrange the group meetings, and is generally responsible for the smooth running of the development process. He is also responsible for building the actual main documents when everyone has completed their tasks that are assigned as well as keeping the correct layout and format through the entire document. In addition, he is the backup programmer for Mateusz who will be the main programmer of our application.

**Nirmoldeep Bajwa**

Nirmordeep has been appointed as the team’s main designer as he has been bringing up new and fresh ideas about what we can achieve and aim as well as how to design specific areas or the project. He will work with Charalambos on this as they will design the parts that are needed for the project set by the manager of the team.

**Mateusz Januszewski**

Mateusz has been appointed as the teams’ main programmer as it is his key skill, being confident and reassuring that all programming should pass through him. He has Ashley for the backup programmer and together they can successfully develop the adequate code that our app requires.

**Charalambos Foti**

Charalambos has been appointed as a designer also, meaning he will help with the structure and planning of the work, working with Nirmoldeep to look for certain areas that could be improved or edited. He is willing to help in all areas of the project, as well as any work that is assigned by the project manager.

# Sources of Information and Resources

In order to develop the application, two pieces of software will be used in concurrence:

1. Android Studio. Android studio is a simple code editor that allows users to develop Android Apps. As we have decided to develop our application on the Android platform due to how common it is, and how quick it is growing, Android Studio is the perfect choice for an application. The Studio offers a smart code editor, code templates, git hub integration for easy backups (Thus helping minimize the risk of data loss) and all manner of virtual device emulation.
2. Visual Studio 2015. The newest Visual Studio has come released with some Android development tools. Whilst Android Studio will be the main piece of software used, VS2015 will be the backup. We all also have experience working with this piece of software.

We will use the Android Software Development Kit with both of the programs mentioned above. We will also make use of Google Calendars to download user calendar information into our application. Whilst we do not need to rely on hardware due to the large amount of available emulators, we will use three different Android-operating hardware devices to test our application outside of the simulated environment.

# Risk Assessment

(See Appendix A)

# Professional, Legal, Social and Ethical issues

**Professional**

* All work that does not belong to the group must be referenced and must not be claimed as ours. (Any code that has been used must be referenced to the correct author)
* Application must not injure others (physically or psychologically, their property or reputation.
* Carry out professional duties with due care and must not use authority for any own personal gain.

**Legal**

* Must not breach any data protection laws
  + Must ask permission before sharing other member’s timetables
  + Must not require any app permissions that it doesn't need i.e. access to device information, text messages or call history.
* Must only access the private data (i.e. timetables, personal data, images) when necessary

**Social**

* Must be able to set some events private so that the user has control over what the third party sees.

**Ethical**

* All professional activity must be carried out in a way so that there is no discrimination on the grounds of gender, sexual orientation, nationality, colour, religion, ethnicity, marital status, age, disability or any other condition or requirement.
* Promote equal rights and equal access in the work.

# Project Plan, Milestones and Timescale

(See Appendix B)

# References

Reference 1:http://techcrunch.com/2015/08/20/peak-android/#.od0s8ma:DG2b

# Appendix A

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Risk ID | Risk Name and Impact Area | Risk score (1 low, 4 high) | Risk Impact | Probability of risk event | Description | Response | Triggers |
| 1 | Scheduling problems | 1 | Low | Medium | Problems can arise when trying to schedule meetings. | Re-arrange meetings to fit schedules. If not possible, communicate via other means. | -Team members having busy schedules  -Team members having irregular schedules  -Incorrect planning of meetings |
| 2 | Inability to work together | 2 | High | Low | Inability to work together and a hostile group atmosphere. | Try to address any issues someone might have with other group members. | -Team members disliking each other  -Team members feeling they are doing more than others |
| 2 | Lack of Communication | 2 | High | Low | Team members not communicating enough with others. | Arrange more meetings and seek alternate means of communication. | -Team members not attending meetings  -Team members not having internet access or other member phone numbers |
| 3 | Bad time management | 2 | Medium | Medium | Inability to use time correctly and wasting time. | Revise schedules and make sure team members are motivated. | -Team members not being motivated to do the work  -Discussing irrelevant things in meetings |
| 4 | Short term inability to attend meetings | 1 | Low | Medium | Inability to attend a single meeting. | Make sure team member gets meeting minutes and is clued in. | -Short term illness  -Travel problem  -Laziness  -Bad scheduling |
| 5 | Long term inability to attend meetings | 2 | High | Low | Inability to attend a few meetings in a row. | Discuss meetings with team member. | -Long term illness  -Work  -Laziness |
| 6 | Short term inability to meet targets and deadlines | 3 | Medium | High | Work is delivered a little late of soft deadline. | Make sure deadlines can be met via discussion. | -Laziness  -Team member struggling with work  -Illness |
| 7 | Long term inability to meet targets and deadlines | 3 | High | Medium | Inability to meet hard deadlines consistently. | Discuss issue with members. | -Laziness  -Team member struggling long term  -Other commitments |
| 8 | Misattribution of roles and incorrect division of tasks | 2 | High | Low | Team members not fit for their roles. | Rearrange roles if possible. | -Team member not being there when roles assigned or misevaluating own abilities |
| 9 | Lack of punctuality | 1 | Low | Medium | Team member being late to meetings. | Resolve if short term, rearrange if long term. | -Travel difficulties  -Bad meeting times |
| 10 | Indecisiveness of Project Manager | 2 | High | Low | Team manager not making decisions. | Team member should contact customer or talk to other group members. | -Unmotivated team member  -Unclear project definitions from customer |
| 11 | Indecisiveness of entire group | 2 | High | Low | Group not making decisions. | Talk to customer and have a group discussion about the issue. | -Unclear project definition  -Unclear targets |
| 12 | Ignoring feedback from customer | 1 | Medium | Low | Ignoring customer feedback accidentally or on purpose. | Revise feedback, apologize to customer and implement feedback. | -Not being thorough when reading feedback  -Not liking the customer feedback and trying to ignore it |
| 13 | Misunderstanding customer feedback | 2 | Medium | Medium | Not understanding customer feedback properly. | Revise feedback, discuss with customer | -Unclear feedback  -Not taking long enough to analyse feedback |
| 14 | Not reporting to customer often enough | 1 | Medium | Low | Reports not being frequent. | Have a schedule of reports. | -Not scheduling reports after asking customer |
| 15 | Making an erroneous report to the customer | 4 | High | High | Making a mistake in a report | Take care when drafting reports | -Not being careful enough when drafting |
| 16 | Not being able to implement all features | 2 | High | Low | Not implementing features as specified | Make sure features are not too ambitious and that team members can handle them | -Too ambitious features  -Too many features  -Inability to program  -Not enough programmers |
| 17 | Inability to obtain hardware for tests | 1 | Low | Low | Cannot get the hardware required to test application. | Use emulators of hardware or chose a widely available platform for tests. | -Using an obscure platform  -Using an expensive platform |
| 18 | Data Loss | 4 | High | Medium | Lose valuable project data | Make sure to backup | -Lack of backups  -Using vulnerable storage methods |
| 19 | Inability to program software | 3 | High | Medium | Programmer is unable to program the software either due to lack of technical skills or lack of time. | Chose competent programmers, chose a common platform and assign additional programmers as needed. | -Using obscure languages  -Using bad programmers  -Making up difficult to implement features  -Choosing a platform with lack of technical manuals |
| 20 | Running out of time when implementing | 4 | High | High | Running out of time to implement specified features. | Make sure enough programmers are on the job. | -Too few programmers  -Too many features |

# Appendix B