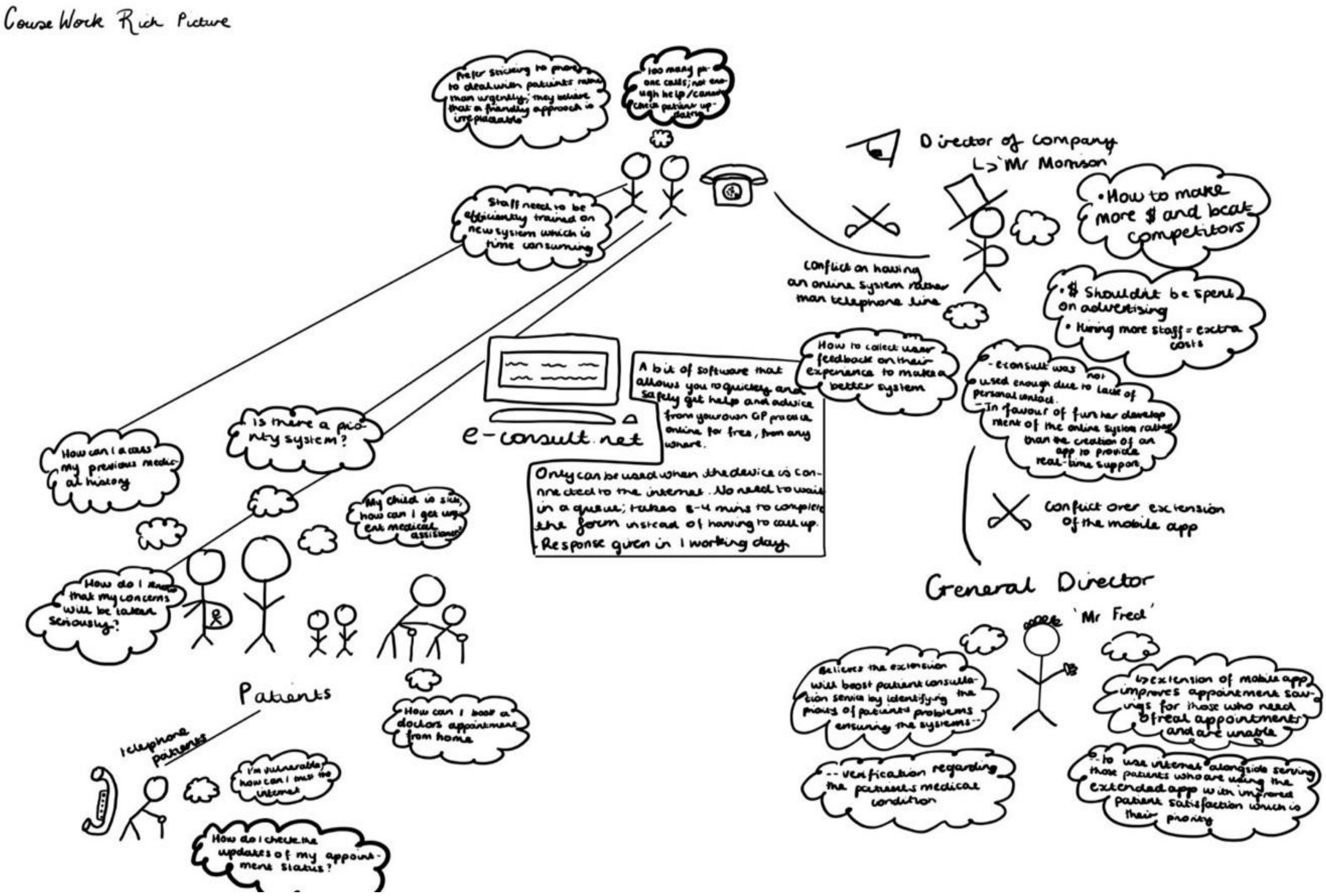
Case study: e-Consult.net

The 5 P's

Problem:

- The system does not check the user's problem description, this creates a problem where the user's needs might not be addressed properly. The new system should therefore allow the users to enter their preferences/requirements and use those to direct them to the right department.
- The system does not yet have a priority queue which is vital in any medical-related application due to the urgency aspect of this field. The new system needs a priority queue with clear requirements regarding the ranking of each situation.
- There is no way to check the status of the consult in the current system which might frustrate some users that are waiting for a long time and cause them to send multiple similar requests that could slow down the system/process.

