

Exam scripts viewing booking system

MSc Computing Dissertation

School of Computer Science and Informatics,
Cardiff University

September 2020

by Emma Terry

Supervised by Helen R Phillips

Contents	Page
Abstract	3
Acknowledgement	4
1.Introduction	5
2. Aims and objectives	7
3. Background	8
4. Problem	14
5. Approach	17
6. Application of the chosen approach	18
7. Product	21
8. Implementation and technology	24
8.. Analysis	35
9.. Conclusion	39
10. Reflection	41
11. References	43
Appendix	46
i) Risk table	
ii) Functions	
iii) User stories	
iv) Class Diagrams	

Abstract

For my dissertation project I was initially tasked with the creation of an Exam scripts viewing booking system. However due to the unforeseen catastrophic global impact of COVID -19. the scope of this project was adapted to address the changing nature of assessments in academia.

The final realist of my proposal is the creation of an overarching assessment feedback booking system. The aim of the system is to provide a consistent point of reference for both students and staff to access and deploy assessment feedback. Assessment is designed to monitor the development of student learning and to provide feedback and support for such learning (Black and Wiliam, 1998).

The current procedures in place were at times ineffective and time consuming to administer. My system would aim to address these concerns, potentially relieving additional pressures from staff having to solves the logistics of organising sessions, but also hopefully provide an application that students will engage with and find beneficial to their learning process whilst at higher education.

My prototype allows a user friendly experience for students to book an appointment from the lecturers personal calendar. The lecturer will be in complete control in regards to the allocation of availability within their calendar. To comply with data protection it will also only show the student their own booking in the calendar.

It also allows lecturers to add the specify modules and assessments relevant to their needs, it is a program that will be easily maintainable for them to reuse in future cohorts. If the prototype is successful I believe it can be easily deployed across other schools within the University.

Acknowledgements

Thanks to:

- Professor Helen R Phillips for supervising this project and providing guidance and feedback
- My classmates for their advice and feedback

1.Introduction

Feedback is at the heart of student progress and future success, having a way to ascertain this feedback is vital. It shapes the students academic journey through university and shapes the roadmap to their future. Mann 2001 went as far as to argue that poor assessment and feedback experiences can actually negatively impact the students experiences of university leading to dissatisfaction and distress in their course.

The challenge is providing feedback in a meaningful and efficient way. My project will examine how technology can assist academic staff in positively shaping students futures by providing them with every opportunity available to them.

The client initially proposed the creation of an Exam scripts viewing booking system. However due to global events that followed the emergence of COVID -19. the scope of this project had to change and adapt just as the universities themselves had to to the uncertainty that gripped the world.

My project would now be an overarching assessment feedback booking system. With the university shift away from traditional closed book examinations towards open book online examinations and assessments. This meant the demands of the system grew, with additional functionality required. These were renegotiated with the client and a design created.

I chose to follow the agile method for implementation due to these circumstances as the client was not sure at the start of the development who requirements may be essential functionality.

This prototype was designed using various scaleable technologies such as java and spring-boot although some frameworks would need to be replaced if this system was deployed due to the project being a cost free prototype, this should not affect the overarching design. In fact by implementing paid services such as Microsoft exchange it would enhance the applications functionality.

The prototype although based along the format of some traditional booking systems, is able to incorporate functionality specific to what the client was looking to implement to university staff. It is elevated beyond a simple scheduling system. It will allow academic staff the resources to link to their personal calendars and view transcripts. This will provide a consistency across the school. Something the client stressed was keen among senior managers. There is also potential to deploy the system across the different schools within the University. This would allow staff and students a clear point of access when providing and receiving feedback.

2.Aims and Objectives

Aims:

1. Build and test a prototype application to meet the needs of academic staff and students in facilitating assessment feedback in Cardiff School of Computer science and informatics.

Objectives:

1. Research what current literature says in regards to feedback in higher education
2. Identify the unmet needs of students and staff at the school of computing in relation to feedback.
3. Produce functional and non functional requirements
4. Implement and test my prototype application
5. Evaluate, and reflect upon my project considering future developments.

3. Background

3. The aim of this section is to provide a better understanding and overview of the project and explain the purpose it will fulfil.

In 2018/2019 there was 2.4 million students at UK higher education institutions completing a wide range of degree programmes (briefing paper, 2000). One of the most important reasons for attending university is to advance your education and in turn future career prospects. Therefore achieving the best grade possible is paramount, receiving feedback is crucial to this, Crook et al (2011) confirms this in their article finding that the provision of feedback to be essential to student learning. However they also found it was a challenge to achieve. In this section I will explore how this feedback is delivered and the challenges students and staff may face.

3.1 National student survey

The National student survey is an annual survey of nearly half a million students from across the UK. It is undertaken by final year students with the aim to gather an honest opinion on their time in higher education. The area that the survey has constantly highlighted as an intense source of student dissatisfaction is assessment feedback Morris et al (2019) This is further supported by findings from Carless and Boud (2018). They found that students were not not satisfied with neither the quality of feedback or the assessment regime in general terms. It was proposed to enhance the satisfaction the process would need to be reorganised for both access and application to the curriculum. Grint (2008) described the dissatisfaction students felt as a complex and 'wicked' problem.

3.2 Previous methods of feedback at Cardiff University.

In this section I will detail the previous systems that were in place specifically for students receiving feedback at Cardiff University school of computer science and informatics. I have chosen to focus on this school as my dissertation is based on the purpose of addressing the provision of feedback at this school. These policies and methods of feedback may not be reflective of all other universities or even other schools within Cardiff University.

The previous method of feedback used by the school was based on the Academic feedback to taught students policy and guidance(2015). The aim of the policy is to improve student learning by setting out the minimal obligations staff should do to provide effective feedback on assessments. The policy is based upon principles that ultimately aim to improve student satisfaction and build a student and tutor relationship (Irons, 2008).

Cardiff Universities previous process was to provide cohort feedback in writing on learning central with an opportunity for students to view their scripts on a particular day. This day was selected by the school. This was based on a trial of second year students only. Second year students were used to test the effectiveness due to the point in the degree they had reached. It was felt they would receive the greatest benefits from the trial as they would still have a lot of exams to sit going into their final year. The decision was made to test a small sample first as staff had concerns they would not be able to manage the uptake of appointments if it was introduced straight away to the entire school. This was facilitated by the use of Doodle poll and proved largely ineffective due to the lack of cohesion within the schools systems. There were many logistical issues of arranging appropriate times between student timetables and coordination with all the academic staffs timetables. Coursework feedback was provided briefly with marks. Routine face to face feedback with all students was not provided. This process ran for about 4 years then rolling out to all students but only for Autumn examinations.

If students sought feedback from their own initiative by contacting staff this was sometimes facilitated. These methods proved largely ineffective there was often very little uptake. One explanation for this may be the limited capability of doodle poll. Appointments were usually scheduled quite far in advance prior to the actual viewing. Doodle poll had no form of reminder system in place. Students may have simply forgot they had booked the appointment in some instances. In my proof of concept I would look to address this by having calendar and email capabilities. There may also be issues due in part to lack of awareness, staff members were not obligated to participate and therefore students may

not have been aware. There was also a lot of 'no-shows' using this method of appointment booking. The system was also very disruptive to academic staff who may have students randomly turning up at their offices to request feedback, which may have been inconvenient to facilitate. Thus negatively impacting the students experience.

3.3 Current procedure

On the date 27th March 2020, the UK prime minister made the decision that due to Covid 19 all higher education institutions were to shut their doors to physical teaching due to the risk the virus transmission posed to staff and student safety. This was unprecedented and unexpected. Universities had very little option but to either close their doors or move to online, remote teaching with little to no time to prepare. McBrien (2009) concluded that "Rapid developments in technology have made distance education easy", however a article by Dhawan.S (2020) proposes that even though the technology is accessible many academic institutions world-wide had been reluctant to make the shift to remote online learning simply due to becoming habitual to these traditional methods. UK universities generally with a few exceptions such as the Open University did not have exiting protocols and infrastructure already in place to facilitate this huge adjustment. There was and still is a big concern from students about how this will impact the quality of education they will be receiving, and the effects on those students socially disadvantaged as discussed in Dhawan, S. (2020) One outcome of this change was to move all examinations to an online format such open book.

