System Analysis & Design 20/21

Jak Waterfall

Matthew Gazzard

Harry Heskin

Pirajan Paramanantham

1 Contents

2 Executive Summary

In the following System Analysis and Design report, the team will plan and implement a structure to draft and develop a virus contact tracing application, built for a global-pandemic scenario. Within this report, the team will analyse requirements and time constraints, choose an adequate software development life cycle to match, create a variety of detailed diagrams that visualise the architecture of the system, and solve some intricacies regarding the functionality of the proposed solution. Features such as system behaviour, database interaction and user interface design will also be considered within the report. Overall, the group aims to fully assess the requirements of the brief, so that a robust and effective application can be assembled.

3 Introduction

3.1 Aim and Initial Ideas

Our team has been tasked with the responsibility of producing a systems proposal report for aiding and helping slow down the spread of COVID-19 on the ABC University campuses. The aim is to create a COVID-19 contact tracing app for use by all the university staff and students that will track all the positive cases in real time; this vital information will be used by the university management team to help guide them on making decisions for the near future.

The app will require the user to undertake a quick health questionnaire to gauge the user's current health risks from the virus as stated by the NHS. This information will be used to inform the user of the specific level of risk they take from exposure to the virus. The app will also include a heat map of all the campuses to give visual information to the user on where the hot spots are and using this data, we will populate a list of the top places to avoid for that day to help mediate the risk of infection.

All users are asked to record their health daily; reporting on all symptoms of COVID-19 as stated by the NHS such as temperature, coughing, tiredness, loss of taste or smell. Personal information will be kept private and confidential to protect user anonymity. News or updates on COVID-19, the universities policies or government guidelines such as lockdown dates or university closures will be curated and presented within the app.

Key locations will be marked on the map such as testing and treatment facilities with general directions provided. The system's own GPS application may also be used. The app will have a curated list of local shops and restaurants that deliver and a list of volunteers that have

signed up to help deliver essentials to those who are isolating.

The app will be designed in such a way that it will meet the needs of any subsequent pandemic or large crisis in the future and will be a great tool in helping to keep staff and students safe.

3.2 Review of Contact Tracing Systems

There are many contact tracing systems currently in use throughout the world's general population because of COVID-19. A great amount of data on their efficacy is already known, for example, in a study by Kucharski, et al. (2020) who used BBC pandemic data found that under certain control measures that combining testing and contact tracing strategies they reduced the R number (effective reproduction number) more than mass testing or self-isolation alone. Kucharski, et al. (2020) state they also found that contact tracing measures "substantially reduced the probability that a primary symptomatic case would generate more than one secondary case."

Almagor & Picascia (2020) found in their study that a contact tracing app can "substantially" contribute to reducing the rates of infection only when "accompanied by a sufficient testing capacity or when the testing policy prioritises symptomatic cases." This would suggest that testing the population is highly important for the efficacy of contact tracing systems. They say that contact tracing apps are a "viable epidemic mitigation strategy" and that if a large proportion of the public embrace contact tracing apps "the spread of the virus" will be "increasingly reduced". The strategy of tracking positive cases and informing them to get tested and to self-isolate will help reduce the spread and infection rates of the whole population, "including those not using CTA" (contact tracing apps).

Conversely, Cochrane (2020) states that while contact tracing systems may reduce the number of secondary infections, this rate is exceeded by manual contact tracing with self-isolation. Digital systems like apps are faster and less prone to data loss than a paper system however they assert "It is unlikely that digital technologies would be the sole method of contact tracing" and that these systems are still unproven to work in the real world and would prescribe a blend of both digital and manual contact tracing for better results.

For ABC University, a strategy which mandates the use of the app while people continue to use the university's facilities would be the most beneficial. This is because the research shows that having as many people as possible embracing the app will increase the app's efficacy. Testing must also be paramount within the university's policies and culture if a

contact tracing system is to be used to its full potential.

3.3 The Team

The system analyst team in ABC University has been tasked with developing a system proposal for a contact tracing application. Matthew is the leader of the team and has been working with Jak, Harry and Pirajan to complete the proposal. The team worked closely together to complete this report with many proposal elements being fulfilled during regular meetings. Not all work was completed during team meetings so work was also assigned by the team leader to be finished by the next scheduled session. The personal strengths of individuals were considered before work was assigned to them, helping to ensure sections of the proposal were efficiently handled and comprehensive. When tasked with the delivery of the proposal, the team thought it essential to discuss the key considerations for contact tracing systems.

3.4 Key Considerations

Firstly, the number of users that engage with an app, no matter the kind, is critical. If too few students and staff of ABC University adopt the contact tracing app, its impact will be minimal. Significant barriers to adoption could be user concerns about privacy and security. Kesswani, Lyu and Zhang (2018, pg. 39541) discuss how downloadable apps, particularly those that are free, have led to “severe privacy concerns.” Users will be required to provide sensitive personal data meaning there needs to be a minimal risk of security failures and data misuse. User trust must be maintained as the efficacy of track and trace on virus numbers depends on users engaging with contact tracing apps (Almagor & Picascia, 2020).

Students and staff must also be incentivised to make use of the app if it is developed to avoid wasting ABC University funds. The NHS contact tracing app reportedly cost in excess of £35 million (Downey, 2020). Funding for the ABC University contact tracing app must therefore be applied judiciously to ensure an ample contact tracing system is produced without negatively impacting on other university matters which require funds.

Finally, some students or staff may have limited access to smart devices. This could be due to several reasons such as having a low income or being elderly and having little understanding of modern technology. Referring again to the review by Cochrane (2020) into digital contact tracing systems, it was concluded that digital technologies would likely not be

the only method used for track and trace and that the technology is “largely unproven in real-world outbreak settings”. ABC University needs to be aware that using a contact tracing app in isolation will not be the most efficient way to tackle campus infections.

4 Methodology

4.1 Structured Methodologies

Before considering each of the various development methodologies, the team felt that it was important to ensure the one that was chosen would provide maximum benefit to the overall solution, since each development process would play out differently. They believed it would be crucial to analyse and evaluate as to which one was most applicable to their project brief. The initial planning arrangements that the team looked at implementing were the “structured methodologies”.

The first example was the Waterfall model, one of the earliest techniques for tackling software development. This approach consists of specified phases in which the process moves through construction in a linear and sequential manner. The requirements for this process are identified long before the development begins, while the progression to each layer of the cycle depends solely on the deliverables of the previous layer. This means that the importance of achieving these deliverables is paramount within the waterfall methodology. This system was introduced by Dr Winston Royce in the late 1960s to run operations “at an operational state, on time and within costs” (Royce, 1970). The issues that present themselves within the Waterfall model reside within its timings. While the waterfall model presents a structure that is clear and comprehensible from the outset, there are often long waits before visible evidence of the new system is shown. This differs from the brief set out to produce a Covid-19 contact tracing app, as the development and implementation of the product relies on efficiency in a global pandemic scenario. The group felt the long, drawn-out nature of this methodology would not be sufficient in the construction of their solution.

The next model identified, also in the “structured methodology” category, was the idea of parallel development. This system focused on dividing the development procedure into subprojects, allowing for concurrent software development in different areas of the project.

Early forms of this style of production were seen within the work of Michael J. Flynn in the

first iterations of multiprocessing Central Processing Units (CPUs) (McCool, et al., 2012). The nature of the production method means that, compared to the waterfall model, the total project length can be reduced significantly. The model also reduces the chances of requirements changing as the construction process continues and decreases the possible need for rework. However, when put into perspective against all other operations models available in the modern world, the parallel methodology can appear inferior in various ways. The introduction of subprojects means that careful and serious decisions must be made as the project evolves, putting developers under added pressure as the solution progresses. Subprojects in this form can also present the difficult task of combining sub-sections at the end of the program cycle, which can be complex and time-consuming. Therefore, the team decided to part from the overall idea of using a structured methodology for their solution.

Figure 1, (Dennis et al. 2015. Criteria for Selecting a Methodology, p. 16)

	Structured Methodologies		RAD Methodologies			Agile Methodologies	
Ability to Develop Systems	**Waterfall**	**Parallel**	**Phased**	**Prototyping**	**Throwaway Prototyping**	**XP**	**SCRUM**
With Unclear User Requirements	Poor	Poor	Good	Excellent	Excellent	Excellent	Excellent
With Unfamiliar Technology	Poor	Poor	Good	Poor	Excellent	Good	Good
That Are Complex	Good	Good	Good	Poor	Excellent	Good	Good
That Are Reliable	Good	Good	Good	Poor	Excellent	Excellent	Excellent
With a Short Time Schedule	Poor	Good	Excellent	Excellent	Good	Excellent	Excellent
With Schedule Visibility	Poor	Poor	Excellent	Excellent	Good	Excellent	Excellent

4.2 Agile Methodologies

After careful consideration of other methodologies, the team decided that an agile methodology would be best used to develop a contact tracing app. Consulting the table in Figure 1, both XP (Extreme Programming) and Scrum look to be viable for the creation of a contact tracing app (Dennis, et al., 2015, p. 16). Due to the nature of the Covid-19 pandemic, being able to deliver an app within a short time scale is crucial so the app can be released as soon as possible, allowing it to help stop the spread of Covid-19. This is also why schedule visibility is particularly important as ABC University will be able to keep track of progress and see any potential obstacles to development which would delay the release of the app.

Furthermore, reliability of the app is essential as students, staff and the wider public's health is at risk. Choosing a methodology that will help to eliminate issues such as failure to deliver

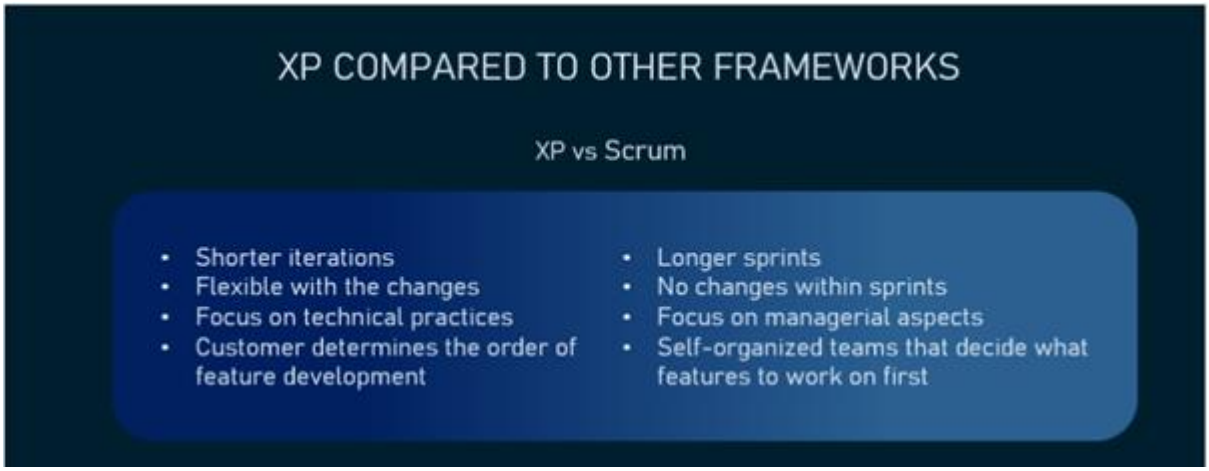
notifications to users, app instability and false negatives or positives is a priority. User requirements may also change as more research is done surrounding Covid-19 and its impact. Figure 1 (Dennis, et al., 2015, p. 16) states how XP and Scrum are both excellent at developing systems that serve to ensure the key issues discussed are tackled in the most efficient way possible. However, using Figure 1 (Dennis, et al., 2015, p. 16) alone to pick a suitable methodology is not possible as they are more different than the table implies.

Figure 2 (Oosterhof, 2016, What scrum, kanban and XP have in common)

As detailed in the Agile Manifesto (Beck, et al., 2001), constant communication between the development team is a cornerstone of any agile methodology. This is particularly true for XP as it normally requires colocation (Wells, 1999). However, Braithwaite and Joyce (2005) discussed how teams can benefit from using XP without colocation. As Covid-19 has led to

social distancing measures being used and sometimes even lockdowns, colocation will not be possible for the development of the app. However, as all normal working practices have been affected by the Covid-19 pandemic and because of evidence from Braithwaite and Joyce (2005), the team decided not to let this have too great an impact on which methodology was chosen for the app development process. When deciding whether to adopt XP or Scrum for the development of the app, the team noted the similarities between the two, resulting in the decision depending on their key differences.

Figure 3 (Altexsoft, 2021, Extreme Programming: Values, Principles, and Practices)



XP iterations are shorter than the sprints of Scrum, lasting from 1 to 2 weeks rather than 2 to 4 weeks (Cohn, 2007). During Scrum sprints, changes are not allowed to be made once the sprint backlog has been decided upon during the sprint planning meeting (Cohn, 2007). As detailed by Cohn (2007) and in an article on XP hosted by Altexsoft (2021), changes can occur during XP iterations. Features can be substituted with other, equivalent sized features if work has not already begun on the feature it is replacing (Cohn, 2007). The team thought this would be very useful due to the ever-changing environment caused by the Covid-19 pandemic. This would improve efficiency when incorporating features desired by ABC University as time would not be wasted on features that were deemed of less importance or no longer being necessary.

The party that decides on the importance of features is also another key difference between Scrum and XP. In Scrum, the product owner sets the product backlog however the Scrum team will choose the order this is delivered in, sometimes not picking the highest priority item for their sprint backlog (Cohn, 2007). Cohn (2007) explains how XP differs because an XP team must work in a “strict priority order” set by the customer. As contact tracing apps are

relatively unfamiliar due to only being recently conceived, the team thinks that it would be beneficial for the customer to decide on what is a priority. Finally, XP has a heavy focus on engineering practices, explicitly mandating the use of automated testing, refactoring and pair programming (Cohn, 2007). This is unlike Scrum where the team can decide whether to incorporate such procedures (Cohn, 2007). As the team is developing a contact tracing app where reliability is essential, it is imperative that as much testing as possible is performed during development to ensure delivery of a stable product. A more reliable app will have a greater impact on the health of the students and staff at ABC University.