Rich Picture:



Couse Work Rich Picture
Prefer sticking to phones to deal with patients rather than urgently; they believe that a friendly approach is irreplaceable
100 many phone calls; not enough help / cannot check patient up-dates
Staff need to be efficiently trained on new system which is time consuming
Director of company
L> 'Mr Morrison
• How to make more $ and beat competitors
• $ Shouldn't be spent on advertising
• Hiring more staff = extra costs
conflict on having an online system rather than telephone line
How to collect user feedback on their experience to make a better system
- econsult was not used enough due to lack of personal contact.
- In favour of further development of the online system rather than the creation of an app to provide real-time support
conflict over extension of the mobile app
A bit of software that allows you to quickly and safely get help and advice from your own GP practice online for free, from anywhere.
Only can be used when the device is connected to the internet. No need to wait in a queue; takes 3-4 mins to complete the form instead of having to call up. Response given in 1 working day.
e-consult.net
Is there a priority system?
How can I access my previous medical history
My child is sick, how can I get urgent medical assistance?
How do I know that my concerns will be taken seriously?
Patients
How can I book a doctors appointment from home
telephone patients
I'm vulnerable, how can I trust the internet
How do I check the updates of my appointment status?
General Director
'Mr Fred'
Believes the extension will boost patient consultation service by identifying the priority of patients problems ensuring the systems --
-- verification regarding the patients medical condition
L> extension of mobile app improves appointment savings for those who need of real appointments and are unable
- to use internet alongside serving those patients who are using the extended app with improved patient satisfaction which is their priority

Above is the rich picture for the eConsult.net, depicting the concerns and experiences of each entity involved in the creation of this app. It highlights how each entity has intertwining and conflicting concerns and helps identify how to address these concerns for easily that we may otherwise miss. An example of a concern is from the patients: ’How do I check the updates of my appointment status.’ This concern can now be addressed, and a change can be implemented when programming which allows for patients to check the status of their requests.

Project:

When developing a solution to a problem, it is imperative to keep the triple pillars of time, budget and most importantly, performance in mind always (Taherrdoost 2015). The focus is to create an accessible, user-friendly system which allows patients to effectively book their own doctors' appointments from the comfort of their own home and to be allocated the right treatment service depending on the severity of their condition. This app should be able to improve the patient's experience by enabling patients to see their own medical history, their appointment status and be bumped up the waiting queue if they fall within the category of a child under 6 years, elderly patients 65 years or older and patients with critical health conditions.

To ensure that this system is effective, there are several factors to take into consideration like the time frame of 18 months to execute a successful functional system and a budget of £800,000 given by the senior management.
This project needs to be efficiently managed meaning there needs to be different roles assigned to each member of the team. There should be a project manager who will watch over the project and ensure each member of the team is working efficiently and managing their time well to ensure all the expectations from the creation of the app are met and up to par. There need to be well trained developers who can design and create the online system and they should be able to take feedback from other members of the team to ensure that they make the best system possible. There should also be a team who can ensure that quality is maintained throughout the development stage of the app.

Regarding the budget, $800,000 has been allocated to the production of this system so it needs to be distributed effectively amongst the different teams of this system and there should be a part of the budget set aside in case something fails or does not go according to plan. Risk management should be taken into consideration when developing this system as the benefits of it when implemented into projects are tremendous. It is one of the ten knowledgeable spheres in which projects managers must be competent in (PMBOK 2008). Potential risks should be identified like potential server failures, or the system being hacked.

Lastly, for this system to be successful post development, the users of the system need to be properly trained. This includes both the staff members and the patients using the system to ensure that their preferences can be properly implemented when booking their appointment.

Product:

The final product should be a fully functioning system that meets all the requirements stated at the beginning and the changes made along the way. An online system/ app which runs smoothly where the patients can access their own individual health record from anywhere geographically at any given time. The system should include a user guide to ensure every user knows how to use the software efficiently which can be done by creating a help button to answer any queries that they might have about using the system.