The proven and regulated academic structure that the university had been using for many years went through a rapid overhaul. it became physically impossible for students to sit in halls to undertake examinations, these all had to move to a remote online format. The university has decided that due to the shifting climate that moving forward these changes will continue to be implemented and expanded upon. In the forthcoming academic year 2020/21, wherever possible teaching is still to be done remotely with no current plans to revert back to traditional methods. The implications of this are far reaching to both staff and students alike. Within the school of computer science and informatics there will be less focus on formal exam type assessments and an increase towards coursework based assignments and shorter online tests. The way students receive feedback to get the most from their academic studies also needs to be adjusted to reflect these changes.

Going forward at this current time students will be unable to receive in person feedback from teaching staff, as discussed previously when there was the facility to view scripts in a physical room within the university, or knock on an office door and request feedback. In a time where students were concerned about their academic future, feedback is even more vital to help students make the most of their academic studies. Also to safeguard students mental wellbeing which is a key priority for academic institutions.(Office for Students, 2020). At this given time there is no facility to provide students with elaborated verbal assessment feedback in the school, for which I am focusing my research on. Whereas previously students would have had the opportunity to just 'pop in' to see academic staff physical distancing restriction means this is no longer the case and therefore there needs to be a system in place to pick up on this shortfall. There is currently no effective cohesive university wide used system that is a point of reference for both staff and students which is what I hope to address and provide some solutions to with my dissertation.

3.4 Challenges surrounding students accessing feedback

There are many challenges to overcome when accessing and providing feedback. From a students perspective there is the challenge of access in the first instance. Even prior to the current social climate these challenges existed, Covid 19 simply adds another layer of complexity to a struggling system. There are numerous reasons why accessing quality and informative feedback is an issue. Some reasons may arise due to factors beyond the students control such as the lecturer as failed to provide adequate opportunities for the student to access the feedback. Information may not have been made readily available enough for the student to be aware of the protocols in place. There is the physical limitation recently put in place where students are physically unable to meet lecturers to receive the feedback they require. A lot of students at the school of computer science and informatics are international students. Due to covid 19 many of these students may have returned home rather than remained at university residences. This may cause concerns around accessing feedback due to time zone differences and potential internet access issues.

A further challenge some students may face can be linked to lack of awareness, as well as not being actually aware of the available resources, some students may be unaware of the benefits of receiving assessment feedback and how it could vastly improve their future

academic outcomes. A study by Price et al (2012) found that there was a divide between the assessor and the assessed and students would naturally become passive recipients of feedback and assessment as opposed to active participants engaging in the feedback process.

3.5 Challenges surrounding staff providing feedback

Staff also experience many challenges when providing students with feedback many are overlapping with the challenges students face. Staff have to provide feedback to hundreds if not thousands of students. Crook et al 2011 state that the provision of feedback can be both very time consuming and repetitive. There is a huge amount of logistical planning, and subsequently staff may have fallen into the habit of providing the bare minimum on student feedback. Staff at the school have found there is no structured system to enable seamless feedback, and instead previously had students turning up unexpectedly at the office or emailing randomly. This could lead to missed opportunities to provide quality feedback to students and increase pressures on staff workloads.

The assessment and feedback lead for Cardiff school of computer science and informatics reported that previous methods had not been very effective in enabling feedback and that when doodle poll had been used previously it was very hard to keep track and monitor to be efficiently organised. There was previously only 1 day selected to view exam scripts due to timetabling issues, this could have been restrictive to many students. This could be one reason for low attendance. The slots were generally also arranged 3 - 4 weeks prior. This in itself could correlate to issues of non attendance simply due to appointments being forgotten or conflicting arrangements coming up in the meantime. On the day logistical issues meant students may have been unable to attend, the duration of the time slot was only an hour which allows for a very narrow window of opportunity. It was reported by the staff students experienced issues in locating the room that was booked, which led to some non attendees. There was also a lot of non attendance causing additional strain on staff time and resources. The current circumstances due to covid 19 make this an even more difficult task for staff to effectively communicate with students due to the issues discussed earlier.

3.6 Current circumstances

The current circumstances due to covid 19 have meant the original plan for my dissertation has had to be adapted to meet the rapidly changes circumstances of the university assessment process as discussed previously. Originally the intention was to design a exam viewing booking system for in person on university premises feedback. This has had to be adapted to include a further scope of coursework as well to be included. With the exam moving online the students now have a copy of their script plus the exam question paper meaning that feedback on exams is now very similar to the other types of assessment. The client therefore decided to extend the system to facilitate all types of assessment. There also needs to be consideration that feedback will now be given entirely remotely this may affect the way the time slots are structured when booking, and what technological infrastructure needs to be accessible for example will sessions be booked with the incorporation to Microsoft teams. This is a sign feature that will need to be considered. s

4.Problem

This section will look closer at the problem my chosen dissertation topic is aiming to address. Identifying the key issues and challenges involved and looking at ways to solve them. I will examine what products are already available and what improvements can be made to address the needs discussed in the background section.

4.1 Overview of the problem

Feedback is a crucial part of a students educational journey as recognised by Cardiff Universities Academic feedback policy (2015) who state “Feedback is one of the most powerful influences on learning and achievement.”

Students are paying thousands of pounds for their degrees and yet the standard of feedback being received is the sometimes the minimum policies procedures. Staff are overwhelmed and under resourced. Providing feedback while important is often varied with the change from closed book examinations, when and how to provide feedback is not clear at this early stage in the transition.

My dissertation would aim to build a single cohesive system that can be used across the school by all the lecturers and students to provide ease of access for both staff and students. This system would allow lecturers to provide their availability and track appointments, and show students available dates. The time slots could then be managed by the student. it will allow for multiple courses and modules to be active. Feedback can then be booked electronically with a view to be conducted remotely. My project would also have the feature to allow students to load their work and specify which elements they want feedback on.

4.2 challenges to a single digital system

Whilst there are many obvious benefits to an online exam and coursework booking system, there are also many technical challenges. Designing a single cohesive system that fits everyones requirements is a challenge. Staff want something that creates a

seamless service they can use whilst requiring minimal input and time from their already busy schedules.

Slot management is a concern when creating a single platform. Staff will only have certain times and dates they are available to facilitate feedback. The system will need to be designed with the availability to be able to only select certain times. There is also the challenge of enabling the student to cancel the appointment via the application. Whilst it is important to have the option to cancel an appointment as sometimes this is unavoidable. It also encourages potential making bookings you are not invested in. On the other hand if there is no feature to cancel this could lead staff being inundated with messages from students to cancel or not showing up to appointments which leads to staff time being misspent.

4.3 Technological challenges

There are specific technological challenges to this application, which I will discuss in more detail in the implementation and technology section. My project has been developed as a no cost prototype so there is some functionality that will look different in development compared to production. I am using the javascript FullCalendar framework to demonstrate the availability in staff calendars when booking appointments. A better version in a real world application would be to link the staff calendar more cohesively. In Cardiff University all the systems use Outlook. So I would look to use payed services such as Microsoft outlook or google calendars. Another named service which I will explain in greater detail in the evaluation section is Amazon s3 with Minio for the file upload part of my application. Once again this is due to costs impacting my ability to demonstrate and test.

The software is using a Redis cache for storing user sessions. Redis can be setup on University servers but Cloud providers also offer a fairly expensive service, AWS ElastiCache for example. Another paid service that would be required is an email service. AWS offers the SES, simple email service, that will allow sending emails at cost.

4.4 Current digital solutions

The current system the University had been using was doodle poll. Doodle poll is an online scheduling tool that can find a date and time to meet with multiple people. it creates a

polling calendar which is sent for participants to give feedback on availability. There is some functionality that will be the same, but this is true of the general concept of most booking systems. for example both doodle poll and my design will allow for syncing with external calendars and the ability to stay in control by allowing designated time slots.



However there will be functionality my prototype has that will differ in design. My application will be University specific. This means that it can be a standalone system that can be amended and reused for many assessments with simple modification. By this I mean that during the initial set up academic staff will add themselves and their relevant modules to the database. These will then be able to stay in the system and will only need modification based on the assessment that feedback is to be provided for. This should save staff the time consuming process of setting up individual doodle polls for multiple assessments through the term. It will also include the ability to capture student details in order to create a better user experience. In a real world application this will be done through LDAP which I will explain more in my evaluation section. Students will then have the added functionality of being able to upload files to assist in getting personalised feedback. I feel my prototype will be able to provide a more cohesive and user friendly experience than existing systems with this extra functionality.