XP may have been originally envisaged as a methodology to be used by small teams but it has since proven to be scalable (Beck and Andres 2004; Rumpe and Scholz 2003). The team believes it is the correct methodology to use for the development of a Covid-19 contact tracing app.

5 Project Planning

5.1 Feasibility and Risk Assessment

Table 1 demonstrates the risk assessments and feasibility considerations carried out by the team. All possible circumstances were reviewed within these discussions, reviewing proposed feasibility of the application in terms of technical, organisational and economical aspects. The potential drawbacks found by the team were met with solutions and ways to mitigate those circumstances, these being detailed in the table. Inside the table, the fields are as follows: Area of question feasibility, risk description, severity, ways to mitigate and relevant references to support risk analysis.

Table 1 – Risk Assessment

Area of feasibility	Risk description	Severity	Ways to mitigate	Relevant references
Technical Feasibility	Familiarity with functional area: contact tracing apps.	Medium	Research into other contact tracing apps, how they work and what they do. Make sure the interface is as simplistic and easy to use as possible.	(Burgess, 2020)
Technical Feasibility	Familiarity with technology: the ability to use mobile phones and apps such as contract tracing apps on the user interface.	Low	Make sure the interface is as simplistic and easy to us as possible. Help pages and support given with application.	(Ossmium, 2018)
Technical Feasibility	Project size: maintaining small project size.	Medium	Keep the group sizes for development low to promote conciseness and efficiency of communication between the team.	(Wang & Evans, 2019)
Technical Feasibility	Compatibility: functionality on both Apple and Android interfaces.	High	Cross-platform technology that supports universal access to the software.	(Bozhenko, 2017)
Economic Feasibility	Intangible Costs: no income or revenue from the app to support construction and maintenance of the app - only university funds cover expenses - necessary to save lives.	High	Minimise cost of resources and expensive software used - sophistication of the app is not necessary.	(Downey, 2020)

Economic Feasibility	Tangible costs: cost of renting cloud storage space in databases.	High	Use the minimum amount of cloud storage as possible and avoid high-cost databases.	(Harvey, 2018)
Organisational Feasibility	Strategic alignment: all user data will be kept private and secure in an effective way.	High	Come to a sensible decision on whether decentralised or centralised storage methods are used to maximise security.	(Wright, 2020)
Organisational Feasibility	User's acceptance: the app is adopted by university students and staff universally	High	Enforce a University policy to use the app in a mandatory fashion when in a pandemic scenario.	(Norman, 2014)

5.2 Project Scheduling

The project schedule (Figure 4) is a time plan which accounts for all the tasks required to produce a contact tracing system, the plan also displays the people and resources used on the project and which tasks they are assigned to. The schedule is based on the methodology Extreme Programming.

This schedule has three distinct phases; the analysis phase which describes tasks such as creating risk assessments, conducting research, and creating diagrams that outline the software system. The iterations phase which is the main software development life cycle that we have chosen to use (Extreme Programming). The cycle goes though planning, designing, developing, and testing the system in two-week sprints. Finally, the documentation and deployment phase where the user manuals and guides are made, and the systems final deployment is launched.

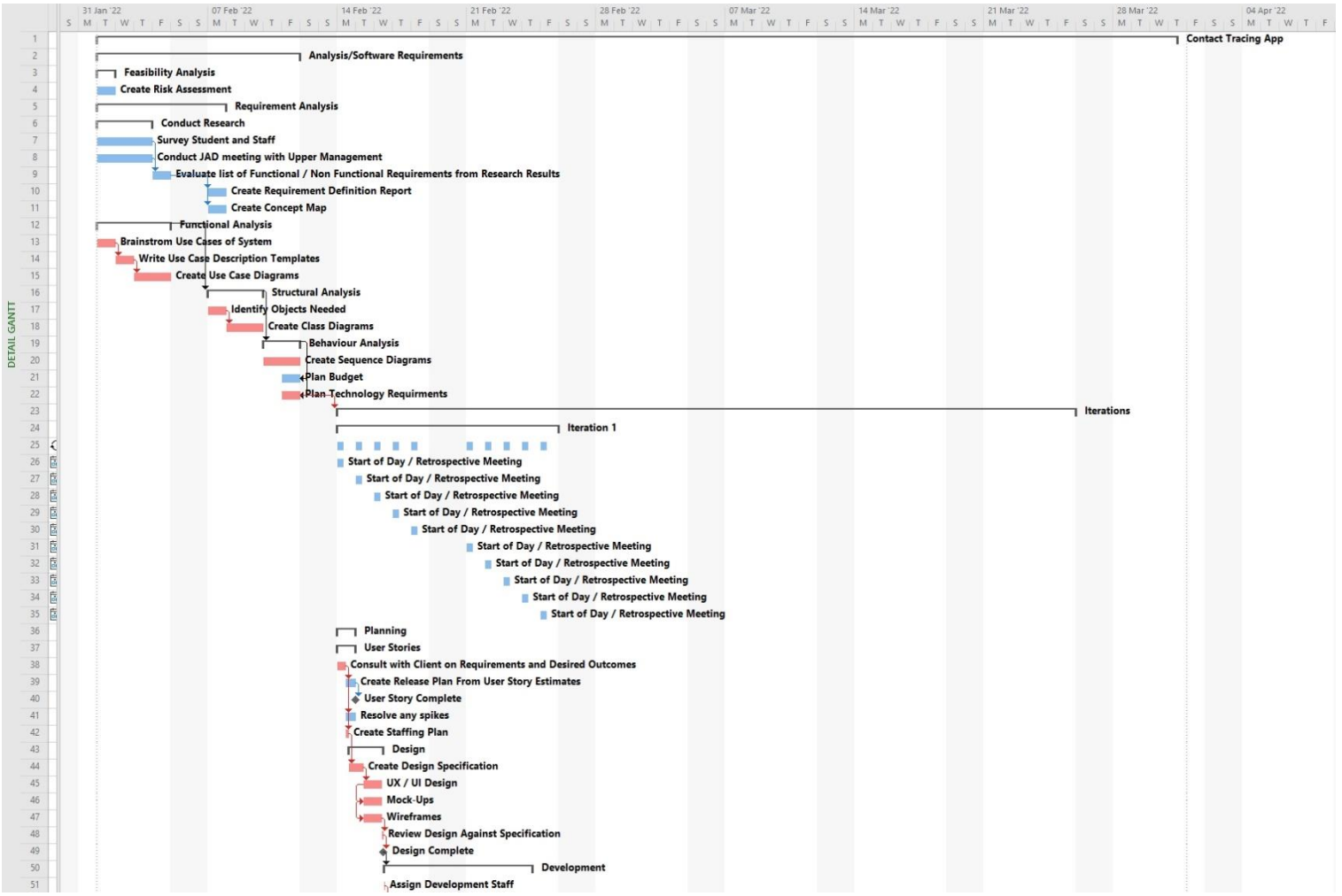
For more information on the project schedule, a PERT diagram with a critical path can be found within Appendix A. The PERT diagram shows the tasks that are critically required to complete the project within the deadline.

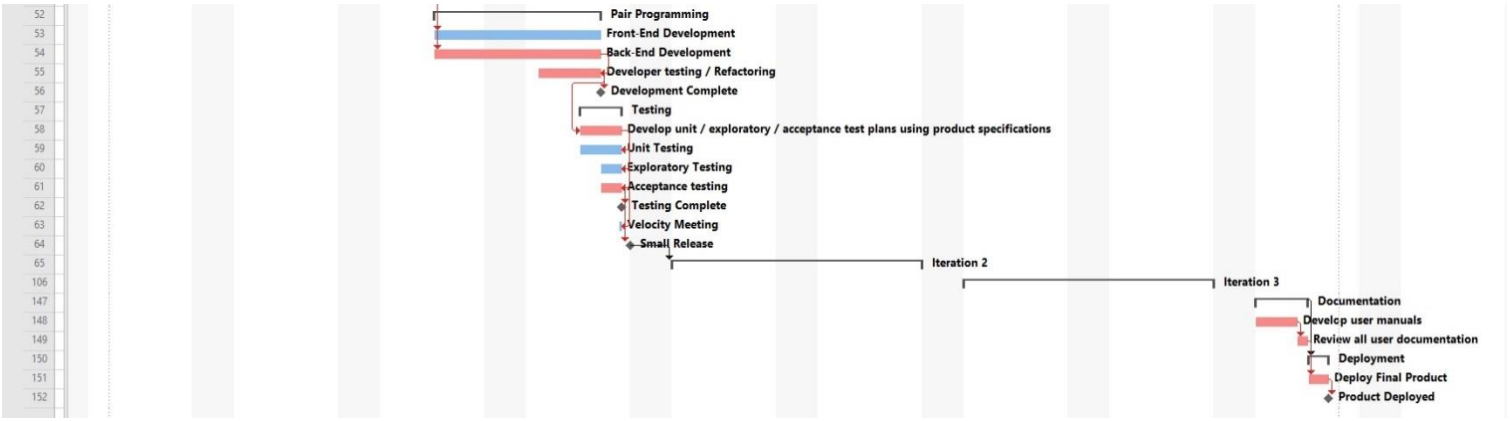
Figure 4 – Project Schedule

DETAIL GANTT

	Task Mode	Name	Duration	Start	Finish	Successors	Resource Names
1		**Contact Tracing App**	**42.5 days**	**Tue 01/02/22**	**Thu 31/03/22**		
2		**Analysis/Software Requirements**	**9 days**	**Tue 01/02/22**	**Fri 11/02/22**		
3		**Feasibility Analysis**	**1 day**	**Tue 01/02/22**	**Tue 01/02/22**		
4		Create Risk Assessment	1 day	Tue 01/02/22	Tue 01/02/22		Alvin,Tina,Omer
5		**Requirement Analysis**	**5 days**	**Tue 01/02/22**	**Mon 07/02/22**		
6		**Conduct Research**	**3 days**	**Tue 01/02/22**	**Thu 03/02/22**		
7		Survey Student and Staff	3 days	Tue 01/02/22	Thu 03/02/22	9	Carol,Jim,Ryan
8		Conduct JAD meeting with Upper Management	3 days	Tue 01/02/22	Thu 03/02/22	9	Charlie,Hannah,Steven
9		Evaluate list of Functional / Non Functional Requirements from Research Results	1 day	Fri 04/02/22	Fri 04/02/22	10,11	Carol,Jim,Ryan
10		Create Requirement Definition Report	1 day	Mon 07/02/22	Mon 07/02/22		Carol,Jim,Ryan
11		Create Concept Map	1 day	Mon 07/02/22	Mon 07/02/22		Charlie,Hannah,Steven
12		**Functional Analysis**	**4 days**	**Tue 01/02/22**	**Fri 04/02/22**	**16**	
13		Brainstrom Use Cases of System	1 day	Tue 01/02/22	Tue 01/02/22	14	Josh,Natalya
14		Write Use Case Description Templates	1 day	Wed 02/02/22	Wed 02/02/22	15	Josh,Natalya,Omer
15		Create Use Case Diagrams	2 days	Thu 03/02/22	Fri 04/02/22		Josh,Natalya,Omer
16		**Structural Analysis**	**3 days**	**Mon 07/02/22**	**Wed 09/02/22**	**19**	
17		Identify Objects Needed	1 day	Mon 07/02/22	Mon 07/02/22	18	Josh,Natalya,Omer
18		Create Class Diagrams	2 days	Tue 08/02/22	Wed 09/02/22		Josh,Natalya,Omer
19		**Behaviour Analysis**	**2 days**	**Thu 10/02/22**	**Fri 11/02/22**	**21FF,22FF**	
20		Create Sequence Diagrams	2 days	Thu 10/02/22	Fri 11/02/22		Alvin,Tina,Omer
21		Plan Budget	1 day	Fri 11/02/22	Fri 11/02/22		Natalya,Josh
22		Plan Technology Requirments	1 day	Fri 11/02/22	Fri 11/02/22	23	Carol,Jim,Ryan,Development tools / Software
23		**Iterations**	**30 days**	**Mon 14/02/22**	**Fri 25/03/22**		
24		**Iteration 1**	**10 days**	**Mon 14/02/22**	**Fri 25/02/22**		
25		**Daily Meeting**	**9.06 days**	**Mon 14/02/22**	**Fri 25/02/22**		
26		Start of Day / Retrospective Meeting	0.5 hrs	Mon 14/02/22	Mon 14/02/22		
27		Start of Day / Retrospective Meeting	0.5 hrs	Tue 15/02/22	Tue 15/02/22		
28		Start of Day / Retrospective Meeting	0.5 hrs	Wed 16/02/22	Wed 16/02/22		
29		Start of Day / Retrospective Meeting	0.5 hrs	Thu 17/02/22	Thu 17/02/22		
30		Start of Day / Retrospective Meeting	0.5 hrs	Fri 18/02/22	Fri 18/02/22		
31		Start of Day / Retrospective Meeting	0.5 hrs	Mon 21/02/22	Mon 21/02/22		
32		Start of Day / Retrospective Meeting	0.5 hrs	Tue 22/02/22	Tue 22/02/22		
33		Start of Day / Retrospective Meeting	0.5 hrs	Wed 23/02/22	Wed 23/02/22		
34		Start of Day / Retrospective Meeting	0.5 hrs	Thu 24/02/22	Thu 24/02/22		
35		Start of Day / Retrospective Meeting	0.5 hrs	Fri 25/02/22	Fri 25/02/22		
36		**Planning**	**1 day**	**Mon 14/02/22**	**Mon 14/02/22**		
37		**User Stories**	**1 day**	**Mon 14/02/22**	**Mon 14/02/22**		