Some artefacts that must be provided in the final product include:

- Documentation on database and product specifications.
- The system prototypes and previous versions of the product.

- Test cases and results
- Terms and conditions of use
- Support/maintenance services

People:

Including: the development team, end-users, project directors/leadership, clients and stakeholders. The project should have experienced, empowering management/leaders to motivate the development team and ensure smooth delivery of the solution. End users should be involved in the development of the solution to ensure a predictable and successful outcome. "Leaders" must sensibly monitor the project at every stage, so they are up to date with the development process and can identify inaccuracies early on. They must also try to recognize that some developers might be slower to adjust to change or simply less capable than others and adapt accordingly (e.g. by distributing some tasks to those developers that are more suited for their expertise or train them).

Process:

- The new system has mostly clear requirements, a tight deadline and a budget to meet, and would preferably not have any downtime due to being integral in helping users with medical concerns and clearing appointment dates for patients with critical conditions.
- Two fitting development methodologies in this case could be: The **V-model** or **Agile model** (SCRUM). More about them in the table below …

V-model VS Agile model (Part of Process)

	V-model	Agile model
Advantages	Is great in projects with clear requirements. It is easy to understand and predictable which could reduce the time needed for the developers to familiarise themselves with it therefore increasing the chances of meeting the tight deadline.	Specialises on iterative development of a problem and constant improvement. Constant user feedback means the final system will certainly meet user requirements and preferences increasing satisfaction.
	Implements constant testing so errors are caught early on in the development stage. Reduces possible downtime for repairs and can improve customer satisfaction and retention.	This model focuses on rapid requirements delivery so most suitable in terms of prioritising critical components such as the priority system for patients and implementing it as fast as possible.
	This model is structured and controlled; it ensures patient data security by reducing errors and mistakes.	Uses pair programming for code quality checks and decreasing human error. Code is more reliable from the beginning, so less time is spent refactoring/fixing it therefore reducing the total time required to complete the project.
Disadvantages		
	Does not adapt to changing requirements well which could happen as the director has different views on how the system should be developed. It will be very expensive to restart the development process using the V-model if the requirements suddenly change.	Complex. It could require more experienced developers (increasing cost) or more time to train existing developers.
	There is limited end-user involvement and feedback. If the system fails to deliver on customer expectations, it will have to be redeveloped which would also be very expensive.	In case there are constant requirement changes, the deadline can be pushed back significantly.

Although both models are suitable for our project, Agile seems to be more appropriate in this case for reasons stated below:

- There seem to be a clash of opinions on the final system (Mr. Morrsion and Mr. Felix) which could incur a spontaneous change of requirements mid development, Agile's flexibility makes

  it possible for any necessary changes to be made while the V-model does not accommodate this.
- It follows an iterative approach, so it allows for constant feedback to be implemented into the system so everyone's expectations are met (The users, stakeholders, development team, Mr. Morrsion, Mr. Felix etc.).
- Agile is quick. It is based on creating a solution that is 'just' viable quickly and updating it to add new features incrementally. This speed is desirable due to the tight deadline of 18 months.

Implementing a work team enforces motivation, productivity and innovation within the creation of the system (Brede Moe 2010) .To break down the responsibility of the agile development, each team member will be allocated one of the following roles: Product Owner, Development team member, stakeholder and SCRUM master, however in smaller teams, the product owner can act as the SCRUM master too. The Product owner is responsible for being the representative of the stakeholders. The Development team member has the responsibility of delivering work through the sprint and ensuring transparency when delivering it. The Stakeholder. The SCRUM master is responsible for ensuring that everything works together efficiently and aids the product owner by helping them understand communicative values and managing backlogs better.