5. Approach

When writing my report I choose to use the Agile method. Agile refers to a methodology in software development. It focus's on iterative development. I felt this was the best methodology to use for my project given the current situations impact upon my project. I have had to adapt my project throughout and make adjustments based on the clients requirements as I've been going,

I have found this approach particularly helpful when implementing my project as I am still a novice in programming and learning new languages and frameworks as I go, when required. Splitting them up into iterations using Agile makes this process more manageable.

During my project I chose to focus on the Kanban methodology from to product management and development. This method focus's on reducing the time it takes to complete a project from start to finish. I tried to achieve this by using a Kanban board to continuously improve my flow of work.

The kanban board I chose to use was Trello. Trello helped me monitor my progress in real time. it is also very user friendly and allows for ease of use due to its responsive design.

During the development of my project I found myself switching languages and frameworks during the development process. originally using Python and sqlalchemy and then switching to Java and Springboot. I made this decision due to the lack of documentation with sqlalchemy and trivial queries, by contrast, Spring has excellent documentation, a wealth of online tutorials and a very active community on StackOverflow

6. Application of chosen approach

When commencing the project I had my initial meeting with professor Helen Stevens of Cardiff School of Computing and Informatics. Who is also the head of student assessment and feedback. We discussed the proposal of a digital system that allowed for students to access exam and coursework feedback. We also discussed how the requirements of the project had changed from the original topic due to covid-19. Originally the project was to focus solely on a booking system for exam script viewing. This now had to have a drastically extended scope to incorporate the changing academic structure of the university in line with government policies around social distancing.

6.1 Risk analysis

In the early stages of the project I performed a risk analysis of the project (see risk table in appendices) . I evaluated the risks I was likely to face and strategies to minimise their impact on my project. This was not an exhaustive list as it is impossible to anticipate every possible risk , however it gave me a broad overview of potential pitfalls, that helped me in my planning and organisation process. I felt risk (1) and risk (2) overlapped and my lack of familiarity with language had quite a high likelihood of occurrence due to my own inexperience of programming. This was a risk that occurred during development. I ended up switching tech stacks in the development stage due to implementation issues with my original choices. I estimated the impact has large upon my project which it was to a certain extent, however due to my perceived knowledge of the risk. I had only coded a small part of my project as a test when I ran into the issue. Therefore I minimised the impact and time detriment to my project, and was able to get back on track with the help of my Trello boards.

Risk (3) did present itself during my project however the impact I initially presented as large, was minimised through planning and organisation. I myself found I had limited availability towards the end of my project due to other commitments work related. This did impact my ability to have contact sessions with my supervisor/client, and I did not have as many as perhaps I originally envisioned. To overcome this I tried to make the appointments I was able to have as effective as possible, by having a clear agenda when entering the meeting.

I evaluated risk (4) as a medium risk with medium impact, i feel this was a fair evaluation. I did discover when developing the application that certain features were paid for services which would have impacted my design and testing process. However i planned that I would look at alternatives which is the route i took, which is discussed in more depth in the evaluation. This strategy served me well, the alternatives I found allowed me to continue with the development of my application with minimal disruption.

The time required to develop the model was always a best judgement guess due to my lack of experience (5). This was evidently a risk I encountered as I progressed in my application. I underestimated the implementation time of some features, meaning I had to revert to my strategy to minimise disruption. Which was to initials work on the essential requirements, then import extra features if time allowed. This proved to be an effective strategy, as I believe I implanted all essential criteria, and some extra functionality. Although admittedly there was some features that I did not get to implement which may have added a visual appeal to the application.

To address risk (6) loss of data I implemented Github as my backup and version control. This appears to have been a useful strategy, and was not an issue i encountered during the application process.

6.2 Requirements

When designing the system I needed to consider the requirement designs for two different types of users. It was determined the users of the system would be academic staff and students. While they would have overlapping requirements there would also be a lot of different requirements from both users. When deciding the academic requirements these were discussed in a meeting with the client. The system is primarily being designed for the clients ease of use. The client in this instance focus is on the requirements of academic staff.

For the requirements of the student user I drew on my own experiences of what I would want to achieve from the system and fed these back to the client for review. Based on

these discussions I wrote functional requirements and prioritised them and non functional requirements. Once these were discussed and agreed with the client I designed my documentation with the use of user stories.

From these discussions and documentation I started to design my MVP. Throughout the project I met multiple times with the client to discuss the design and implication of the requirements. Through the agile method I was able to address and feedback and changes necessary with minimal disruption.

6.3 Version control

For my version control I chose to use Github. This allowed me to back up my work and have a working point of reference I could return to in my code when making changes. It is important as a software developer to use version control as it allows you to track iterative changes that you make to your code. You can experiment with your code without fear of breaking the system or losing your data. You will always have a working point of reference to refer back to.

6.4 Implementation

Using the Agile approach i was able to plan the construction of my project and have milestones and targets to achieve this helped me stay on track. Agile is an iterative approach which helps you maintain better communication with the client throughout the production process allowing for easy change and faster output.

7. Product

7.1 Requirements

Based on meetings with the client over the course of my project I created my design document.

Design document (7.1.1)

Design Document

Features:

Must have:

List of academic staff, modules, assessments

This will be maintained by academic staff

Bookable slots

This will be displayed in the form of access to a tutors calendar for which they would have edited their availability

Email confirmation

An email confirmation will be sent upon completion of booking

Ability to upload files

A student will be able to upload their transcript to their booking information in order to assist the tutor to provide personalised feedback

Should have:

Ability to view/ amend/ cancel bookings

This feature should be available to both staff and students, this will aim to reduce non attendance

Could have:

Reminders

This will be sent a suitable time prior to the appointment in order to help reduce non attendance and enhance user experience

Users:

- Academic staff
- Students

Description of user types:

- **Visitor (not logged in)**

For the purpose of this document a visitor will be defined as anyone who has not logged into the system. They can view the interface of the website but will not have access to academic calendars to book appointments

- **Student (logged in)**

A student who has logged in to the system with authorised credentials will be able to book appointments, view, amend, and cancel existing appointments. They will also be able to upload a copy of their transcript ready for their feedback session

- **Academic Staff (logged in)**

A member of staff who is logged into the system with authorised credentials will be able to add themselves to the system, the modules they teach and any assessments they want to provide feedback for. They will also be able to add their Calendar available for bookings and view bookings that have already been made and associated transcripts to that booking.

From my design document as seen above I was able to get a clear plan of the functions I needed to include in my design and created a function document (see appendices)

This document which was created from previous decisions with the client allowed me to prioritise my processes and manage my time accordingly. From this document I was able to get an idea of what my MVP needed to look like and start designing the architecture.

7.2 Use case diagram and use cases

My use case diagrams and the functional requirements can be found in the appendices. They were designed based on discussions with the client. From these use case diagrams I was able to produce acceptance criteria to allow me to test the functionality.

7.3 Interface design

Based on the design documentation and discussions with the client I was able to start the interface design. The website would be composed of two main sections. The staff area and student area.

The student area would be a simple user friendly interface that students could easily locate their tutor to book an appointment to receive feedback. It would store the session details in a database allowing the student to regain access at a later point in order to view, amend or cancel their booking, this session is linked to a unique UUID stored as a cookie in their browser.

The staff interface would be slightly more involved but still maintain a quick and easy to use interface. Staff will have the functionality once authenticated to add themselves to the system as well as their module and assessment details. Staff are then able to set their calendar to accommodate when they are able to provide feedback. It will also allow staff to view transcripts and student data from the pre existing booked appointments.

8.Implementation and technology

The major issue with this project is selecting an API that will let me work with calendars. Working with dates and scheduling is a difficult task and beyond the scope of this assignment to develop a new one. I looked at several APIs and settled on FullCalendar.

FullCalendar is an OpenSource javascript framework (<https://fullcalendar.io/>) that provides an implementation for displaying and working with calendars and scheduling. It allows a user to input events into an interactive calendar, in this case, one to one meetings with tutors. A user will be able to see the schedule for a tutor and click a specific time that they want their meeting. It allows a user to see available bookings well into the future and has different views for months, weeks and days. This is the perfect framework to achieve a user friendly and interactive component for this assignment.