38	Consult with Client on Requirements and Desired Outcomes	3 hrs	Mon 14/02/22	Mon 14/02/22	39,41,42	Charlie,Omer
39	Create Release Plan From User Story Estimates	5 hrs	Mon 14/02/22	Mon 14/02/22	40	Charlie,Jim,Katie
40	User Story Complete	0 days	Mon 14/02/22	Mon 14/02/22		
41	Resolve any spikes	5 hrs	Mon 14/02/22	Mon 14/02/22		Natalya,Omer,Ryan,Tina
42	Create Staffing Plan	3 hrs	Mon 14/02/22	Mon 14/02/22	44	Alvin
43	**Design**	**1.75 days**	**Mon 14/02/22**	**Wed 16/02/22**		
44	Create Design Specification	4 hrs	Mon 14/02/22	Tue 15/02/22	45	Carol,Josh
45	UX / UI Design	1 day	Tue 15/02/22	Wed 16/02/22	46SS	Charlie,Jim
46	Mock-Ups	1 day	Tue 15/02/22	Wed 16/02/22	47SS	Katie,Natalya,Hannah
47	Wireframes	1 day	Tue 15/02/22	Wed 16/02/22	48	Ryan,Tina,Omer
48	Review Design Against Specification	2 hrs	Wed 16/02/22	Wed 16/02/22	49	Alvin,Steven
49	Design Complete	0 days	Wed 16/02/22	Wed 16/02/22	50	
50	**Development**	**6.13 days**	**Wed 16/02/22**	**Thu 24/02/22**		
51	Assign Development Staff	1 hr	Wed 16/02/22	Wed 16/02/22	53,54	Carol
52	**Pair Programming**	**6 days**	**Wed 16/02/22**	**Thu 24/02/22**		
53	Front-End Development	6 days	Wed 16/02/22	Thu 24/02/22		Carol,Hannah,Josh,Omer[80%],Steven,Tina
54	Back-End Development	6 days	Wed 16/02/22	Thu 24/02/22	55FF	Alvin[80%],Charlie,Jim,Katie,Natalya,Ryan[50%]
55	Developer testing / Refactoring	3 days	Mon 21/02/22	Thu 24/02/22	56,58FS-1 day	Ryan[50%]
56	Development Complete	0 days	Thu 24/02/22	Thu 24/02/22		
57	**Testing**	**2 days**	**Wed 23/02/22**	**Fri 25/02/22**		
58	Develop unit / exploratory / acceptance test plans using product specifications	2 days	Wed 23/02/22	Fri 25/02/22	59FF,60FF,61FF	
59	Unit Testing	2 days	Wed 23/02/22	Fri 25/02/22		Alvin[20%],Omer[20%]
60	Exploratory Testing	1 day	Thu 24/02/22	Fri 25/02/22		Hannah,Josh
61	Acceptance testing	1 day	Thu 24/02/22	Fri 25/02/22	62,63FF,64	Katie,Carol
62	Testing Complete	0 days	Fri 25/02/22	Fri 25/02/22		
63	Velocity Meeting	2 hrs	Fri 25/02/22	Fri 25/02/22		
64	Small Release	3 hrs	Fri 25/02/22	Fri 25/02/22	65	
65	**Iteration 2**	**10 days**	**Mon 28/02/22**	**Fri 11/03/22**		
106	**Iteration 3**	**10 days**	**Mon 14/03/22**	**Fri 25/03/22**		
147	**Documentation**	**2.5 days**	**Mon 28/03/22**	**Wed 30/03/22**	**150**	
148	Develop user manuals	2 days	Mon 28/03/22	Tue 29/03/22	149	Carol,Josh,Omer,Tina
149	Review all user documentation	4 hrs	Wed 30/03/22	Wed 30/03/22	151	Carol,Josh
150	**Deployment**	**1 day**	**Wed 30/03/22**	**Thu 31/03/22**		
151	Deploy Final Product	1 day	Wed 30/03/22	Thu 31/03/22	152	Alvin
152	Product Deployed	0 days	Thu 31/03/22	Thu 31/03/22		




52
53
54
55
56
57
58
59
60
61
62
63
64
65
106
147
148
149
150
151
152
Pair Programming
Front-End Development
Back-End Development
Developer testing / Refactoring
Development Complete
Testing
Develop unit / exploratory / acceptance test plans using product specifications
Unit Testing
Exploratory Testing
Acceptance testing
Testing Complete
Velocity Meeting
Small Release
Iteration 2
Iteration 3
Documentation
Develop user manuals
Review all user documentation
Deployment
Deploy Final Product
Product Deployed

6 Requirement Analysis

6.1 Requirement Capturing Method

Requirement analysis is the most important stage of system development according to Dennis et al. (2015). As requirements analysis is the stage of the process where details of what the system will become are discussed, ensuring care is taken in this will result in fewer problems occurring as the system progresses. Reducing the possibility of scope creep is also key as the timely delivery of a contact tracing app to ABC University is a top priority, overrunning the schedule and cost would greatly impact the team's ability to deliver on time.

When making the decision to determine how the team was going to gather requirements for the contact tracing app, they considered the most common methods: interviews, questionnaires, observation, Joint Application Development (JAD) and document analysis (Dennis et al. 2015). From these methods, it was decided that a questionnaire and electronic JAD (e-JAD) would be the most efficient. Multiple methods were chosen as according to Dennis et al. 2015, p. 95, "The best analysts thoroughly gather requirements using a variety of techniques." As time is a factor, it was decided by the team to use two requirement gathering techniques rather than a larger variety as this would delay app production. The team discounted observation and document analysis because they would provide little benefit whilst being time consuming and difficult to perform given the current social distancing measures. Interviews were also discounted as JAD follows the same principles of design and structure as interviews and the team has limited time to gather requirements (Dennis et al. 2015).

A questionnaire was discussed as an effective way of gathering feedback from students along with staff and any other potential users. Dennis et al. (2015) discuss how email questionnaires have markedly higher response rates in comparison to web-based questionnaires. Making the questionnaire available to all ABC University staff and students via email would allow for a large, relevant sample. As many people attending or working at the university as possible will be expected to engage with the app so their opinions are necessary to create an app they are inclined to use. The team decided that the questionnaire should be released and have its results reported on before the e-JAD process begins. This is so the results of the questionnaire can be incorporated into the discussion and decision-making processes of e-JAD. Requirements gathering is helpful for "building political support for the project and establishing trust and rapport between the project team building the system and the users," (Dennis et al. 2015). The team believed a questionnaire

would be most effective at doing this as it asks the user base to directly contribute to development, showing that the team values the users.

E-JAD was chosen over regular JAD for several reasons. Firstly, social distancing measures prevent a group of participants attending a JAD session in person, especially if they do not work within the same workplace. Furthermore, using groupware means that participants remain anonymous, leading to greater participation from contributors as they are more likely to speak truthfully and without fear of retaliation, preventing individuals from dominating the session (Dennis, et al., 2015; Carmel, et al., 1992). A journal article by Dennis, Hayes and Daniels (see Dennis et al. 2015, p. 102) even suggests that e-JAD can “reduce the time required to run JAD sessions by 50 to 80 percent.” Once again, as timely delivery of the app is a factor, the team thought the use of a time efficient process was best. It would be remiss of the team to ignore that some studies mention how the difficulty of sitting in one place, particularly whilst using computing equipment, for extended periods of time can be problematic and distracting (Carmel, George and Nunamaker Jr. 1992). However, the team decided that e-JAD would still be preferable despite this problem as alternatives are currently limited. As previously stated, social distancing measures do not allow for collocation in the workplace so the majority of office work is being performed remotely, making it essential to use IT based solutions to achieve goals. ABC University may also not have a JAD trained facilitator, meaning the cost of a consultant would have to be considered as experience in JAD is vital to achieving the best possible session outcomes (Yatco, 1999). Hiring a consultant does have its positives as the JAD facilitator must remain neutral and training in-house members of staff in JAD can be costly (Dennis et al. 2015). A consultant is also potentially more likely to maintain neutrality as they are not part of ABC University themselves and it is unlikely they will have pre-existing personal relations with JAD participants. Facilitator neutrality is further helped by the participant anonymity enabled by e-JAD.

6.2 Requirement List

The requirements list is made up from the results we gathered from the student survey (see Appendix B) and the results of the E-JAD meetings which are detailed in Appendix C.

Functional Requirements

Process-oriented

The system must allow the user to view a heat map of high traffic areas. This feature will help mitigate the risk of infection by lowering the possible number of people the user may interact with by providing a real time analysis of the foot traffic at all university campus locations.

The system must allow users to fill out a health questionnaire. This feature will assess the user's unique risk to COVID-19 and provide them instructional health information and guidelines as provided by the NHS to help them keep safe during the pandemic.

The system must notify the user if they have been in contact with an individual positive for COVID-19. The requirement is the backbone of the system and is most important for the contact tracing process. Notifying people to isolate before they start experiencing symptoms can pre-emptively help stop the user from spreading the virus and in the long term lower the effective reproduce rate (R number).

Using the data collected from the traffic heat map the system must populate a list of top places to avoid for that day. This feature will help users quickly decipher immediately where the hot spots are and avoid those areas.

A daily symptom checker must be provided by the system; the user inputs any symptoms they are experiencing that day and the system will suggest isolating and contacting a local health centre if the criteria is met. This is another way the system will root out possible positive cases and remove them from the public.

Information-oriented

A database must be used to hold all the users private encrypted information on.

The system will provide information on local shops and restaurants that deliver food. This will help users stay in social isolation by providing a convenient and quick curated list of all the local eatery options. While menu information will not be available the list will provide links to

external sites to make the orders.

The system will provide a list of local volunteers that have signed up to help people who are socially isolating with tasks such as shopping. The list will be streamed into the app from volunteer and community websites. An option for users to sign up to these programs will also be provided.

Non-Functional Requirements

Operational

The application must run on all devices including both android and iOS, this is to ensure it is adopted by the biggest audience possible. The greater the percentage of the population using the application, the greater the efficacy the system has on helping to control and slow the spread of COVID-19 (Almagor & Picascia, 2020).

The system must have simple and easy to use controls and features. This is to help combat confusion, user error and help integrate the application into daily life by keeping interactions with it quick and precise. This is also because of the wide audience it is intended to be used by as some users may not be smartphone literate. Settings to enlarge text or read out aloud will be present for those with sight impairments.

The application must be able to be adopted to fit any pandemic level scenario. This is a requirement so that should there be any more pandemic events in the future the university will be ready from day one with a solution to help mitigate risk to the staff and students.

Performance

The information provided by the app must run in real time when connected to the internet with the latest data (heat map, policy, guidelines) to draw upon. This will help users make informed decisions regarding keeping them safe from COVID-19.

The system must operate 24 hours a day 7 days a week. This requirement is vital because user's lives outside of university have not been put on hold and the threat of catching the virus is ever present.

Security

The system must keep all user's private information secure. This requirement is important as we are legally obligated to keep user's data private in accordance with the Data Protection Act 2018 and because a system with poor security regarding user's confidential information will be less trusted, thereby disincentivising its use.

Cultural and Political

The system must be focused on the university's ecosystem and be specialised to the university lifestyle to best suit the needs of its target audience instead of being designed for a broader, more general audience.

The system must be integrated into university policy and be a requirement to download and use the app when on university grounds. This is to boost the effectiveness of the system by ensuring the greatest number of participants.

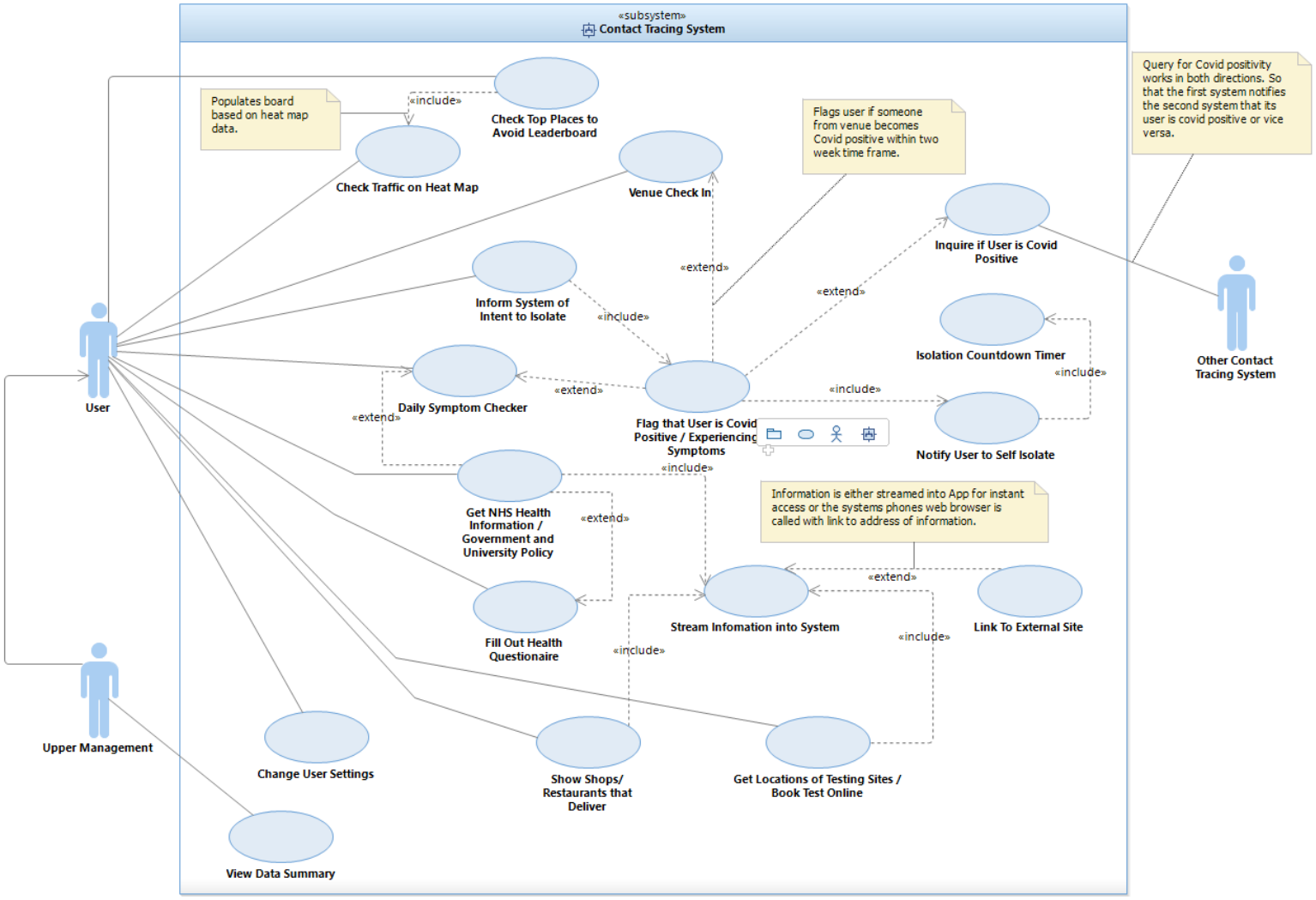
The student union must be involved with the design of the app and its features. This is to represent the ideas of the students and get them on board with using the system by providing them with as many of the features they require from the app.