Functional Requirements

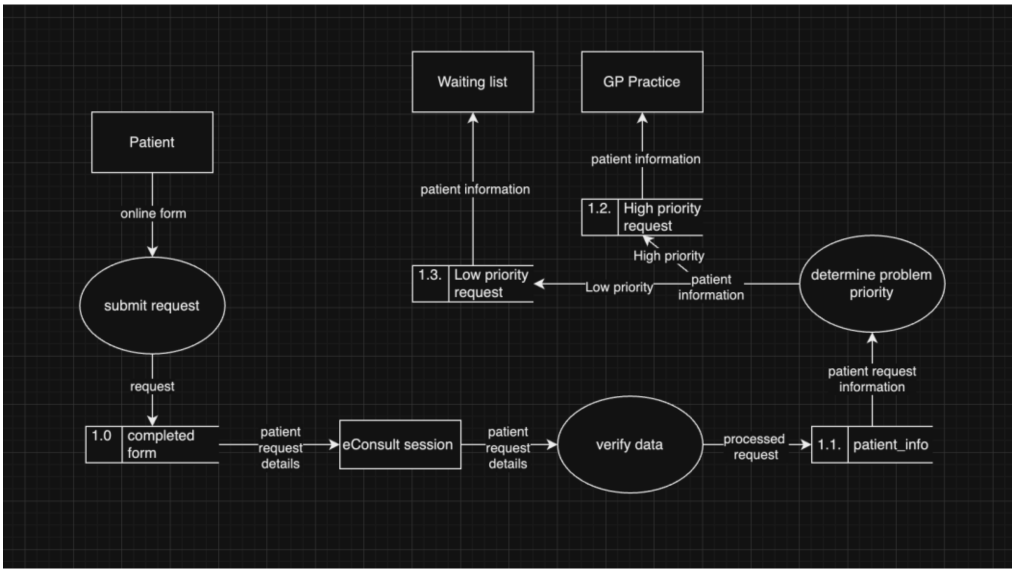
- Users must be allowed to enter their preferences (address, nearest GP practice) to better tailor the eConsult session to their needs and circumstances.
- Users must be able to create an account and log in to access their information.
- Databases must be implemented to efficiently store and retrieve data about each entity involved in the system.
- Needs to have an accessibility option for users to be able to speak into the microphone to select options.
- The system must implement a priority queue in this order:
  1. Children under the age of 6
  2. Elderly patients 65 or older
  3. Patients with critical health conditions like heart problems
  4. Everyone else

Non-Functional Requirements

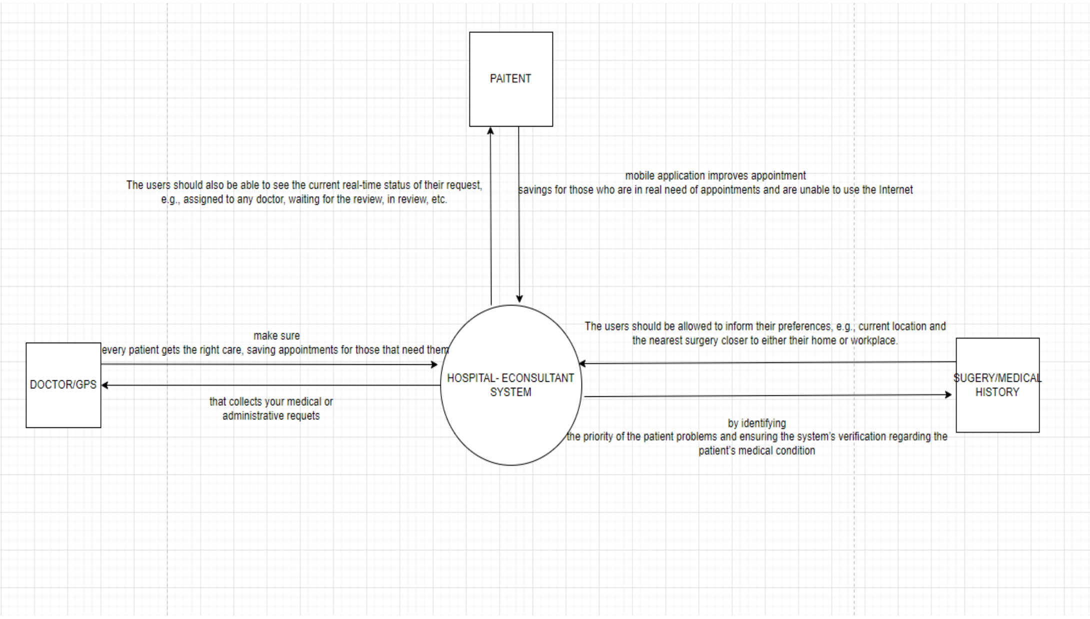
- The project should fit within a budget of £800,000 and should be finished by the deadline of 18 months.
- It's important to consider security in this system to protect the patients' personal information (e.g. address, medical records, telephone number etc.) and prevent legal action against the developer company.
  - Any implemented databases should be protected by a firewall.
  - The users are required to set up a strong password when creating an account (e.g. minimum of 10 characters, including numbers and special characters).
  - The account should have another layer of validation like authenticator apps/message confirmation or security questions to further protect against unauthorized access.
- The system should be run on a sufficiently powerful server to handle high traffic, ensure scalability potential, and prevent crashing/downtime. The implementation of this server requires new computers/hardware to be bought and set up by experts.