8.1 Technologies used

Docker

One of the issues with developing this system fairly early on is how many different components are required for getting the full system to run. The system, being composed of a separate frontend and backend with different technologies and databases requires several daemon processes to run. These are broken down as follows:

Minio - This is the file storage system used for storing Transcripts, it is the free alternative to AWS S3, it offers security and is less antiquated than a standard FTP server.

Redis - The memory cache used by the frontend to manage student sessions. A student might decide not select an appointment after having uploaded a transcript. A short lived session will clear any redundant data.

Frontend - The nodejs application responsible for rendering html pages. Separated specifically from the backend as a means to play to the strengths of different languages. NodeJs with the use of Express offers a very well documented and industry standard means of implementing the Model-View-Controller architectural pattern.

Backend - The Java Springboot application that persists data and provided the built of the business logic, it directly integrates with the SQL database.

SQL - The database used by the backend to store all persistent data. It would be possible to create a series of scripts to ensure that the system can be started in a local environment for development but that would require a user to have all necessary tools configured on their system. A lot of development time can be lost simply setting the system up and debugging issues unrelated to the application logic itself. For this reason I chose the H2 in memory SQL database. Easy integrations are provided by Spring without the need to write any code at all.

There are so many moving parts that I had great difficulty in setting up and getting everything running and in a stable state; I chose Docker to mitigate this issue to allow for a system that is easy to develop and maintain. Docker is a containerisation technology that provides virtualisation at the OS level. Different applications will be stored within separate containers and can be run on any operating system that has docker installed. The aforementioned services have their own docker containers and are provided by a specific docker product called docker-compose. The containers can be built locally or can be downloaded from a relevant docker hub repository. The Redis and Minio containers are both freely available on docker hub. My H2 SQL database, being in memory did not require a docker container to start, but had I chosen a database such as MySQL, a docker container would have been readily available for me.

The idea behind docker is to create a containerised environment for the sole purpose of running one specific process. For this assignment we have four separate processes, thus we need four separate docker images to run inside a docker environment. Minio and Redis are third party and all have a docker offering. Setting them up is as easy as running a docker-compose command, the yaml file is detailed below

This will expose 2 ports to each individual container defined within this file allowing us use of the third party applications without having to set them up on my operating system.

```
version: '3'
services:
  redis:
    container_name: redis
    image: redis:6.0.6-alpine
    ports:
      - "6379:6379"
    networks:
      - dissertation_default
  minio:
    container_name: minio
    image: minio/minio
    environment:
      - MINIO_ACCESS_KEY=AKIAIOSFODNN7EXAMPLE
      - MINIO_SECRET_KEY=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
    volumes:
      - ./data:/data
    command: server /data
    ports:
      - "9000:9000"
    networks:
      - dissertation_default

networks:
  dissertation_default:
    external: true
```

Frontend

The frontend is written in NodeJS with Express as the framework for handling MVC. Our webpages are rendered by the Nunjucks templating framework.

MVC is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
  <version>2.3.3.RELEASE</version>
</dependency>
```

handle specific development aspects of an application. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects. The model represents the shape of the data. Model objects store data retrieved from the database. The view represents the data to the user and enables them to modify it. The controller handles the user request. Normally the user uses the view and raises an http request which is handles by the controller. The controller processes the request and returns the appropriate view as a response. There are many advantages to using an MVC. It saves time and is an effective resource as it separates business logic so it facilitates multiple views of the same data at the same time.

The Nunjucks templating engine allows us to render HTML server side. ExpressJs offers us a very simple and intuitive interface for supplying a data model to our views. I specifically chose to use server side rendering rather than client side due to its ease of use and my familiarity with the framework from a previous assignment. An alternative approach would be to use a framework such as Angular or React to implement some client side rendering. Given the number of new technologies that I had to learn for this project; I decided that using the familiar Nunjucks framework would be the better path.

Backend

The backend is written with Java and Spring boot. I specifically chose Java for this due to the fantastic community support and on going development of the Spring boot framework. Spring boot is a framework that delegates much of the boiler plate code to its core product.

```
@SpringBootApplication
public class Application {
    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }
}
```

Eg. Setting up a HTTP server in Java is non trivial and I could spend a lot of time trying and will provide an inferior solution. Spring has this built into its starter-web package:

spring-boot-starter-web comes with Tomcat application server bundled. Simply creating a class with a static main method and a single line of initialisation code is enough to build and run an application server:

This simple code snippet will launch a Tomcat application server, bind to localhost and listen on port 8080 by default, the only bit of configuration that is required is the list of dependencies. For this assignment, I chose Maven as my Java dependency manager, for very much the same reason as I chose Spring, Maven has an active community and excellent documentation supplied by Apache.

The dependencies are defined in the projects pom.xml. I used the Spring initialiser app <https://start.spring.io/> to generate a basic pom.xml and maven wrapper. A pom.xml is far more verbose than the equivalent npm package.json but the result is just the same, Maven will create a target/ folder with all libraries and byte code to run much the same as npm and the node_modules/ folder. Another benefit of using Java is that it is an interpreted language which gives us the benefit of being platform agnostic. This coupled with Docker allows us to run our application on any platform. During development I decided to use the H2 SQL database, Spring supports this and it proved exceptionally useful for unit testing database changes.

Spring Data offers database queries by interface method name alone without having to write any explicit SQL. This allowed me to efficiently use my time and focus on important logic rather than trying to figure out a new and difficult to use framework, which is what happened when I initially chose Python and SQLAlchemy.

Unit tests can run without any external database running allowing for rapid development. This also has the added benefit of easy maintainability. A developer can come in and clone the project and instantly run unit tests without having to follow any intermediary steps.

With the frontend and backend separated there is the issue of how they can communicate. Several options were available to me. I could have used a messaging system to allow for

asynchronous communication between the 2 systems, RabbitMQ or Kafka might have been a decent technology for this. Ultimately I settled on HTTP communication between the 2 services with a clearly defined contract written as part of the Open api standard. This clearly defined contract allows the two services to be developed independently. If the services follow and respect their contract then they don't require constant testing against each other. Unit testing can be extremely thorough and ensure that the contract is followed.

Unit testing

The backend uses extensive unit testing. A unit test is a test of a unit of code, be it a function or a class, thus is it very important to write isolated code that favours abstractions. One of the issues that can come about with unit testing is code that is reliant on communication with an external system, such as minio.

```
public String parseFileContents(String filename) {
    File file = minioClient.getObject("bucketName", "objectPath");
    return file.doStuff()
}

@Test
public void testFileRetunsLine() {
    assertEquals("hello world", parseFileContents("file.txt"));
}
```

In this contrived example, the unit test is attempting to assert that the file contents are "Hello world". This would be a simple test to perform but the issue is the `minioClient.getObject` function. It will attempt to fetch the file from the Minio file server. For the sake of a unit test we dont care about going off to a third party to get the file, we simply want a file ready to test. We can solve this problem via Mocking.

I used mocking and the Mockito framework extensively in this assignment. Instead of using the really `MinioClient` in this example, we replace it with a mock object that we can

manipulate to simulate the call to an external system, we simply tell the mock that when “getObject” is called then return any file that we choose.

The Mock object minioClient is injected back into our test class. This can be done easily with the Mockito Annotations @Mock to create our mock object and @InjectMock to inject it.