7 Functional Analysis

7.1 Use Case Diagram

The use case diagram below (Figure 5) represents the way the contact tracing app processes behave with users, each other and other contact tracing systems. They are useful for establishing system architecture. The descriptions for each use case of the system are established in appendix D.

Figure 5 – Use Case Diagram

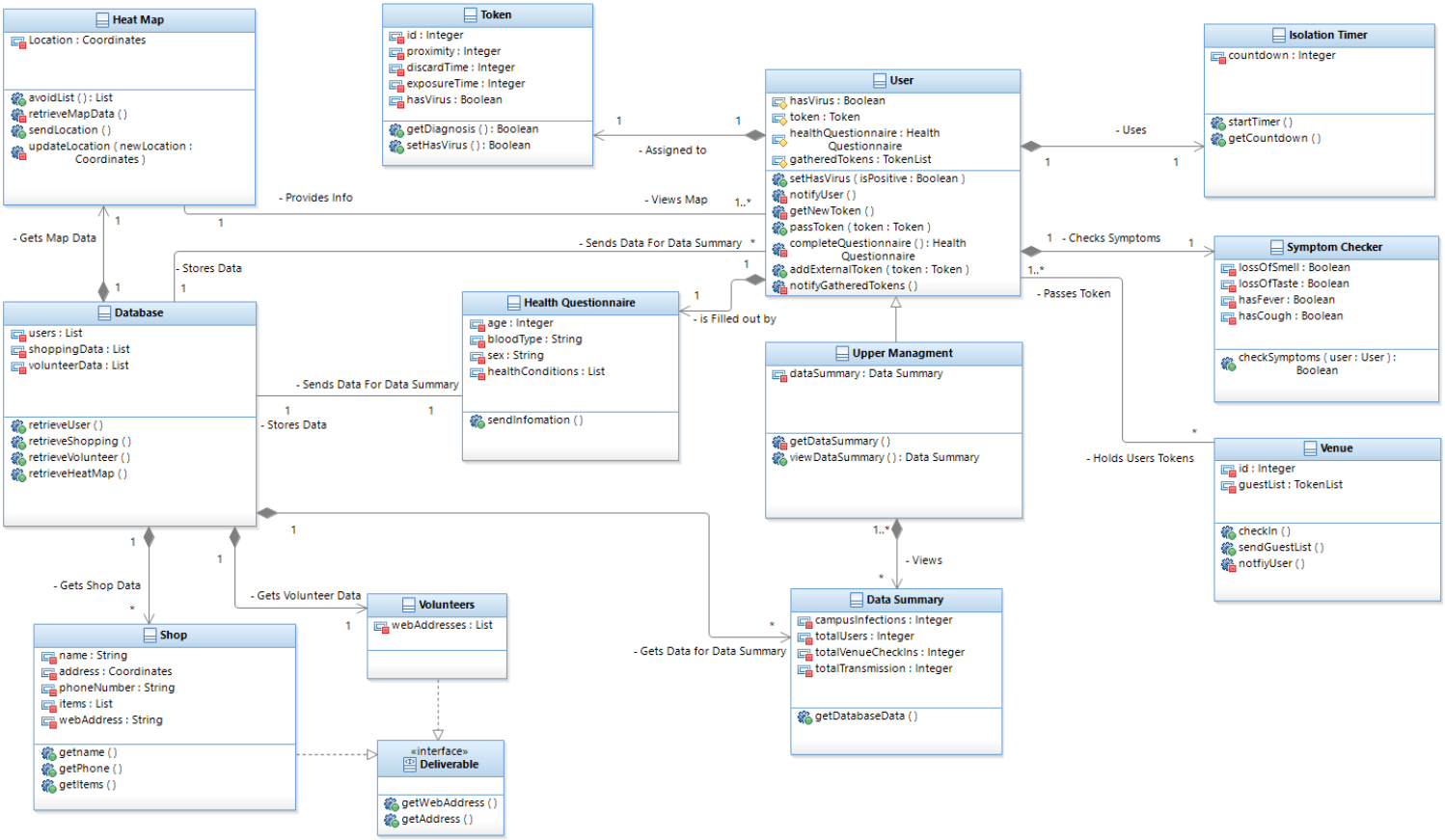


8 Structural Analysis

8.1 Class Diagram

The class diagram (Figure 6) models the classes of the app and their relationships with one another. Classes store and manage information and have methods which can be used to access and use this information.

Figure 6 – Class Diagram

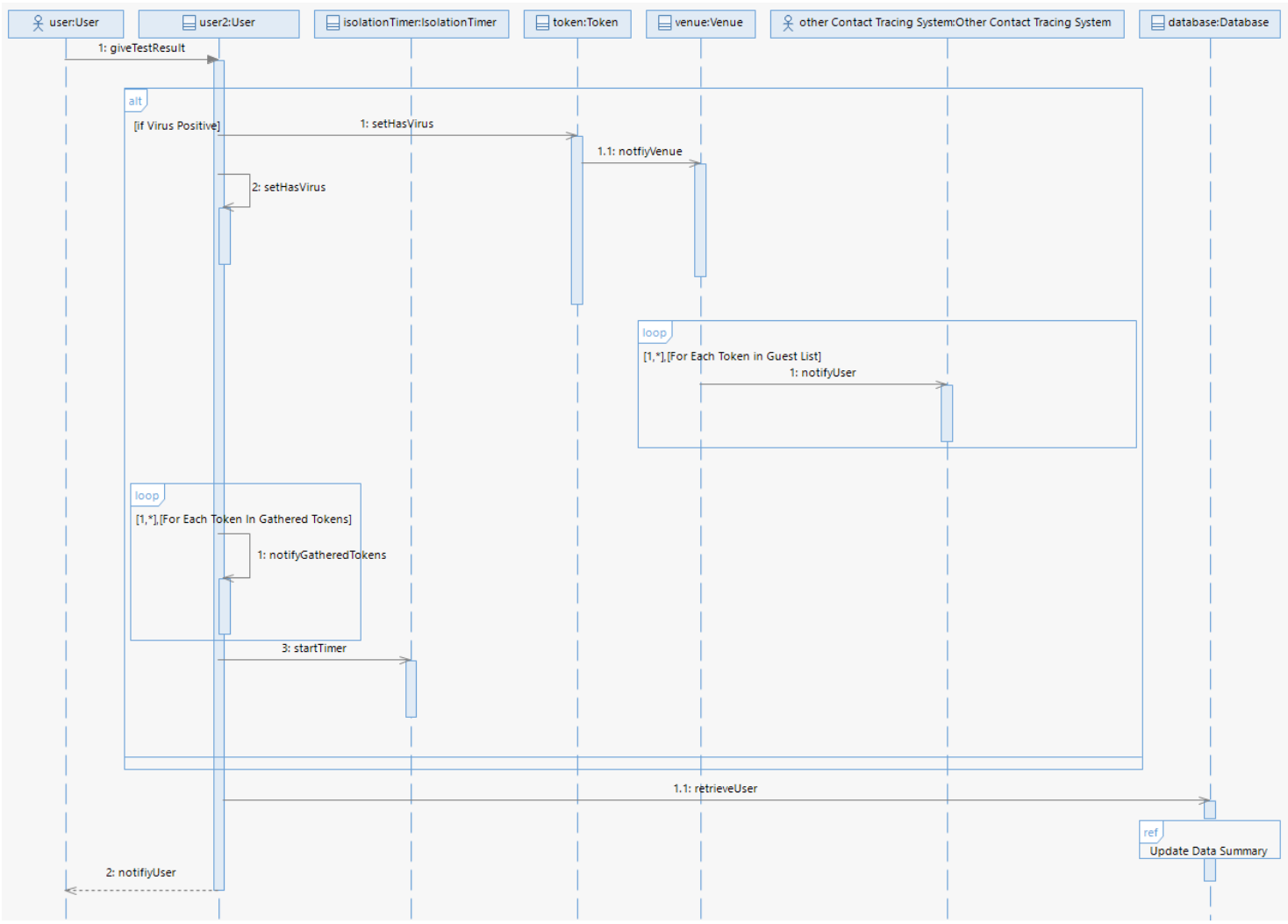


9 Behaviour Analysis

9.1 Sequence Diagram

The purpose of sequence diagrams is to display the objects that are utilised within use cases and the messages that are passed between them. They are a type of interaction diagram. Figure 7 is a sequence diagram detailing how user test results are processed.

Figure 7 - Sequence Diagram

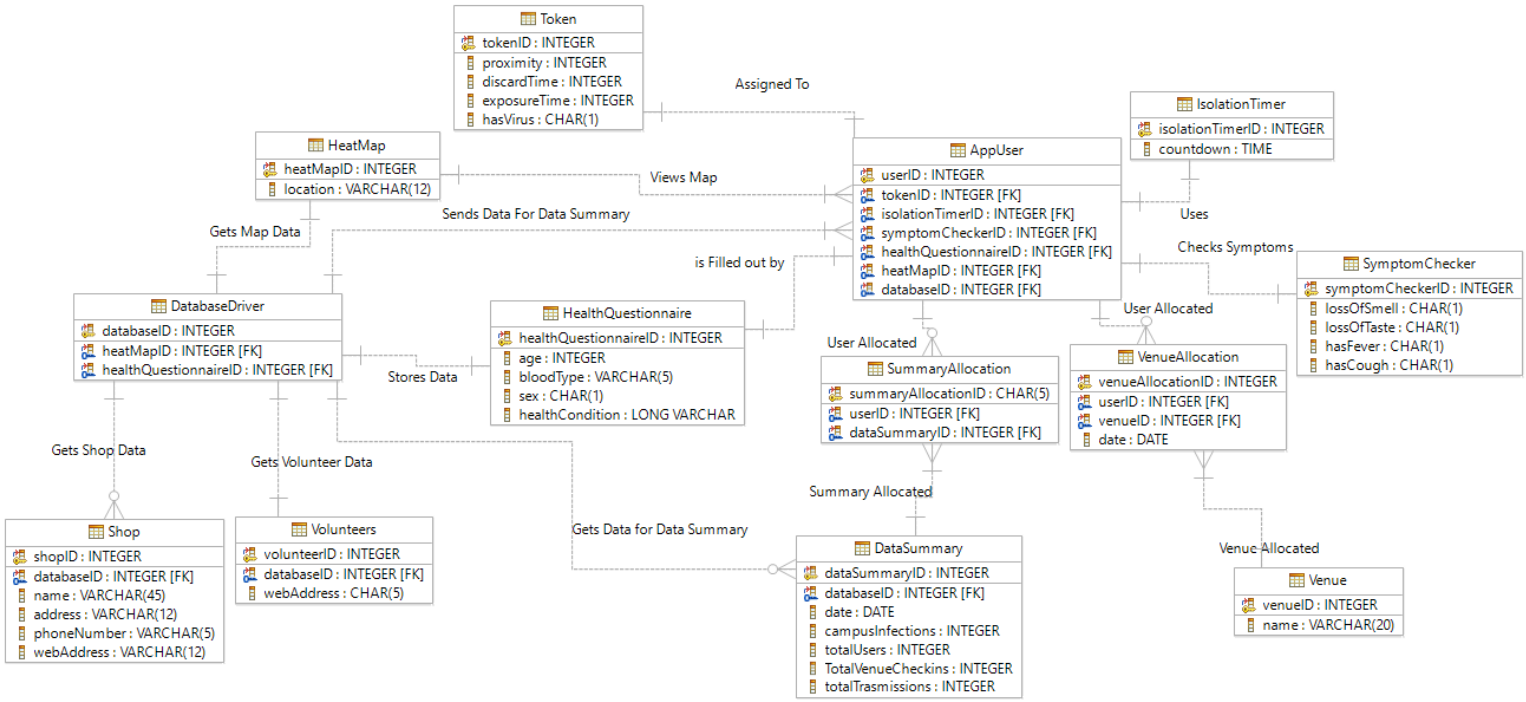


10 Database Design

10.1 Entity Relationship Diagram

While Entity Relationship Diagrams look similar to class diagrams, they are used in database design and debugging. The diagram (Figure 8) shows the relationships of the entities stored within the database.

Figure 8 - Entity Relationship Diagram



10.2 Sample SQL

The diagrams below show a representation of the Structured Query Language (SQL) that would communicate with the database(s) associated with the groups contract tracing app. Here, the data types and variables can be seen, as well as some SQL commands. These commands can add, delete, combine and pull certain values from databases themselves. The two example SQL statements shown depict: pulling the alert data (that a specific user has received) from the table (Figure 9) and Figure 10 depicts pulling the locations/campuses that a user has been to on a specific date from the table.