- The user interface should include a status display to inform the user of the progress of their request in real-time (request sent, awaiting review, review in process, completed, booked appointment etc.)
- EConsult are targeted to users of all ages and demographic meaning we need to ensure the software is user-friendly and easy to operate for feasible use.
- The software should be accessible for users who are at a disadvantage. For example, individuals with visual impairment or hearing difficulties.

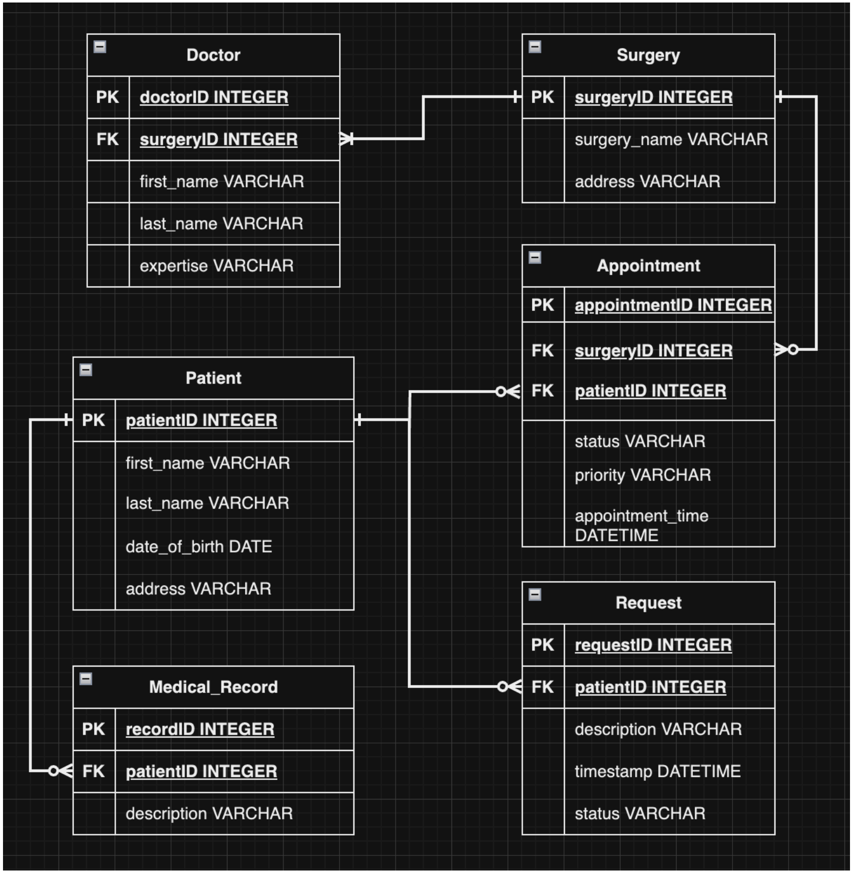
Data Flow Diagram



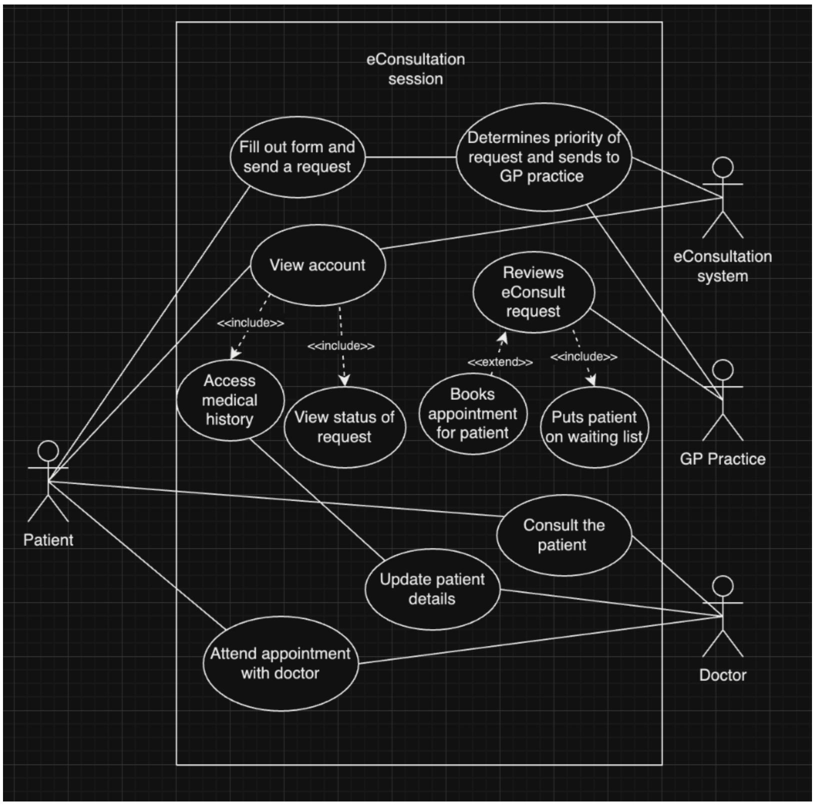
Level (0)


PAITENT
The users should also be able to see the current real-time status of their request, e.g., assigned to any doctor, waiting for the review, in review, etc.
mobile application improves appointment savings for those who are in real need of appointments and are unable to use the Internet
make sure every patient gets the right care, saving appointments for those that need them
DOCTOR/GPS
HOSPITAL- ECONSULTANT SYSTEM
that collects your medical or administrative requets
The users should be allowed to inform their preferences, e.g., current location and the nearest surgery closer to either their home or workplace.
SUGERY/MEDICAL HISTORY
by identifying the priority of the patient problems and ensuring the system's verification regarding the patient's medical condition