```
Mockito.when(mockMinioClient.getObject("bucketName", "objectPath"))  
        .thenReturn(somePreDefinedFile);
```

Our mocked minioClient will no longer attempt to communicate with the third party server but will instead immediately return a predefinedFile. This file will be used to test the unit of code that we care about (file.doStuff()) in this case.

Contract testing

It is feasible that even though a contract has been defined, it might not be followed, I was not always able to stick to a rigid agile process and often I found myself implementing a new feature that was premature. This quickly led to test errors that were confusing and difficult to deal with. I needed to ensure that the contract I had put in place in my design was strictly followed. I investigated Consumer driven contract testing to achieve this.

In this project the Backend supplies the contract and thus the backend is considered the provider and the Frontend is the consumer. With a consumer driven contract test, it is the responsibility of the consumer to generate and provide test contracts for the provider to verify. This way I can ensure that the backend is following its own contract and that no breaking changes are made when I create something new in the frontend.

As an example; the backend contract supplies a GET endpoint called /modules/{moduleCode}. This returns the JSON

```
{  
  "module-code": "CMT-123",  
  "name": "python computing",  
  "dateAdded": "2020-01-01"  
}
```

Without contract testing it might be feasible for the backend to change the format of the date, the contract remains the same but the date is no longer useable by the frontend and thus even though all unit tests pass in the backend, the application no longer works due to an invalid date format. If the frontend generates a test contract it can ensure that the backend is following a strict ISO format.

When the backend comes to verify the test contract against the breaking change the contract verification will fail notifying the developer of the precise error giving a very quick feedback loop. I wouldn't have to spend time debugging the broken application.

For this project I investigate contract testing using the Pact framework, which has support for both NodeJS and Java. Ultimately I decided against it as I found it difficult to implement and often found myself overwhelmed with the technologies that I had chosen.

I wanted a way to automate the testing of the application without having to test it manually. I found that development slowed quite considerably as the application grew. This was because it became increasingly time consuming to test the application when a change was made, much of my time was spent debugging new issues that I had caused when some unforeseen error was thrown. I chose Selenium and Cucumber helping with this.

Using the Gherkin syntax I was able to define several automated scenarios that the Cucumber application could run through. Selenium will then run and translate these features into browser actions. This also allowed me to automate testing in different browsers, Firefox and Chrome were tested thoroughly using this method.

I found that once I had implemented Cucumber testing my work rate increased significantly, I was less apprehensive about the changes I was making and had increased confidence that a change I made in one part of the system was not going to affect another.

SQL

I settled on a SQL database for the backend, Tutors are assigned to modules, tutors have appointments, appointments have transcripts. The data structure was a natural fit for a relational database. The specific database chosen for development was H2 due to it being an in memory database which Spring supports. something that is production worthy would require a managed database solution such as MySQL or Postgres. These can be used

with Docker or a cloud provider can manage them for you. Amazon for example offers MySQL and Postgres as part of their RDS service at a price.

I briefly considered an alternative NoSQL database, MongoDB as a document store. Its JSON structure I thought would fit really nice with the type of data I was generating, however, I decided against this as I had no experience with NoSQL within my studies and the strain of learning new tech was too high. see Database diagram in appendices.

Redis

I settled on Redis for the frontend as it is a short lived memory cache. It supports many data structures and is incredibly fast. It can withstand failures due to its scalability and can provide an uninterrupted service. A students session should be relatively short lived. If they navigate to the website, they might try using it without fully committing to setting an appointment. I needed a way to save student data temporarily without persisting temporary data. For example: A student logs in and selects a module and tutor but doesn't commit to an appointment. I don't want to save any of this data and want it erased after a certain time. Redis is perfect for this. Data is only persisted in SQL when an appointment has been selected.

Continuous integration

All development was done within Github. Github offers a new product called Github Actions which allows for all continuous integration, its a free service, even for private repositories.

When I git push a change, GitHub actions will pickup the change and run all of my unit tests against it, notifying me if there are any failures. This speeds development and ensures that tests are always run. Continuous integration is vital for agile development. It improves the quality of your development and reduces risk. With a continuous integration pipeline you can ensure that every commit you publish to GitHub is tested.

My GitHub action here shows my "test" build. When code is committed, Github will checkout the code repository, provision a ubuntu virtual machine, install the Java 11 JDK and run the unit tests via a maven command "mvn clean test". Should any of these steps fail then I would be notified and could make the required changes to fix. This helped me greatly in this assignment as I often forgot to run my tests after a code change.


```
jobs:
  build:
    name: test
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
        with:
          fetch-depth: '0'
      - name: JDK
        uses: actions/setup-java@v1
        with:
          java-version: 11
      - uses: actions/cache@v1
        with:
          path: ~/.m2/repository
          key: ${{ runner.os }}-maven-${{ hashFiles('**/pom.xml') }}
          restore-keys: |
            ${{ runner.os }}-maven-
      - name: unit test
        run: |
          cd backend-spring
          mvn clean test
```

Sending Emails

When a user creates an appointment, 2 emails should be sent out. One to the tutor and one to the student. For the purpose of this assignment I chose to investigate Amazons Simple Email Service (SES) solution. This gives access to an email server allowing me to test sending emails. This is not a free solution, however, so I created a simple NOOP email service for the purpose of simulating emails.

Configuration

it is important to differentiate between a local development environment and a production ready application. One example of this is the configuration. The frontend must communicate with the backend and does this via HTTP calls defined in the openapi spec. Originally I hardcoded this configuration so that it would communicate over the localhost. However, this would not be a valid option for a production system. In production it is likely that the frontend and backend could live on different virtual machines or within a containerised environment. I cannot assume that localhost would be a valid option, thus I have added config to be able to chose the host address and port of the backend, the Minio client and the Redis database. Spring offers an easy implementation for choosing your database driver and connection url.

```
spring.datasource.platform=h2
```

The config item here shows the h2 database has been selected, i could swap this out for mysql with no issue, provided that the correct dependency has been supplied the pom.

The configuration is supplied as environment variables as described in the 12 factor app specification. This is highlighted in the services.yml file, each service has an environment variable section.

9. Analysis

In this section I will discuss my thoughts on the production of my prototype application. Evaluating what aspects I feel were well implemented and what changes could be made.

9.1

Overall I am please with the end result of my application. I feel I have manage to incorporate all the required functionality that I intended to on the initial design stage to satisfy the clients requirements and also potentially some extra functionality. I feel t effectively addresses the problem of creating a consistent cohesive booking system. That is user friendly and effective for both staff and students. The view of the senior manager is that the system if used by every lecturer will enable a constant school wide system in place. I feel that if this was implemented beyond a prototype it would be successful in achieving that. In the deployment stage it would need to have some changes to the technologies used as described in 8.2. However I am confident that after the initial set up of assessment details it would provide a time saving and useful purpose to academic staff.

9.2 Alternative technologies

LDAP - For my application prototype I have deployed a proxy log in page, in that it serves no functional security requirement. It is designed for demonstrative purposes only. When a user enters their details, no validation takes place, any details could be entered and access would still be granted. It has been assigned in this way as it is purely a proof of concept at this stage. If this was to be deployed into real time it would need to conform university policies and protocols. Cardiff Universities security model relies upon a zone based approach, coupled with an ldap-backed role-based access control mechanism for services such as VPN termination (information security, 2020). LDAP (Lightweight Directory Access Protocol) is an open and cross platform protocol used fir directory services authentication. It is a way of seeking to an active directory server, it provides the communication language to to communicate with other directory services to store user details. In the case of Cardiff University LDAP is used across all the applications students access such as learning central and mail. This system if deployed would follow the same protocol. It would authenticate users through their Cardiff University credentials. The type

of authentication access Cardiff university grants is name/password authentication. This grants access to the server based on credentials applied. However it also needs confidentiality protection. To achieve this SASL authentication binds the LDAP server to another authentication mechanism, and uses the ldap protocol to send an LDAP message. This then needs something like a TLS encryption to keep the information secure as it gets passed as clear text, so could be vulnerable.

The syntax for an LDAP query would look similar to the example below:

```
(&(objectClass=user)(sAMAccountName=yourUserName)
(memberof=CN=YourGroup,OU=Users,DC=YourDomain,DC=com))
(Varonis, 2020)
```

AWS S3 - In my application I have implemented a file upload function. To allow students to upload a copy of their transcript into their booking form to allow for more tailored and concise feedback. To achieve this I used Minio to store the files as discussed earlier in the technology section. Minio is an AWS S3 emulator that allows us to do local development without having to integrate directly with AWS S3. In the production system Minio would be dropped for AWS S3. it was constructed in this way for my application as it is a cost free proof of concept. Where as AWS S3 is a paid service.

AWS S3 is a storage service for the internet that is designed to make web-scaling computing easier for developers. Its web service interface allows for ease of use when storing and retrieving any amount of data, at anytime from anywhere on the web. Amazon (2020) states it gives the developer access to the same highly scaleable, reliable fast, storage and infrastructure that Amazon uses to run its own networks. I would use use AWS S3 in deployment as I feel it would be the infrastructure to use for the previous reasons plus it can be used in conjunction to LDAP By connecting an on premises active directory to AWS using AD connector. it also also the advantages of creating buckets to store an infinite amount of data. You can use score permissions to stop unauthorised access. AWS S3 also uses a standard interface which uses the standards- based REST and SOAP.

Microsoft Outlook and Exchange - When creating my application as cost free as discussed earlier for the calendar requirement I used fullcalender io discussed in more detail in the technology section and for email I used amazon simple email service (SES). In deployment I would use Microsoft Outlook as this is the application used by Cardiff University across its network. Microsoft outlook is an information manager from Microsoft it can used by individuals as stand alone software or by organisations as a multi user software through Microsoft exchange server. This allows for shared functions such as mailboxes and calendars and appointment scheduling. So in a production ready product the application would be used through Microsoft exchange so when a student books an appointment, using their university credentials they would receive an email confirmation to their student email address. This would be a requirement in the booking system that their university email address was used. It would also be able to link up with their outlook calendar and have the functionality to integrate with Microsoft teams for video calling.

9.3 What went well

I feel that there was a few aspects that worked really well, the first was the use of docker. It allowed the containerisation of many different processes and made it really easy to create, deploy and run my applications. I could put all my dependencies in one package that meant configuration was simple. It also means that i can have confidence that my application should run on other machines despite their configurations. Docker would also be great for scalability if the prototype was to go live as it has rapid deployment across multi cloud platforms.

I am also pleased with the implementation of Github actions. This was a new technology for me and I found it helped with rapid development as i was confident every commit was working. GitHub Actions work by helping you "automate tasks within your software development life cycle" (Github 2020). Due to their event driven nature every time someone creates a pull request of a repository you can automatically run command that a executes a test.

9.4 what I could improve upon

My main dissatisfaction with my prototype is the use of the authentication codes i generated using UUID V4. A UUID V4 is a unique identifier that is generated randomly and with no inherent logic. Once a student had completed the booking process the system, the student was directed to the completion page where a UUID V4 was generated. This is a unique identity number that would allow the student to then use to access and edit their existing booking. The probability of UUID'S being duplicated is not zero however it is so low it is considered negligible. I feel that LDAP would provide a better all round user experience, however as discussed earlier this was not possible to implement.

Currently when a file is uploaded, it is done through nodejs. The entire contents of the file is loaded into memory via multer. A better solution would be to stream the file from the frontend directly to the AWS S3 server there is a possible issue that a spike of students could upload a file all at the same time, if there files were sufficiently large then we could run out of memory which would crash the frontend streaming the file would not have this issue as each specific chunk will be removed from memory when it is forwarded to the S3 bucket which limits the amount of memory the frontend will consume

10. conclusion

I feel through my prototype design I have created a suitable digital solution to meet the needs of academic staff when arranging the delivery of feedback of assessments to students.

My background and problem section helped me structure my project and address the requirements needed in the design process.

I tried to keep the design as clean and user friendly as possible for both the students and academic staff. With this in mind I tried to keep the layout of the interface quite minimal. This allowed me to have a clear vision of what I wanted the interface to look like. I also wanted consistency throughout. The use of bootstrap allowed me to achieve a clean and concise interface and maximise efficiency.

Through a very small sample of students I have been able to gather some user feedback who have all commented on the consistency and ease of use of the design, and would feel confident in using the application.