Figure 9 – Alerts SQL

```
SELECT APP.APPUSER.USERID, APP.TOKEN.HASVIRUS
FROM APP.APPUSER, APP.TOKEN
WHERE APP.APPUSER.TOKENID = APP.TOKEN.TOKENID AND APP.TOKEN.HASVIRUS = 'T'
;
```

Figure 10 – Location on date SQL

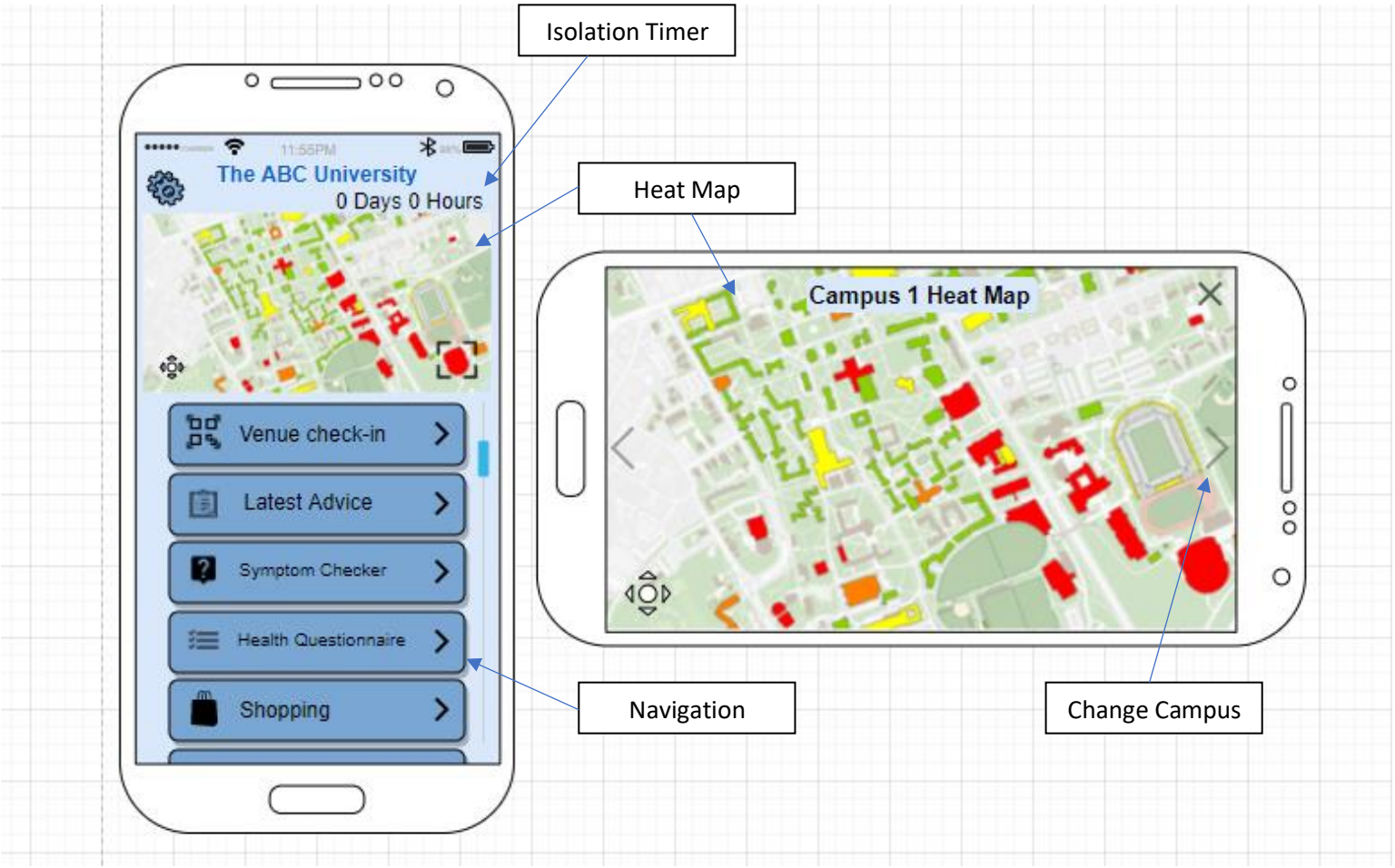
```
SELECT APP.APPUSER.USERID, APP.VENUE.NAME, APP.VENUEALLOCATION.DATE
  FROM APP.APPUSER, APP.VENUE, APP.VENUEALLOCATION
  WHERE APP.APPUSER.USERID = 2 AND APP.VENUEALLOCATION.DATE = '2021-02-03'
    AND APP.APPUSER.USERID = APP.VENUEALLOCATION.USERID
    AND APP.VENUE.VENUEID = APP.VENUEALLOCATION.VENUEID
;
```

11 User Interface (UI) Design

11.1 UI Prototype

The UI Prototype (Figure 11) is a design for how the user interface will look and feel. The design is simple in nature to help with ease of use. A minimised heat map is contained in the top half of the screen with the isolation timer (if activated). The bottom half contains the scrollable navigation buttons which allow the user to access the rest of the system. Once pressed, the heat map will expand to full screen to allow easier manipulation of the map and enable the user to view individual campuses.

Figure 11 – Prototype UI Design



12 Individual Reflection

1) Introduction & Summary - Introduction & summary completed by **Matthew**, **Harry** and **Jak**. Executive summary written by **Harry**. Aim and initial Ideas and Review of contact tracing systems was written by **Jak**. The Team and Key Considerations was written by **Matthew**. Methodologies section written by **Matthew** and **Harry**.

2) Planning - **Jak, Matthew and Harry** worked on ideas for risks and feasibility. Final risk assessment was compiled together by **Harry**. The Gantt chart and project schedule and PERT Diagram was created by **Jak.**

3) Requirement Analysis - Requirement Capturing Method was written by **Matthew**. The Student Survey and description was created by **Harry**. The E-JAD Document was created by **Jak**. Jak, Matthew and Harry worked on ideas for the requirement list, the final list was compiled and written by **Jak**.

4) Functional Analysis - **Jak, Matthew and Harry** worked on ideas for the use cases required. Use case diagram was compiled by **Jak**. Use case descriptions were written by **Harry** and **Matthew**.

5) Structural Analysis - **Jak, Matthew and Harry** worked on constructing the class diagram. Class diagram was compiled by **Jak**. Description was written by **Matthew**.

6) Behavioural Analysis - **Jak, Matthew and Harry** worked on constructing the sequence diagram. The sequence diagram was compiled by **Jak**. Description was written by **Matthew**.

7) Database Design - **Jak, Matthew and Harry** worked together on constructing the Entity Relationship Diagram. The Entity Relationship Diagram was constructed by **Jak**. SQL examples were created by **Jak**. Description was written by **Harry**.

8) User-Interface Design - **Jak, Matthew and Harry** worked on constructing the design. Design was created by **Harry** and **Jak**.

9) References - Compiled by **Matthew**, **Harry** and **Jak**.

Matthew

I was the leader of the team and did not assign particular group roles as I believed the report would benefit from everybody being involved in the creation of each section. **Harry**, **Jak** and I worked closely together to complete this report with many proposal elements being fulfilled during regular meetings. Not all work was completed during team meetings, so work was also assigned by me to be finished by the next scheduled session. Strong contributions were made by **Harry**, **Jak** and I. **Harry** and **Jak** had a positive attitude throughout and were willing to work towards the highest standards and attend regular meetings. **Pirajan** was engaged at the beginning in the creation of the report but this changed early on. **Pirajan** would occasionally attend meetings but would not complete work assigned to him to an acceptable standard after having multiple extensions. His engagement in meetings was also poor, providing little to the overall discussion. His contributions to the report are minimal, no work in this report was written by him, just a general outline for The Team and Key Considerations sections which I had to rewrite and complete for the report.

In terms of my own contribution, I was involved in all sections of the report, as seen in the breakdown above. I also scheduled the meetings and wrote the minutes for them.

Jak

I made contributions to all areas of the report, I wrote two sections of the introduction (the aims and review section), created the Gantt chart and PERT diagram for the project schedule and came up with several ideas for the functional and non-functional requirements, as well as I wrote up the final requirements list. I also created the E-JAD post session write up and session minutes for the appendix. For functional analysis I came up with several ideas for the different use cases and then created the use case diagram on RSA. For structural Analysis I came up with several ideas for classes, variables and methods for the class diagram. I also created the class diagram on RSA. For behavioural analysis I came up with several ideas for the different methods that would interact with each other and created the diagram in RSA. For database design I came with ideas for tables and created the diagram on RSA. Regarding the User-Interface Design I gave input into the design while Harry created it on teams. Later I added some of my own designs to Harry's diagram. I helped compile the reference list and proofread them at the end of the assignment.

Regarding my own work and team contribution I feel I have contributed ample amounts to the team effort and have always been on time in meetings and active in the team WhatsApp group giving ideas and help if any member was in need.

Harry

In the report I made contributions to all sections. Within the introduction I completed a methodologies analysis description that led to the decision of our software development life cycle. I also wrote the executive summary that documented what the report would include. I put forward ideas towards the feasibility and risk analysis, before drawing up the table to document the results of our discussion. I then created the student survey within requirement analysis, along with a sample email, before plotting test data in a graph to visualise survey findings within the report. For functional, structural and behavioural analysis, I attended all meetings and gave consistent contributions, to allow us to draw up the necessary diagrams on Jaks shared screen. After giving similar contributions within the SQL diagrams, I wrote a description on the database commands within the SQL diagram. Within User Interface design I made the initial core templates, before Jak finished them off and added detail. I helped compile references and was punctual to all meetings.

13 References

Almagor, J. & Picascia, S., 2020. *Exploring the effectiveness of a COVID-19 contact tracing app using an agent-based model*. Scientific Reports, 10(1).

AltexSoft, 2021. *Extreme Programming: Values, Principles, and Practices.* [Online]
Available at: https://www.altexsoft.com/blog/business/extreme-programming-values-principles-and-practices/
[Accessed 27 January 2021].

Beck, K., et al., 2001. *Manifesto for Agile Software Development* [online]. Agile Alliance. Available at: http://agilemanifesto.org/
[Accessed 27 January 2021].

Beck, K. and Andres, C., 2004. *Extreme Programming Explained: Embrace Change* [online]. 2nd ed. New Jersey: Pearson Education. Available at: https://ptgmedia.pearsoncmg.com/images/9780321278654/samplepages/9780321278654.pdf
[Accessed 27 January 2021].

Bozhenko, K., 2017. *Why Having Both IOS And Android App Versions Is So Important.* [Online]
Available at: https://octodev.net/why-having-both-ios-and-android-app-versions-is-so-important/
[Accessed 17 March 2021].

Braithwaite, K. & Joyce, T., 2005. XP Expanded: Distributed Extreme Programming. In: *Extreme Programming and Agile Processes in Software Engineering.* Berlin: Springer Berlin Heidelberg, pp. 180-188. Available at: http://dx.doi.org/10.1007/11499053_21
[Accessed 27 January 2021].

Burgess, M., 2020. *Everything you need to know about the new NHS contact tracing app.* [Online]
Available at: https://www.wired.co.uk/article/nhs-covid-19-tracking-app-contact-tracing
[Accessed 17 March 2021].

Carmel, E., George, J. F. & Nunamaker Jr, J. F., 1992. SUPPORTING JOINT APPLICATION DEVELOPMENT (JAD) WITH ELECTRONIC MEETING SYSTEMS: A FIELD STUDY. In: *Proceedings of the Thirteenth International Conference on Information Systems.* Dallas: University of Minnesota, p. 223–232. Available at: https://core.ac.uk/download/pdf/301364018.pdf
[Accessed 11 Febuary 2021].

Cochrane, 2020. *Digital contact tracing technologies in epidemics: a rapid review*. [Online] Available at: https://www.cochrane.org/news/featured-review-digital-contact-tracing-technologies-epidemics-rapid-review
[Accessed 17 March 2021].

Cohn, M., 2007. *Differences Between Scrum and Extreme Programming.* [Online]
Available at: https://www.mountaingoatsoftware.com/blog/differences-between-scrum-and-extreme-programming
[Accessed 27 January 2021].

Dennis, A., et al., 2015. *Systems Analysis & Design: An Object-Oriented Approach with UML*. 5th ed. Hoboken, New Jersey: John Wiley & Sons.

Downey, A., 2020. *Total cost of NHS contact-tracing app set to top £35 million.* [Online]

Available at: https://www.digitalhealth.net/2020/09/total-cost-of-nhs-contact-tracing-app-set-to-top-35-million/
[Accessed 28 March 2021].

Wang, D. & Evans, J. A., 2019. *Research: When Small Teams Are Better Than Big Ones.* [Online]
Available at: https://hbr.org/2019/02/research-when-small-teams-are-better-than-big-ones
[Accessed 17 March 2021].

Harvey, C., 2018. *Cloud Storage Pricing of 2021: Compare Cloud Storage Providers.* [Online]
Available at: https://www.enterprisestorageforum.com/cloud/cloud-storage-pricing/
[Accessed 17 March 2021].

Kesswani, N., Lyu, H. & Zhang, Z., 2018. Analyzing Android App Privacy With GP-PP Model. *IEEE Access,* Volume 6, pp. 39541-39546. Available at: https://ieeexplore.ieee.org/document/8395277
[Accessed 28 March 2021].

Kucharski, A. J. et al., 2020. *Effectiveness of isolation, testing, contact tracing, and physical distancing on reducing transmission of SARS-CoV-2 in different settings: a mathematical modelling study*. The Lancet Infectious Diseases, 20(10), pp. 1151-1160.