Entity Relationship Diagram

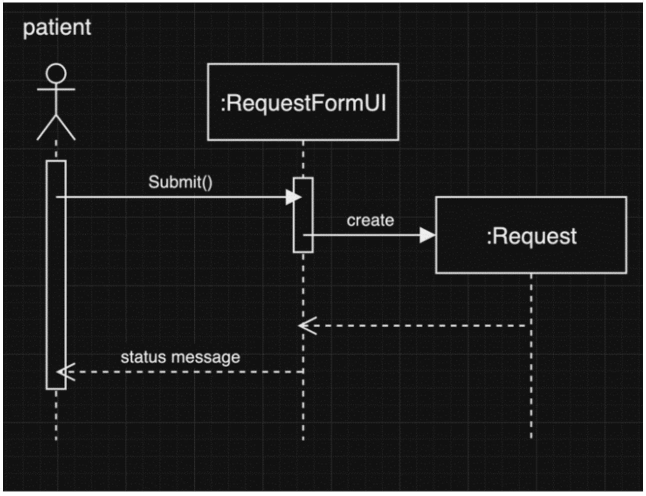


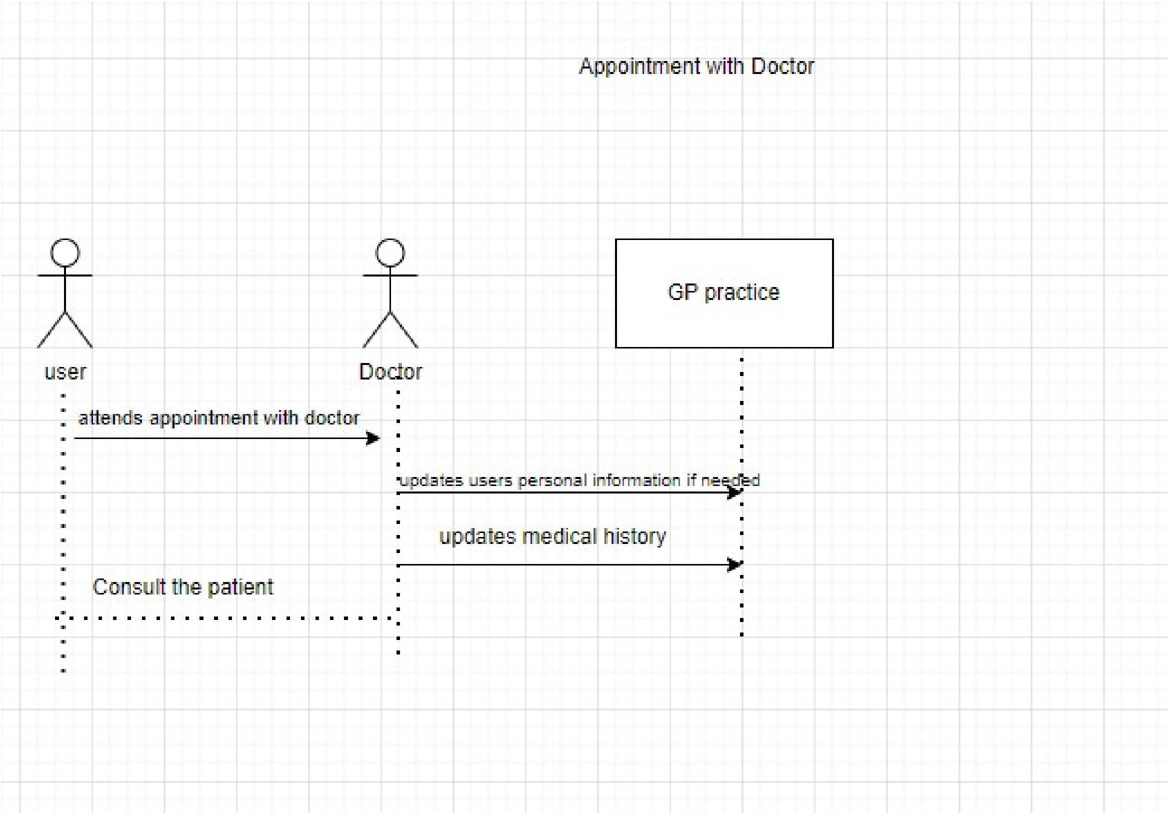
Use Case Diagram