In order to get a better understanding of whether this could move beyond a proof of concept to would need to be distributed to a larger sample size of both students and staff. This is something that I unfortunately did not have time for in this project. For example the initial test could be done with one module. This would give a good idea of student uptake and any issues that arise could be addressed and improved. The client expressed the design to focus more on the ease of use of staff. It would need to be tested among staff members to illicit if they feel its a system that they could incorporate into the assessment regime, and if it would be beneficial to how they organise and deliver feedback, or if it simply would be of no benefit to them. It could be the case that the system might provide some aspects that staff would find beneficial but would need some amendments. This kind of feedback could only be acquired through proper sampling methods.

If the prototype was found to be an effective tool to facilitate feedback it would have the potential to be utilised with the school of computer science and informatics for hopefully all years and modules. If effective it could also be trialled within other academic departments. It would require minimal administration from staff after the initial set up. Staff would just

need to check module information was correct and update, and then simply set their availability for that module year after year when required.

11. Reflection

This was a project that I was excited to pick as I felt I could further develop a lot of the skills I had learnt so far in the course and have a go at learning a lot of new technologies I had not encountered. I also felt as a student myself I could relate to the project and hopefully bring value to it for future students on the programme.

The product has given me the opportunity to better develop my understanding of the developer client relationship in regards to product and design requirements. When completing my project the modules I had previously studied that I found useful and was able to further develop were software engineering, web development and Java. There's a lot of technologies that I used that I had never previously encountered, or worked very little with. These included mySql, springboot, docker, minio, fullcalender, authentication, and GitHub actions.

11.1 Skills development

On reflection I feel this project has given me invaluable insight into a whole host of new technologies and made me more self aware in trying out and experimenting with different frameworks and applications. I feel my skills as a developer have grown during the course of this project. I learnt how to source new and alternative technologies such as fullcalender and minio, which I had never previously heard of before doing this project. I also got to increase my knowledge of concepts I had only touched on previously such as springboot, docker, ldap, and GitHub actions. I am able to take this new found knowledge and apply it onto future projects.

11.2 Desired production state

During the design process I discovered that I had been overly ambitious with some of my design elements. I was unable to incorporate certain design features into my prototype as discussed in the evaluation. These issues were something I had not previously considered when starting the design process. I feel this can be credited to my lack of in depth knowledge of using a lot of different systems and tools and the complexities required

when using them. A good example of this is Microsoft exchange. I did not anticipate during the initial design process that I would not be able to use this application. Only upon researching the implementation did I discover the barriers to implementation in a prototype design.

Tools such as trello board I found extremely helpful in tracking my progress when I utilised them. I feel in future projects I would try to make better use of all the features it offers. I do not feel I made adequate use of Trello boards during this project. This could be do to the fact I was working alone on my project. Although I attempted to keep a structure to my project, as I was not accountable to others did was not always the case in practice. Going forward in my career I anticipate a lot of my projects will be working within a team. I feel that Trello boards would be a very good tool to use to allow for effective collaboration in the future.

In future projects to enhance my learning further, I think I would like to focus on scalability of the application. One avenue to possibly explore would be the use of the alternative technologies as mentioned in 8.2. I am particularly interested in using the cloud to enhance scalability of the product and linking the application to a relational database such as mysql as opposed to an in memory database.

References

- Black P and Wiliam D. 1998. Assessment and classroom learning. Assessment in Education: Principles, Policy and Practice 5(1): 7–74.
- Blischak, D et al. 2016. A quick introduction to version control with Git and GitHub. Public library of science(PLoS). PloS computational biology (12) 1. doi: [10.1371/journal.pcbi.1004668](https://doi.org/10.1371/journal.pcbi.1004668)
- Bloom et al. 2018. Perceptions and performance using computer-based testing: One institution's experience. Currents in Pharmacy Teaching and Learning. 10. pp 235 - 242.
- Bolton, P. Higher education student numbers . House of Commons Briefing Paper. Number 7857, 14 July 2020 [accessed 15.09.20]
-
- Cardiff University. 2015. Academic Feedback to taught students, policy and guidance. pp 1 - 10.

Cardiff University. 2020. Information Security. Available at: <https://sites.cardiff.ac.uk/isf/for-researchers/q-how-does-the-university-defend-itself-against-viruses-and-malware/> [Accessed: 17 September 2020].