McCool, M., Robison, A. D. & Reinders, J., 2012. Background. In: T. Green, ed. *Structured Parallel Programming.* Boston: Morgan Kaufmann, pp. xix-xxii. Available at:
https://www.sciencedirect.com/topics/computer-science/single-instruction-single-data
[Accessed 11 Febuary 11].

Norman, T., 2014, 5 - Electronics Elements (High-Level Discussion). In: T. NORMAN, ed., Integrated Security Systems Design (Second Edition). Boston: Butterworth-Heinemann, 2014, pp. 49-55.[Online]
Available at: https://www.sciencedirect.com/science/article/pii/B978012800022900005X
[Accessed 17 March 2021].

Oosterhof, R., 2016. *What scrum, kanban and XP have in common.* [Online]
Available at: https://medium.com/@richardoosterhof/what-scrum-kanban-and-xp-have-in-common-99353d20a8b6
[Accessed 27 January 2021].

Ossmium, 2018. *The Essence Of Simplicity Or Psychological And Social Foundation Of The Mobile App Success.* [Online]
Available at: https://medium.com/nyc-design/the-essence-of-simplicity-or-psychological-and-social-foundation-of-the-mobile-app-success-5b592dfcb640
[Accessed 17 March 2021].

Royce, W. W., 1970. Managing the Development of Large Software Systems. *Proceedings of IEEE WESCON,* pp. 1-9. Available at:
https://leadinganswers.typepad.com/leading_answers/files/original_waterfall_paper_winston_royce.pdf
[Accessed 17 March 2021].

Rumpe, B. & Scholz, P., 2003. Scaling the management of extreme programming projects. *Projects & Profits. Special Issue on Management of Extreme Programming Projects,* III(8), pp. 11-18. Available at: https://arxiv.org/ftp/arxiv/papers/1409/1409.6604.pdf
[Accessed 27 January 2021].

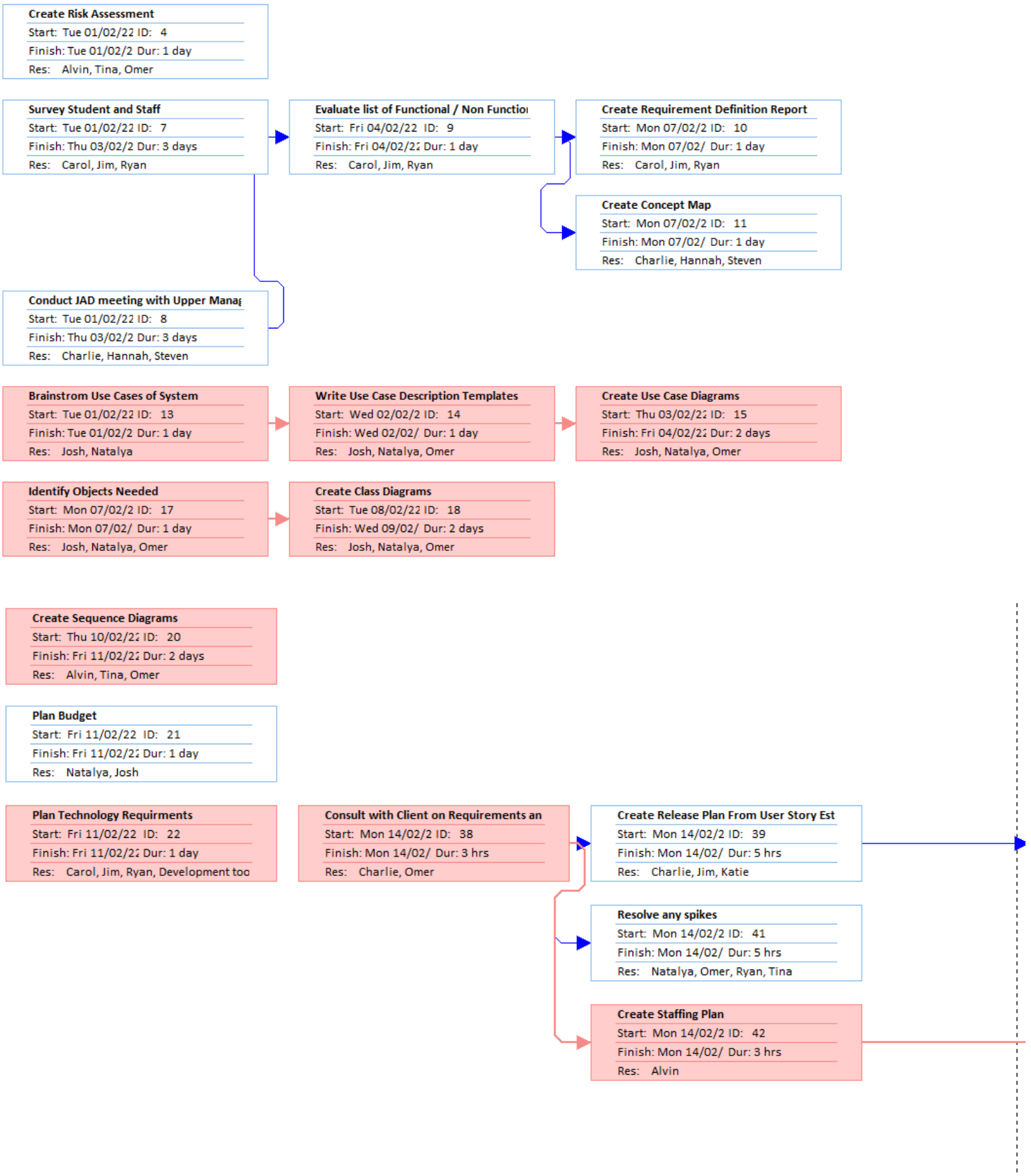
Wells, D., 1999. *When should Extreme Programming be Used?.* [Online]
Available at: http://www.extremeprogramming.org/when.html
[Accessed 27 January 2021].

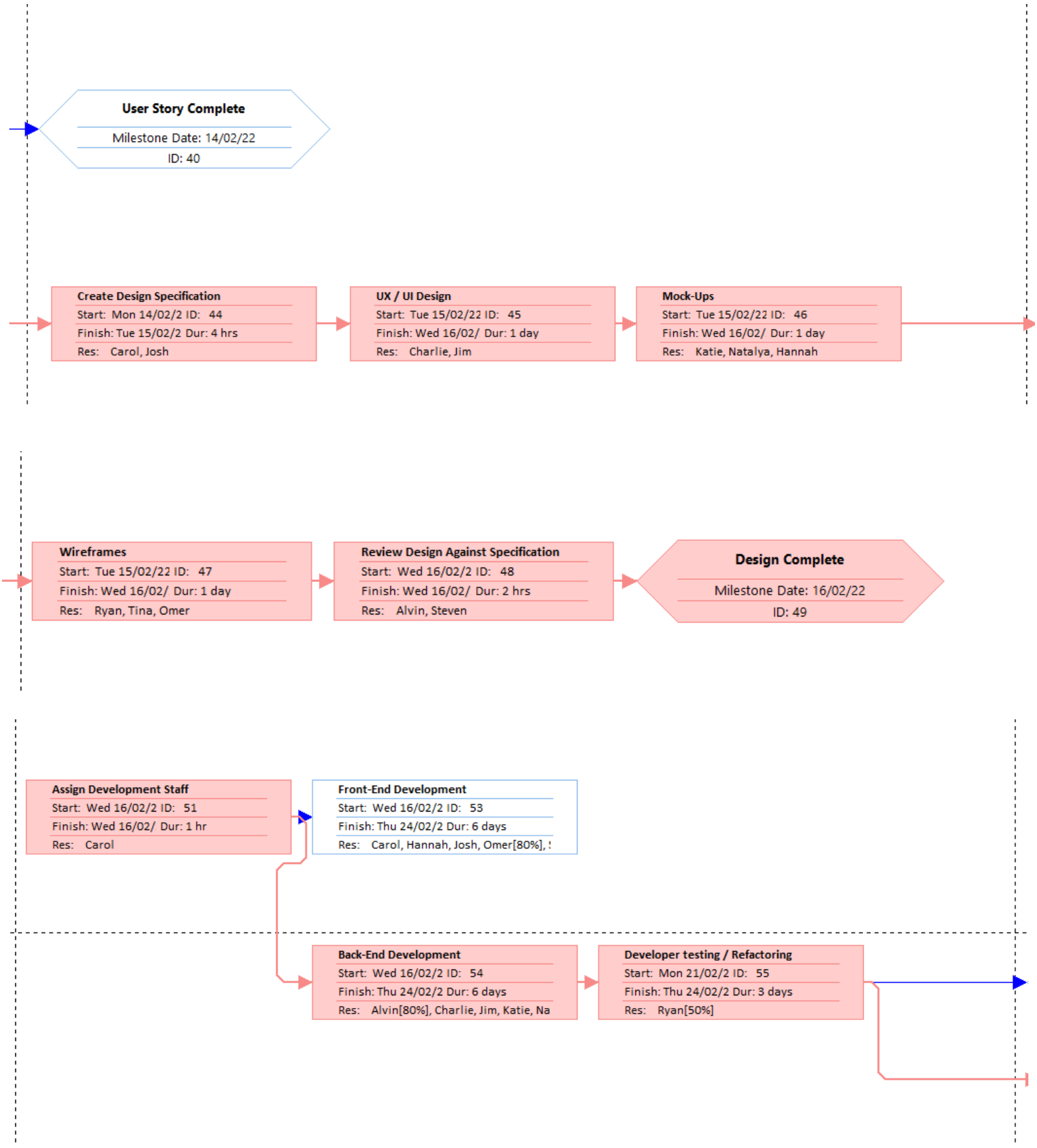
Wright, S., 2020. *Contact tracing apps for COVID-19.* [Online]
Available at: https://post.parliament.uk/contact-tracing-apps-for-covid-19/
[Accessed 17 January 2021].

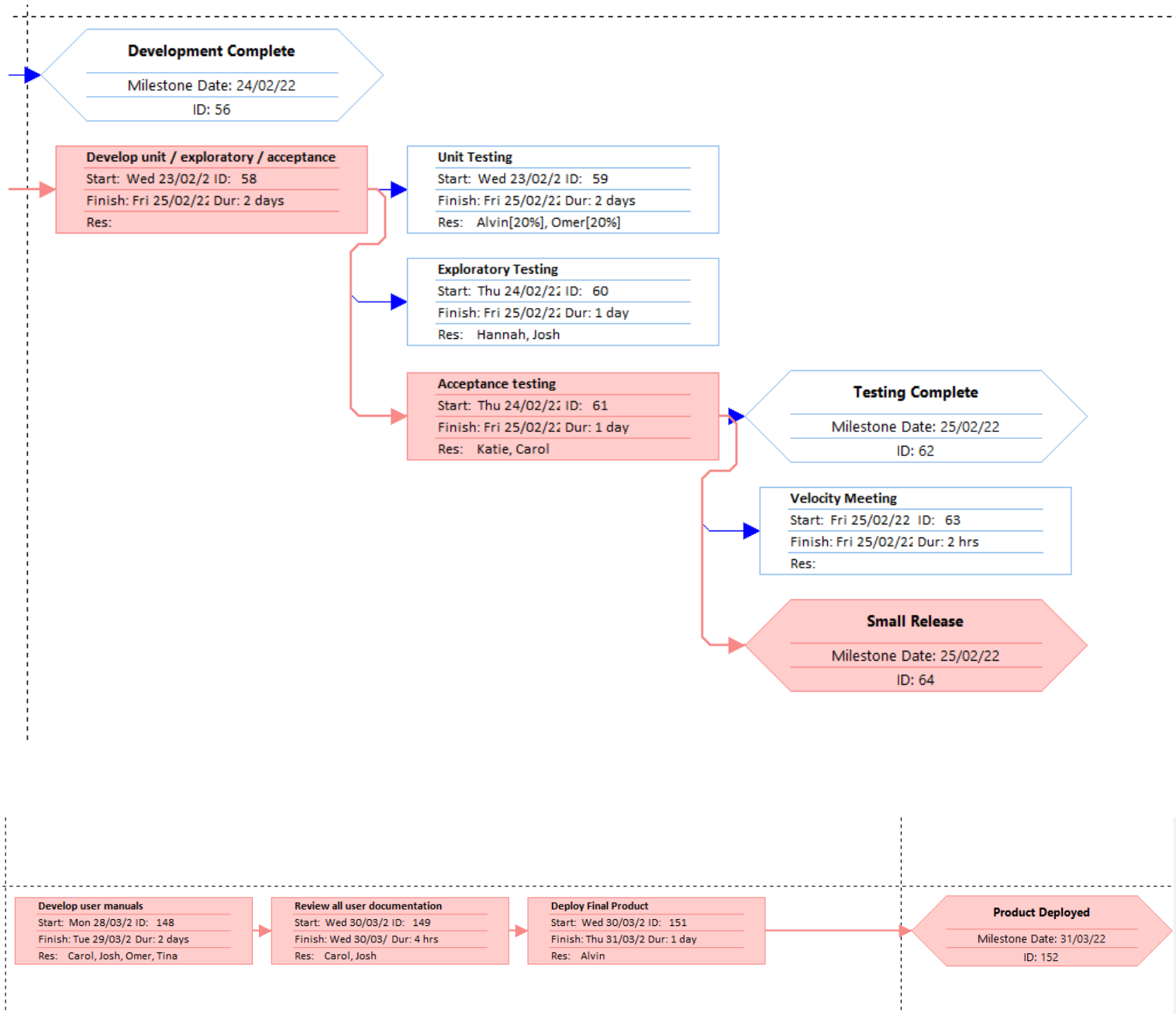
Yatco, M. C., 1999. *Joint Application Design/Development.* [Online]
Available at: http://www.umsl.edu/~sauterv/analysis/JAD.html
[Accessed 11 Febuary 11].