- Carless, D. and Boud, D. 2018. The development of student feedback literacy: enabling uptake of feedback, Assessment & Evaluation in Higher Education, 43:8, 1315-1325, DOI: 10.1080/02602938.2018.1463354
- Crook et al. 2011. The use of video technology for providing feedback to students: Can it enhance the feedback experience for staff and students. Computers and Education. 58. pp 386 - 395.
- Deeley, S.J. et al. 2019 Exploring the 'wicked' problem of student dissatisfaction with assessment and feedback in higher education, Higher Education Pedagogies, 4:1, 385-405, DOI: 10.1080/23752696.2019.1644659

- Dhawan, S. 2020. Online Learning: A pancreas in the time of COVID-19 crisis. Journal of Educational Technology Systems 2020, Vol. 49(1) 5–22 sagepub.com/journals-permissions DOI: 10.1177/0047239520934018
- Flores, M,A. et al. 2015 Perceptions of effectiveness, fairness and feedback of assessment methods: a study in higher education, Studies in Higher Education, 40:9, 1523-1534, DOI: 10.1080/03075079.2014.881348
- Grint, K. (2008) 'Wicked problems and clumsy solutions: The role of leadership', Clinical
- Leader1(2) <http://leadershipforchange.org.uk/wp-content/uploads/Keith-Grint-Wicked-Problems-handout.pdf>
- Irons, A. 2008. Enhancing learning through formative assessment and feedback. Key guides for effective teaching in higher education. Abingdon, UK: Routledge.
- Lane,D,J. and Gibbons,F.. 2007. Social Comparison and Satisfaction: Students' Reactions After Exam Feedback Predict Future Academic Performance. Journal of Applied Social Psychology 37, (6), pp. 1363–1384.
- Mann, S.J. 2001. Alternative perspectives on the student experience: Alienation and engagement. Studies in Higher Education, 26(1), 7–19.
doi:[10.1080/03075070020030689](https://doi.org/10.1080/03075070020030689)
- McBrien, J. L., Cheng, R., & Jones, P. (2009). Virtual spaces: Employing a synchronous online classroom to facilitate student engagement in online learning. The International Review of Research in Open and Distributed Learning, 10(3), 1–17.
- McCarthy, J. 2017. Enhancing feedback in higher education: Students' attitudes towards online and in-class formative assessment feedback models. 18(2) pp 127 -141
- Miller,T. 2009. Formative computer-based assessment in higher education: the effectiveness of feedback in supporting student learning, Assessment & Evaluation in Higher Education, 34:2, 181-192, DOI: 10.1080/02602930801956075

- Morris, C. et al. 2019. Case study: suggesting choice: inclusive assessment processes, Higher Education Pedagogies, 4:1, 435-447, DOI: 10.1080/23752696.2019.1669479
- Office for Students. Mental health: Are all students being properly supported? Available at: <https://www.officeforstudents.org.uk/publications/mental-health-are-all-students-being-properly-supported/> [Accessed: 01 September 2020]
- Thompson, B. 2009. What role for TV in wired world? Available at: <http://news.bbc.co.uk/1/hi/technology/8115671.stm> [Accessed: 10 September 2016].
- Sobers, R. 2020. The difference between active directory and LDAP. Available at: <http://varonis.com/blog/the-difference-between-active-directory-and-ldap>. [Accessed 17.09.20]
- Wibowo, S, et al. 2016. A Pilot Study of an Electronic Exam System at an Australian University Journal of Educational Technology Systems . 45(1), pp 5–33.

Appendices

i)

Table 1

	Risk	Likelihood	Impact	Strategy to minimise disruption
1	Lack of familiarity with the languages and programmes I'm using leading to errors	High	Large	1. Learn the code and frameworks needed for the project prior to coding
2	Changing direction leading to project delay	Medium	Large	Due to my unfamiliarity with some programmes, I may have to test out various options. I will try to test a small sample at the start to determine the best fit.
3	Illness or unavailability	Medium	Large	If myself, client or supervisor is unavailable this could delay the project. To minimise this impact I will do lots of iterations to keep track of progress
4	4. Technical problems cause delays in the project. (e.g. no access to a necessary software)	Medium	Medium	I will look at alternative software
5	The time required to develop as model is underestimated	Medium	Large	I will initial work on the essential requirements and add more features if time allows
6	Loss of data	low	Large	I will use Github as backup and version control

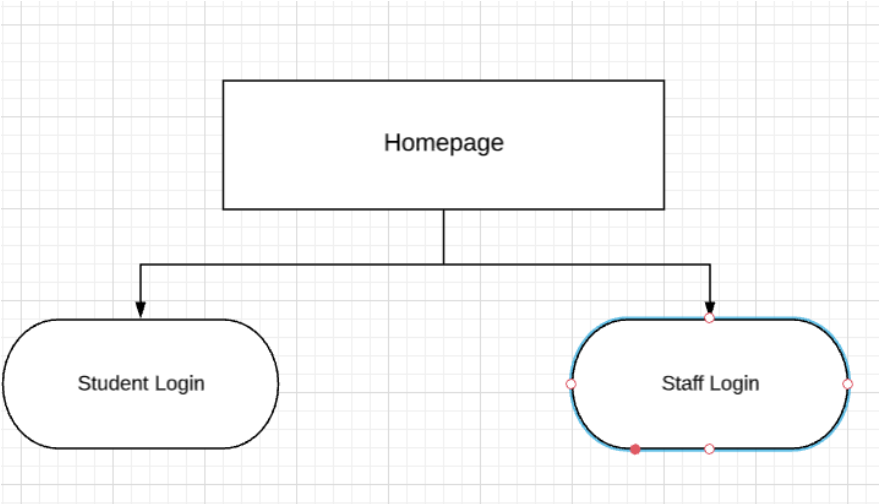
ii)

User Stories		
User Story ID	Given	When
1	I am a Student	I am on the appointment page
2	Student	I make an appointment
3	Student	I receive an email reminder
4	Student	I am on the appointment page
5	Student	I am on the appointment page
6	Student	I am on the appointment page
7	Lecturer	I am on the appointment page
8	Lecturer	a student books an appointment
9	Lecturer	I am on the appointment page
10	Lecturer	I am on the appointment page
11	Lecturer	I am on the appointment page
12	Lecturer	I am on the appointment page
13	Lecturer	I am on Blackboard
14	Lecturer	I am on the appointment page
15	Lecturer	I log into the app

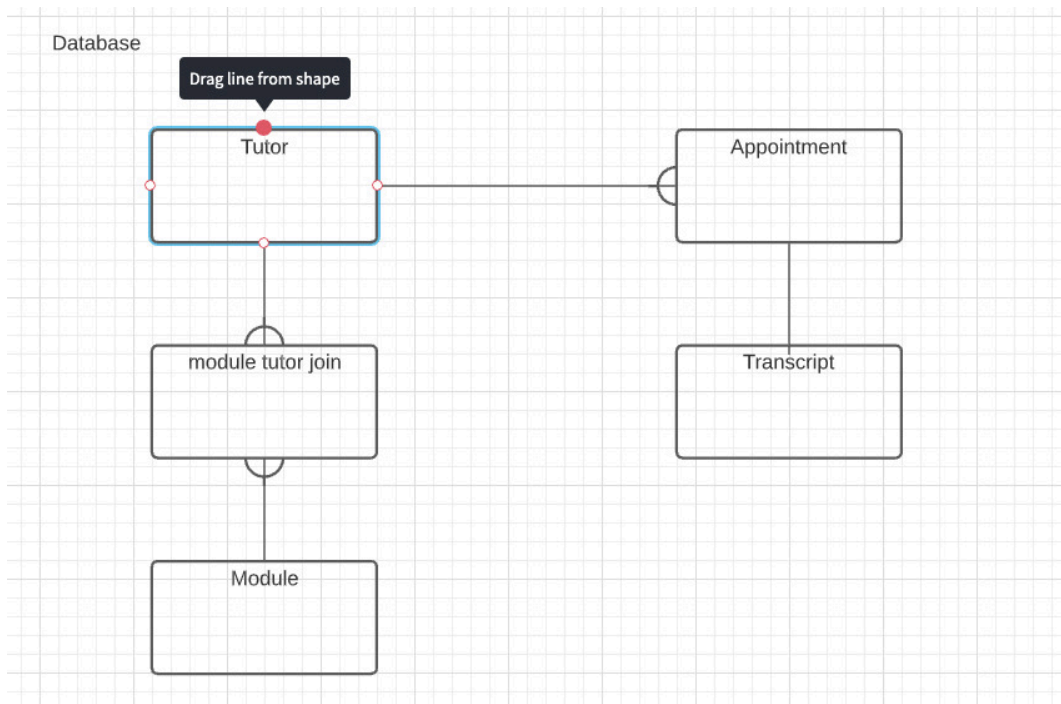
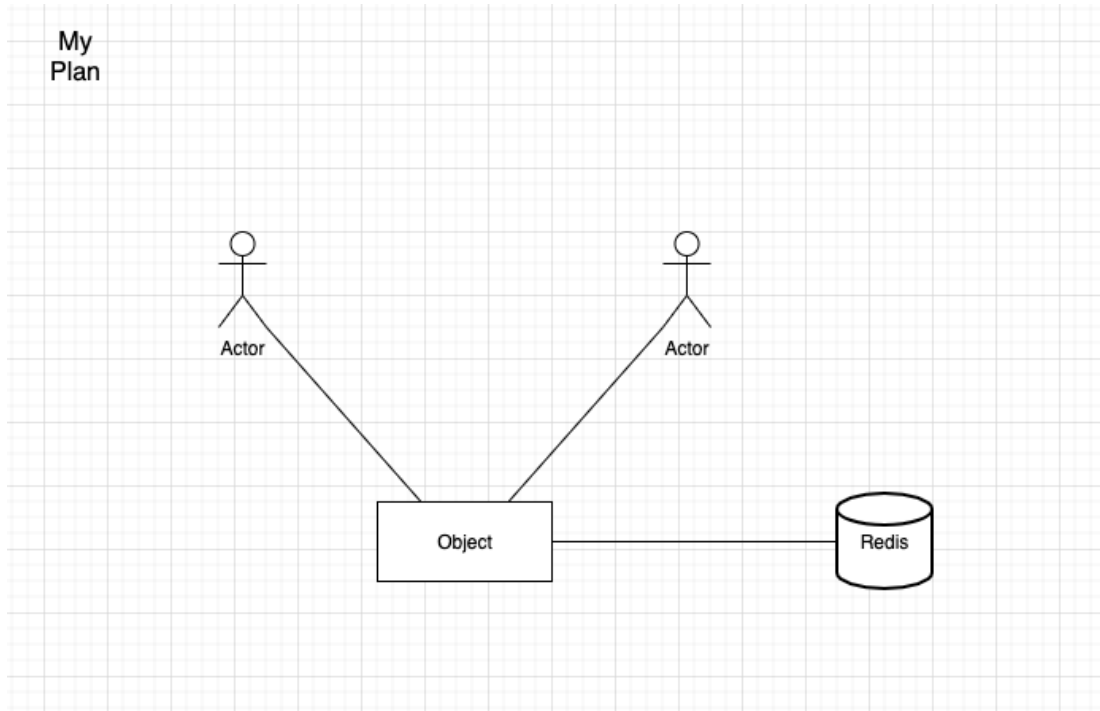
User Stories

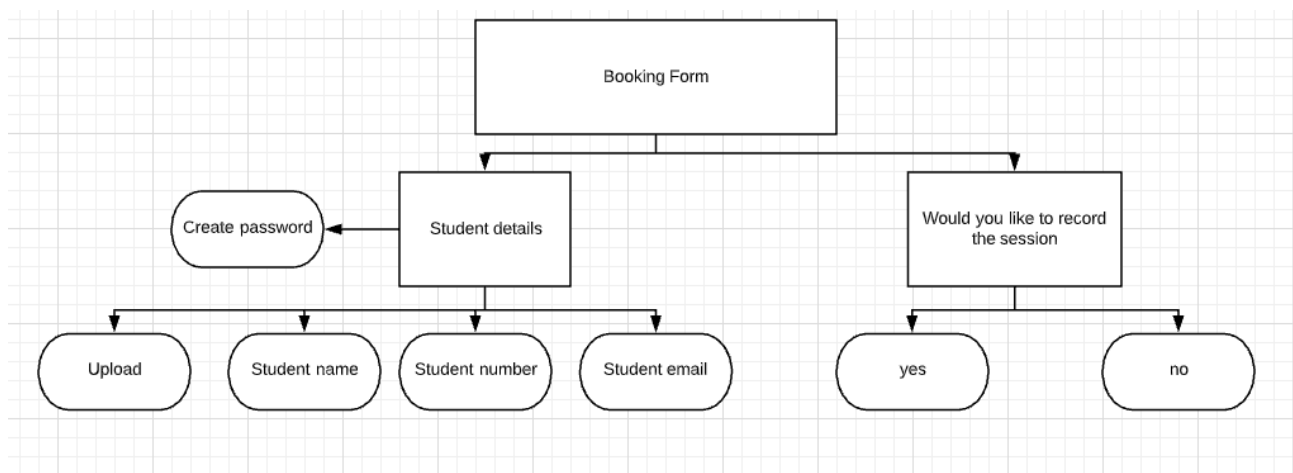
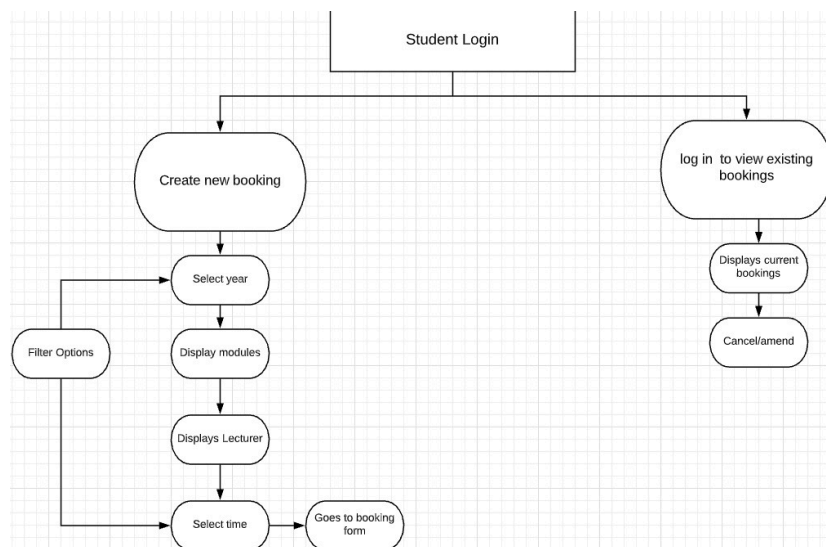
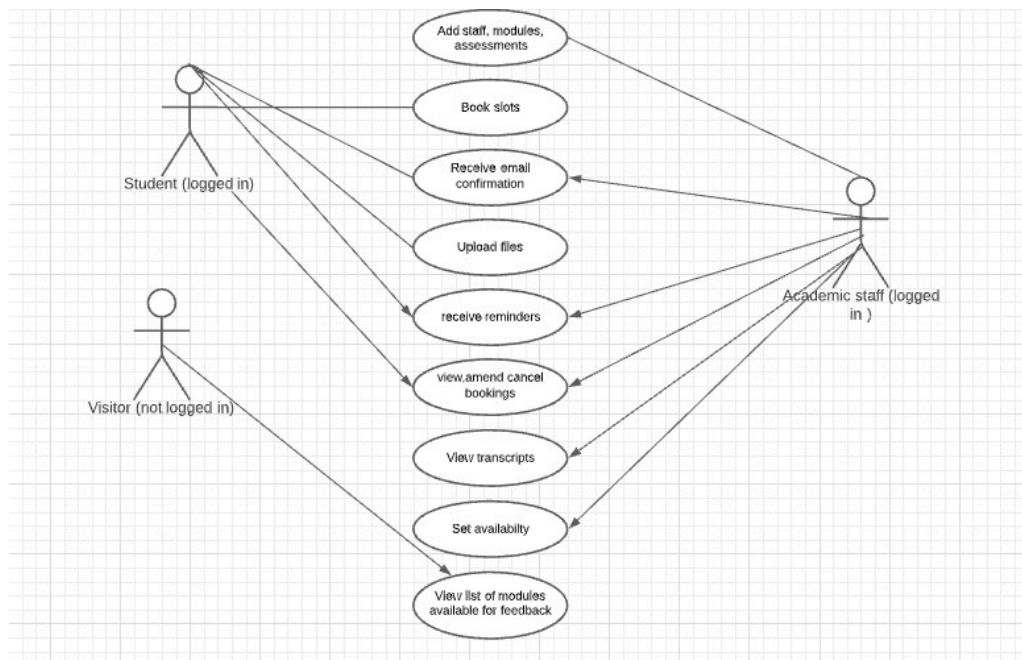
User Story ID	As a	I want to:	Acceptance Criteria	Essential	Story point estimate
1	Student	Chose the time i want feedback from the options available		Yes	2
2	Student	Receive a confirmation email to my university email address	only Cardiff university email	Yes	8
3	Student	Display my appointment date and time linking to my calendar	only outlook		spike
5	Student	Have the option to upload the section I wish to receive feedback on			8
6	Student	Have the option of selecting feedback in different formats ie video or chatbox			2
7	Student	Have the option to record the session so I am able to review the feedback at my leisure			2
8	Student	Be able to cancel or amend my appointment	must be within the open slot timeframe		5

iii)



iv)





One Plan