Appendix A – PERT Diagram

Figure 12 – PERT Diagram




User Story Complete
Milestone Date: 14/02/22
ID: 40
Create Design Specification
Start: Mon 14/02/2 ID: 44
Finish: Tue 15/02/2 Dur: 4 hrs
Res: Carol, Josh
UX / UI Design
Start: Tue 15/02/22 ID: 45
Finish: Wed 16/02/ Dur: 1 day
Res: Charlie, Jim
Mock-Ups
Start: Tue 15/02/22 ID: 46
Finish: Wed 16/02/ Dur: 1 day
Res: Katie, Natalya, Hannah
Wireframes
Start: Tue 15/02/22 ID: 47
Finish: Wed 16/02/ Dur: 1 day
Res: Ryan, Tina, Omer
Review Design Against Specification
Start: Wed 16/02/2 ID: 48
Finish: Wed 16/02/ Dur: 2 hrs
Res: Alvin, Steven
Design Complete
Milestone Date: 16/02/22
ID: 49
Assign Development Staff
Start: Wed 16/02/2 ID: 51
Finish: Wed 16/02/ Dur: 1 hr
Res: Carol
Front-End Development
Start: Wed 16/02/2 ID: 53
Finish: Thu 24/02/2 Dur: 6 days
Res: Carol, Hannah, Josh, Omer[80%],
Back-End Development
Start: Wed 16/02/2 ID: 54
Finish: Thu 24/02/2 Dur: 6 days
Res: Alvin[80%], Charlie, Jim, Katie, Na
Developer testing / Refactoring
Start: Mon 21/02/2 ID: 55
Finish: Thu 24/02/2 Dur: 3 days
Res: Ryan[50%]


Development Complete
Milestone Date: 24/02/22
ID: 56
Develop unit / exploratory / acceptance
Start: Wed 23/02/2 ID: 58
Finish: Fri 25/02/22 Dur: 2 days
Res:
Unit Testing
Start: Wed 23/02/2 ID: 59
Finish: Fri 25/02/22 Dur: 2 days
Res: Alvin[20%], Omer[20%]
Exploratory Testing
Start: Thu 24/02/22 ID: 60
Finish: Fri 25/02/22 Dur: 1 day
Res: Hannah, Josh
Acceptance testing
Start: Thu 24/02/22 ID: 61
Finish: Fri 25/02/22 Dur: 1 day
Res: Katie, Carol
Testing Complete
Milestone Date: 25/02/22
ID: 62
Velocity Meeting
Start: Fri 25/02/22 ID: 63
Finish: Fri 25/02/22 Dur: 2 hrs
Res:
Small Release
Milestone Date: 25/02/22
ID: 64

Develop user manuals
Start: Mon 28/03/2 ID: 148
Finish: Tue 29/03/2 Dur: 2 days
Res: Carol, Josh, Omer, Tina
Review all user documentation
Start: Wed 30/03/2 ID: 149
Finish: Wed 30/03/ Dur: 4 hrs
Res: Carol, Josh
Deploy Final Product
Start: Wed 30/03/2 ID: 151
Finish: Thu 31/03/2 Dur: 1 day
Res: Alvin
Product Deployed
Milestone Date: 31/03/22
ID: 152

Appendix B - Student Survey

Questionnaire

1. Would you like the app to give you more personalized help based on your specific health risks? Y/N

2. Would you like to be notified if there is a high possibility you have come into contact with a COVID positive individual? Y/N

3. Would you like the ability to update your health status daily, if you experience symptoms or receive positive test results? Y/N

4. Do you feel a heat map of high-risk areas would be helpful to avoid coming into contact with COVID-19 at University? Y/N

5. Are there any features that you would like to see on the app that would be helpful to avoid coming into contact with COVID at University? _____.

6. Would you like to have a list of restaurants and shops that deliver within the app in order to avoid leaving the house for food? Y/N

7. Would it be helpful to have a list of volunteers that have signed up to pick up and deliver food so that you do not have to? Y/N

8. Is there any other helpful information that you would like to see broadcast on the app? _____ .

9. Which operating system would you most likely use to run the application? iOS / Android / Both

10. Would you say that simplicity is a big priority of the application? Y/N

11. Would you like the app to provide real-time information and updates while connected to the internet? Y/N

12. Do you feel it would be helpful to have the system operational twenty-four hours of the day? Y/N

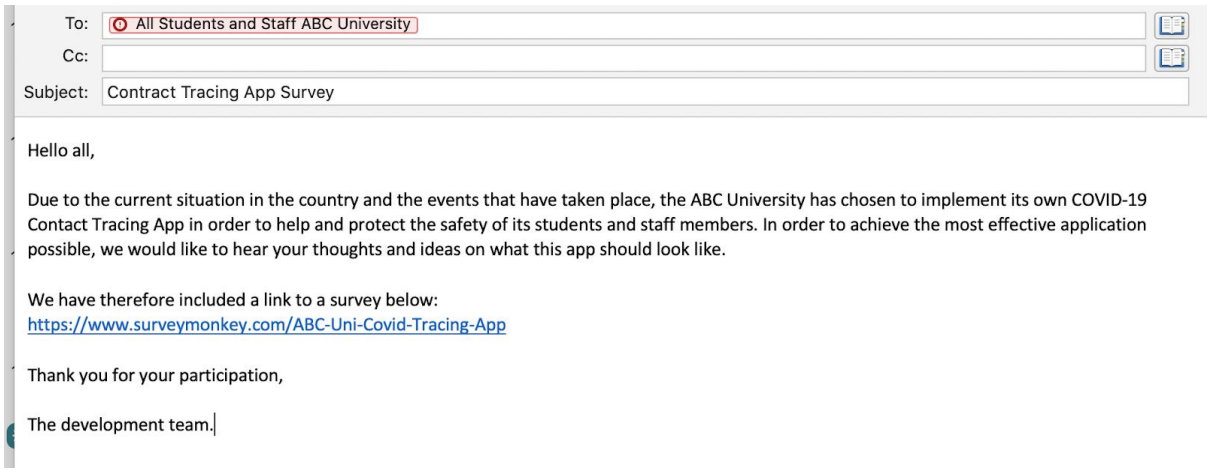
13. Would you like all data on the app to be kept anonymised to others? Y/N

14. Do you think that it would be beneficial for the student union to have an involvement in the development of the app to represent student ideas? Y/N

15. Are there any other student bodies that you feel could benefit the production of the app with their input? _____ .

Email and Results

Figure 13 - Survey Email to all staff and students


To: All Students and Staff ABC University

Cc:

Subject: Contract Tracing App Survey

Hello all,

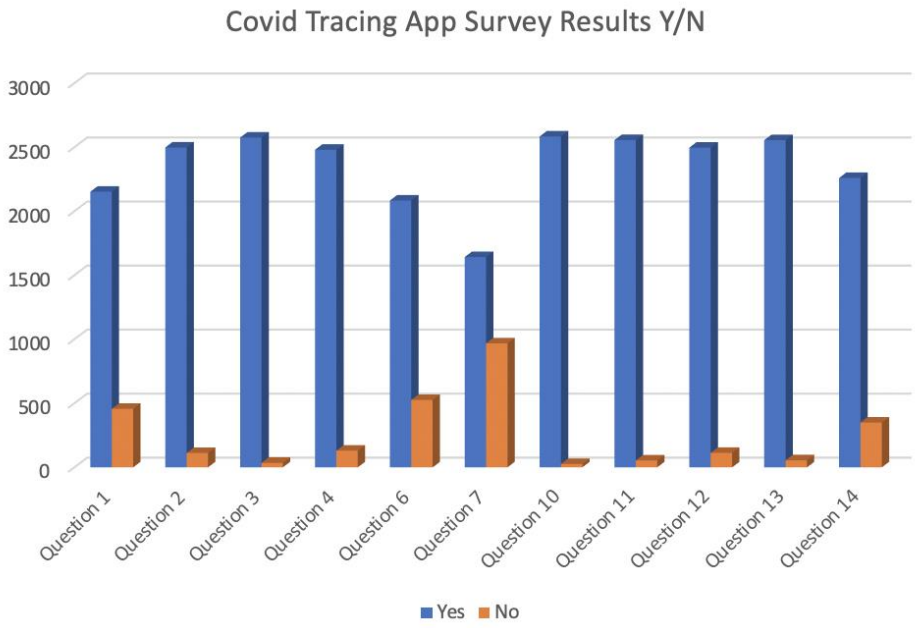
Due to the current situation in the country and the events that have taken place, the ABC University has chosen to implement its own COVID-19 Contact Tracing App in order to help and protect the safety of its students and staff members. In order to achieve the most effective application possible, we would like to hear your thoughts and ideas on what this app should look like.

We have therefore included a link to a survey below:
https://www.surveymonkey.com/ABC-Uni-Covid-Tracing-App

Thank you for your participation,

The development team.

Figure 14 – Survey Results



The figures plotted in Figure 14 match a linear increase of numbers drawn up from a sample test within those working on the application. The survey findings would be shown in a chart such as this and would heavily impact the features implemented in the methods and procedures to come. Figure 14 models the idea that if the ABC University holds about 26,000 students and 1000 staff members in all, so if 10 percent of the population carried out the survey, these would be the figures apparent.

Some of the more significant responses for the written-answer questions can be seen as follows:

5) I think that one feature that could be extremely helpful is one where you can see a large list of, on average, the highest known areas of Covid-transferring instances. This would be great because it would give me an idea of where to avoid going and help me boost the chances of not getting COVID-19.

8) The delivery idea is brilliant! It means I can feel more secure about getting some of my meals in these scary Coronavirus times. It would be good to have this easily accessible straight from the app's page. Also, maybe some links to the student union schemes to tie into the community where we are at?

15) I believe that some relevant, verified student societies might want to get on board with developing the app, like maybe the Computer Science society or the Health and medical ones.

These suggestions will certainly be considered by the team and many features given by those answering the survey could be seen in the final solution, such as the idea to incorporate a daily list of places to avoid for the students and staff at the university.

Appendix C – E-JAD

E-JAD Memo

Post Session Report

Session Members:
Edward Peck, Vice-Chancellor and President
Sharon Huttly, Deputy Vice-Chancellor - Academic Development and Performance
Nigel Wright, Deputy Vice-Chancellor - Research and Enterprise
Steve Denton, Chief Operating Officer and Registrar
Cillian Ryan, Pro Vice-Chancellor - International
Jane McNeil, Interim Pro Vice-Chancellor - Education
James Lacey, Director of Finance
Emma Leech, Director of Marketing and Communications
Guy Mallison, Director of Business Development and Analytics

Facilitator, Emily Gladwell

Purpose of JAD Session:
- Discuss possible features for COVID contact tracing system.
- Discuss requirements that the COVID contact tracing system must adhere to.
- Discuss results of student / staff survey.

Summary of Session:
Functional:
- The heat map of all high infection areas must be replaced by a heat map of high traffic areas to help keep all personal information private and to avoid leaking sensitive health information.
- A health questionnaire was discussed that assesses the user's risk to COVID-19.
- The system must notify the user if they have possibly been in contact with someone who is COVID positive.
- The system must allow the users to input their health daily (reporting on symptoms they may have).
- The system must hold all the user's information in a database.
- The system must hold information on local shops and restaurants that deliver.
- The system must hold a list of local volunteers that have signed up to help deliver food to those who are isolating.
- The system must hold the heat map data.

Non-Functional:

- The system must run on android and iOS devices.
- The system must be able to be adopted for any pandemic scenario.
- The information provided from the app must be in real time when connected to the internet.
- The system must operate 24/7.
- The system is focused on the university ecosystem (those related to university - students and staff).
- The system must be in compliance with the Data Protection Act 2018.
- It must be university policy to download and regularly use the application when on university property to maximise the systems efficacy.
- The student union must be involved in the making of the app to help represent

student ideas.

Open items:

No open items.

Detailed notes:

See JAD Interview minutes.

JAD Interview minutes

08/02/2021 - 10:32

Duration: 1 hour.

Participants:

Emily Gladwell - Facilitator
Edward Peck - Participant
Sharon Huttly - Participant
Nigel Wright - Participant
Steve Denton - Participant
Cillian Ryan - Participant
Jane McNeil - Participant
James Lacey - Participant
Emma Leech - Participant
Guy Mallison - Participant

Minutes

Participants names were anonymised as per E-JAD principles. Names listed as participants 1 to 9.

Question asked:

- Which features found from surveying the public do you wish to implement, change or lose from the system?
    - Participant 1 brought forth possible privacy issues regarding the heat map of infections idea.
    - Participant 3 suggest that the heat map be changed to a map of high traffic areas instead to help sway people from traveling there and reduce the traffic. Everybody agreed.
    - All agreed that the system should run on both iOS and android devices to cover a broad user base of staff and students.
    - All agreed that the system must operate 24/7 as it must cover those staying in hall and university accommodations.
    - All agreed that the system should hold information such as local shops and restaurants that deliver food during the pandemic.
- Which features do you wish to include in the system?
    - Participant 9 suggests that the app can be retro fitted and fit for purpose with other pandemic level events.

    - Participant 2 suggests a health questionnaire is used to assess the user's risk to covid and supply recommended information based on this information.
    - Participant 4 brought the idea that the users should have a way to report their symptoms in case they were unsure if their current symptoms pointed to them being infected. Lacey insisted that this feature be a daily occurrence to catch as many possible cases as possible, all were in agreement.
- What must the system do?
    - All agreed that the system should work similarly to the NHS contact tracing system and should notify the user upon coming in contact to a positive covid case and suggest social isolation and testing.
    - Leech stated that the system must be in compliance with the Data Protection Act to legally operate.
    - Participant 5 and Participant 7 came to the agreement that it would be important for the students (most potential users) that the student union be involved in the creation and changes after launch of the system to best represent the needs of the students with the bonus of accruing more ideas for possible features.
- What information should the system present to you and the users?
    - All agreed that the system should provide upper management a daily summary of its findings, including total infections and rate of infections per day to help them make decisions on the future of the university.
    - Participant 7 noted that he liked the idea of presenting the users with a list of local volunteer websites that may be used to help deliver food while the user is isolating or that the user may sign up to help others.
- What information does the system require to operate?

    - Participant 6 suggested that while amid the current pandemic a university policy should be put in place that required all staff and students to use the system while on university grounds to maximise its effect.

Appendix D – Use Case Descriptions

These are the use case description; they describe the different use cases an actor utilises when interacting with the contact tracing system.

Use Case Name: Check traffic on heat map	ID: 1	Importance Level: medium
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – wants to find out where the most foot traffic is around the University at that time to avoid those areas.		
Description: This use case describes how the heat map is checked by the user.		
Trigger: No specific trigger – in operation constantly. Type: Temporal		
Relationships: Association: System Include: Extend: Generalisation:		
Normal Flow of Events: 1. User navigates to Heat Map section of app through menu. 2. App displays Heat Map with relevant location of the user. 3. App confirms whether user has moved location and updates the heat map.		
Sub Flows: S-1:		
Alternate/Exceptional Flows:		

Use Case Name: Check top places to avoid leader board	ID: 2	Importance Level: medium
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User - wants to find out where the top places to avoid around the University are (on that day) to avoid those areas.		
Description: This use case describes how the top places to avoid leader board is checked by the user.		
Trigger: No specific trigger – in operation constantly. Type: Temporal		
Relationships: Association: System Include: Extend: Generalisation:		
Normal Flow of Events: 1. User navigates to leader board section of app through menu. 2. App displays leader board with relevant location of the user.		

Sub Flows: S-1:
Alternate/Exceptional Flows:

Use Case Name: Venue check in	ID: 3	Importance Level: high
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – wants to check the venue code into their app as they enter, so that in future they can be notified if there was a breach in Covid, in that place, within a certain period.		
Description: This use case describes how a user checks in to a venue upon entering.		
Trigger: User entering a venue Type: External		
Relationships: Association: System Include: Extend: Generalisation:		
Normal Flow of Events: 1. User navigates to venue check-in section of app through menu. 2. App allows user to enter venue ID. 3. User footprint is assigned to the venue. 4. Data is stored in database.		
Sub Flows: S-1:		
Alternate/Exceptional Flows:		

Use Case Name: Inform system of intent to isolate	ID: 4	Importance Level: high
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – wants to inform the system that they have received a positive test result / are experiencing symptoms and that they intend on isolating.		
Description: This use case describes how the user informs the system that they are going to isolate.		
Trigger: User decides to isolate Type: External		
Relationships: Association: System Include: Extend: Generalisation:		
Normal Flow of Events: 1. User navigates to Isolation section of app through menu.		

2. User inputs into the app that they wish to start isolating. 3. App confirms that the user wishes to isolate. 4. Isolation timer starts. 5. Isolation timer is outputted to the user in real time.
Sub Flows: S-1:
Alternate/Exceptional Flows:

Use Case Name: Daily symptom checker	ID: 5	Importance Level: high
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – wants to inform the system of their daily symptoms (whether they have changed and that they are now eligible to take a test or start isolating).		
Description: This use case describes how a user enters the new, daily information on whether they have gained symptoms in the last twenty-four hours.		
Trigger: A new day Type: Temporal		
Relationships: Association: System Include: Extend: Generalisation:		
Normal Flow of Events: 1. App asks the user if their symptoms have changed within the last 24 hours. 2. If user answers 'yes', data is asked to update their symptoms with the relevant changes, these are then stored on the system. 3. The system checks if the symptoms are significant enough to warrant isolation, and if so, the isolation timer starts.		
Sub Flows: S-1:		
Alternate/Exceptional Flows: 2b If user answers 'no', no changes are made to the system.		

Use Case Name: Flag that the user is Covid positive / experiencing symptoms	ID: 6	Importance Level: high
Primary Actor: System	Use Case Type: Detail, Essential	
Stakeholders and Interests: System – wants to update its records on the user if their Covid situation has changed; so it can then perform other operations for other possibly affected users.		
Description: This use case describes how the system recognises and updates the fact that a user has changed from non-suffering to either a Covid-negative who's experiencing symptoms or Covid-positive.		

Trigger: The user is experiencing symptoms Type: Temporal
Relationships: Association: User Include: Extend: Generalisation:
Normal Flow of Events: 1. After a positive covid symptom has been registered through the symptom checker, the system moves to respond to and flag this information. 2. The data is updated in the private database. 3. The system checks the footprint of tokens to and from the user. 4. The system notifies all users who are in possession of the (now-isolating) individuals token.
Sub Flows: S-1:
Alternate/Exceptional Flows:

Use Case Name: Inquire if user Covid positive	ID: 7	Importance Level: high
Primary Actor: System	Use Case Type: Detail, Essential	
Stakeholders and Interests: System – wants to ask the user if they are a certain positive case, as a result of a reliable test.		
Description: This use case describes how the system enquires about whether the user is definitely positive.		
Trigger: User comes within close contact to another user Type: Temporal		
Relationships: Association: User Include: Extend: Generalisation:		
Normal Flow of Events: 1. The system checks if the user has documented having experienced recent and new symptoms. 2. If this is true, then after 2-3 days has elapsed (so the user has time to order and carry out a test), the system asks the user if their Covid-19 test results were positive or negative. 3. If positive, the system updates its records on that user and moves to notify users with shared tokens.		
Sub Flows: S-1:		
Alternate/Exceptional Flows: 3a If negative, the records are reset and the system notifies the user of the fact that they are free to move out of isolation.		

Use Case Name: Isolation countdown timer	ID: 8	Importance Level: low
Primary Actor: System	Use Case Type: Detail, Essential	
Stakeholders and Interests: System – wants to execute a countdown that will allow the user to see the amount of time they need to isolate for and how long they have left to isolate. User – wants to know the current progression of their isolation period.		
Description: This use case describes how the system creates a countdown timer for the user.		
Trigger: Started isolating Type: Temporal		
Relationships: Association: User Include: Extend: Generalisation:		
Normal Flow of Events: 1. The system uses data inputted in the symptom checker or the test result checker to reach its outcome. 2. The system checks if this data is sufficient to warrant isolation, and if so, the isolation timer starts.		
Sub Flows: S-1:		
Alternate/Exceptional Flows:		

Use Case Name: Notify user to self-isolate	ID: 9	Importance Level: high
Primary Actor: System	Use Case Type: Detail, Essential	
Stakeholders and Interests: System – wants to notify the user that they need to isolate as a result of close contact with a Covid case and possible contraction of the illness. User - wants to receive notification on whether they should isolate for their own health and the health of others' sake.		
Description: This use case describes how the user is quickly told once they have been in contact with a possible Covid case.		
Trigger: Close contact with Covid case. Type: Temporal		
Relationships: Association: User Include: Extend: Generalisation:		
Normal Flow of Events: 1. The system checks the footprint of tokens to and from the user. 2. The system notes all users who are in possession of the (now-isolating) individuals token. 3. These users are then notified that they have been in contact with a possible positive Coronavirus case and should now start isolating themselves while they wait to be tested		
Sub Flows:		

S-1:
Alternate/Exceptional Flows:

Use Case Name: Fill Out Health Questionnaire	ID: 10	Importance Level: High
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – Wants to fill out questionnaire. Upper Management – Wants users to identify the risk of Covid-19 on user health to raise awareness/protect them.		
Description: This use case describes how user can complete a general health questionnaire to help identify their risk of susceptibility to Covid-19.		
Trigger: User navigates to the health questionnaire section of app. Type: External		
Relationships: Association: User Include: Extend: Get NHS Health Information/Guidelines Generalisation:		
Normal Flow of Events: 6. User navigates to questionnaire section of app through menu. 7. App asks user to fill out or edit questionnaire. 8. If the user has not filled out a questionnaire before and wishes to i. Display questionnaire. 9. If user has already filled out a questionnaire and wants to edit it i. The S-1: Edit questionnaire sub flow is performed.		
Sub Flows: S-1: Edit questionnaire 1. Allow user to edit their old responses. 2. Summarise results of changed questionnaire.		
Alternate/Exceptional Flows: 2a The user does not wish to fill out questionnaire, questionnaire is not opened.		

Use Case Name: Get NHS Health Information/ Government and University policy	ID: 11	Importance Level: Medium
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – Wants to see the latest advice from the NHS/Government/University on Covid-19. University/Upper Management – Wants app users to follow NHS/Government/University advice/guidelines to protect university staff/students. NHS – Provides health information and services related to Covid-19 for people residing in the UK. Government – Provides guidelines and laws related to Covid-19 which they want the public to follow.		
Description: This use case describes how user can access advice from the		

NHS/Government/University and head to their external site for more information.

Trigger: User navigates to the Latest Advice section of the app.
Type: External

Relationships:

- Association: User
- Include: Stream Information into System
- Extend: Fill Out Health Questionnaire
- Generalisation:

Normal Flow of Events:

1. User navigates to Latest Advice section through menu.
2. App provides a summary of NHS/Government/University information/guidance.
3. If user wants to leave app to visit NHS website
   i. The S-1: Visit NHS website sub flow is performed.
4. If user wants to leave app to visit University website
   i. The S-2: Visit University website sub flow is performed.
5. If user wants to leave app to visit Government website
   i. The S-3: Visit the Government sub flow is performed.

Sub Flows:

S-1: Visit NHS website

1. Links to NHS website will be present.
2. If user selects link, they will be taken to external NHS website.

S-2: Visit the University website

1. Links to University website will be present.
2. If user selects link, they will be taken to external University website.

S-3: Visit the Government website

1. Links to Government website will be present.
2. If user selects link, they will be taken to external Government website.

Alternate/Exceptional Flows:

3a User does not wish to visit NHS website, user remains on the app.
4a User does not wish to visit University website, user remains on the app.
5a User does not wish to visit Government website, user remains on the app.

Use Case Name: Get Locations of Testing Sites/Book Test Online	ID: 12	Importance Level: High
Primary Actor: User	Use Case Type: Detail, Essential	

Stakeholders and Interests:
User: Wants to see where they can get a test and book one if necessary.
Upper Management: Wants users of the app to be aware of any campus/local testing areas and be able to book a test to determine user health.
Testing Station/Doctors: Wants to test for potential positive cases of Covid-19.

Description: This use case describes how a user can get information about nearby locations where they can get a test and book a test at one of these locations if necessary.

Trigger: User navigates to testing section of app.
Type: External

Relationships:

- Association: User
- Include: Stream Information into System

• Extend: • Generalisation:
Normal Flow of Events: 1. User navigates to testing section through menu. 2. App provides location of testing centres. 3. App confirms if user would like to book a test. i. The S-1: Visit gov.uk website sub flow is performed
Sub Flows: S-1: Visit gov.uk website 1. User will be taken to the UK Government website to book a test.
Alternate/Exceptional Flows: 3a User does not wish to book a test, they are not taken to the booking website.

Use Case Name: View Data Summary	ID: 13	Importance Level: High
Primary Actor: Upper Management	Use Case Type: Detail, Essential	
Stakeholders and Interests: Upper Management: Wants to see summary of anonymous data collected by the app so they can help guide the ABC University Covid-19 response.		
Description: This use case describes how a user who is part of Upper Management can view data collected by the app to help them make decisions.		
Trigger: Upper Management User accesses data section of app in settings. Type: External		
Relationships: • Association: Upper Management • Include: • Extend: • Generalisation:		
Normal Flow of Events: 1. User navigates to data section of app through settings. 2. App confirms user is of type Upper Management. 3. Displays summary of anonymised app data.		
Sub Flows:		
Alternate/Exceptional Flows: 2a User is not of type Upper Management, summary of data is not shown to the user.		

Use Case Name: Change User Settings	ID: 14	Importance Level: Medium
Primary Actor: User	Use Case Type: Detail, Essential	
Stakeholders and Interests: User – Wants to update app settings.		
Description: This use case describes how a user can view the settings for the app and personalise them.		
Trigger: User accesses Settings section of app. Type: External		
Relationships:		

- Association: User
- Include:
- Extend:
- Generalisation:

Normal Flow of Events:

1. User navigates to User Settings section of app through menu.
2. App displays settings such as Accessibility and Language Selection.
3. App confirms whether user changes made wish to be saved.
   If yes, changes are saved.
   If default, settings are set back to their default values.

Sub Flows:

Alternate/Exceptional Flows:

3a User does not wish to save changes, changes are not saved.

Use Case Name: Show Shops/Restaurants/Volunteers that Deliver	ID: 15	Importance Level: Low
Primary Actor: User	Use Case Type: Detail, Essential	

Stakeholders and Interests:
User: Wishes to see where they can get food from.
Shops/Restaurants: Want people to buy their goods.
Volunteers: Want to help their community.

Description: This use case describes how a user can view shops/restaurants/volunteers in their proximity that deliver items/food.

Trigger: User accessed Shops/Restaurants/Volunteers section of app.
Type: External

Relationships:

- Association: User
- Include: Stream Information into System
- Extend:
- Generalisation:

Normal Flow of Events:

1. User navigates to Shops/Restaurants section of app through menu.
2. App displays shops/restaurants that are within the proximity of the user.
3. App confirms whether user wishes to visit shop/restaurant website.
   i. The S-1: Visit shop/restaurant website sub flow is performed.
4. App confirms whether user wishes to visit volunteer's website.
   i. The S-2: Visit volunteer website sub flow is performed.

Sub Flows:

S-1: Visit shop/restaurant website

1. Links to displayed shops/restaurants will be present.
2. User will be taken to shop/restaurant website that they select.

S-2: Visit volunteer website

1. Links to volunteer website will be present.
2. User will be taken to volunteer website that they select.

Alternate/Exceptional Flows:

3a User does not wish to visit shop/restaurant website, user remains on the app.

3b User does not wish to visit volunteer website, user remains on the app.