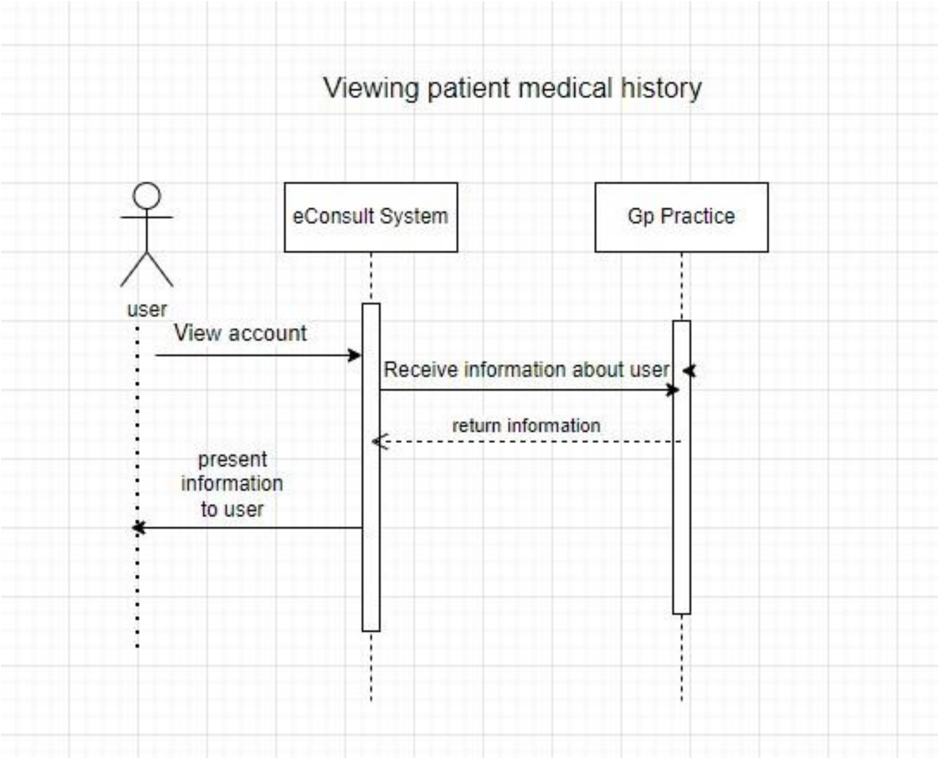


Sequence Diagrams

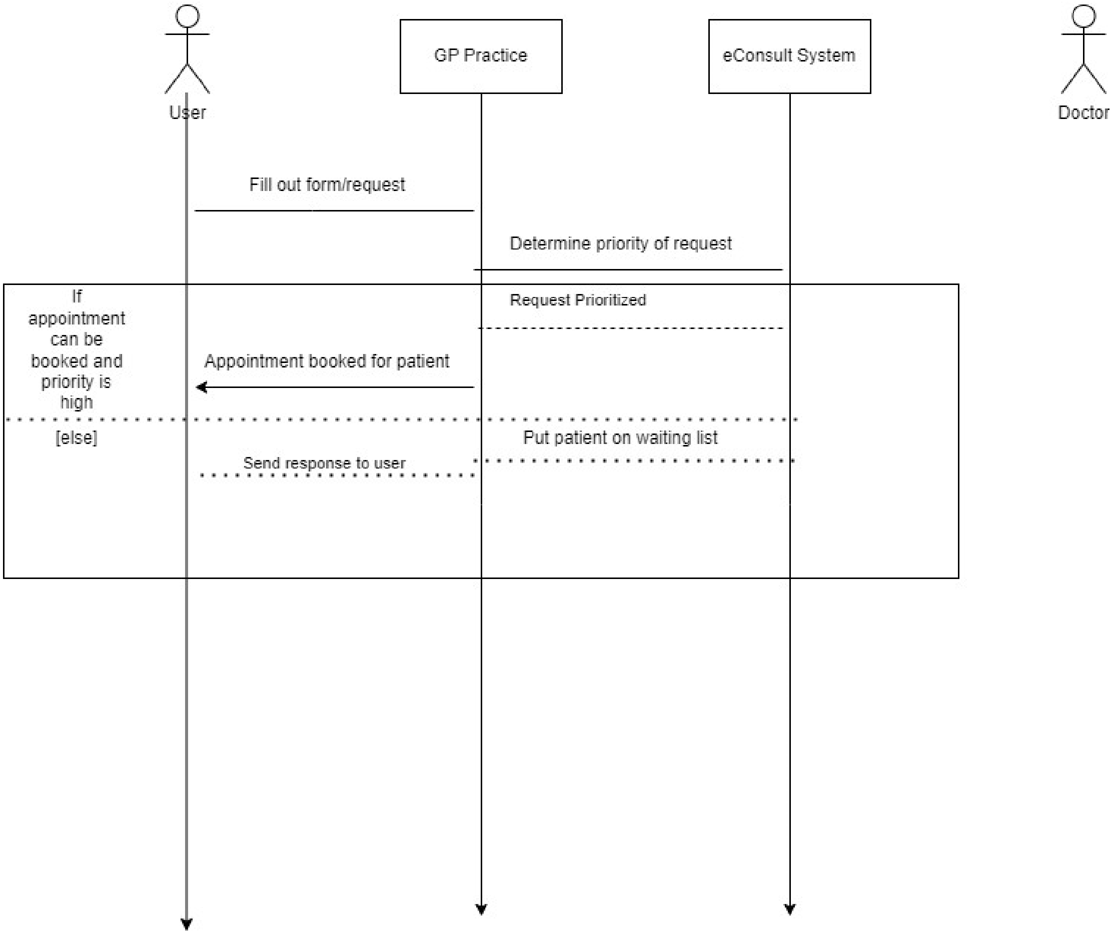
Patient fills out and submits the form to create a request. A status message is returned (e.g. “Submitted successfully”, “Error” etc.)



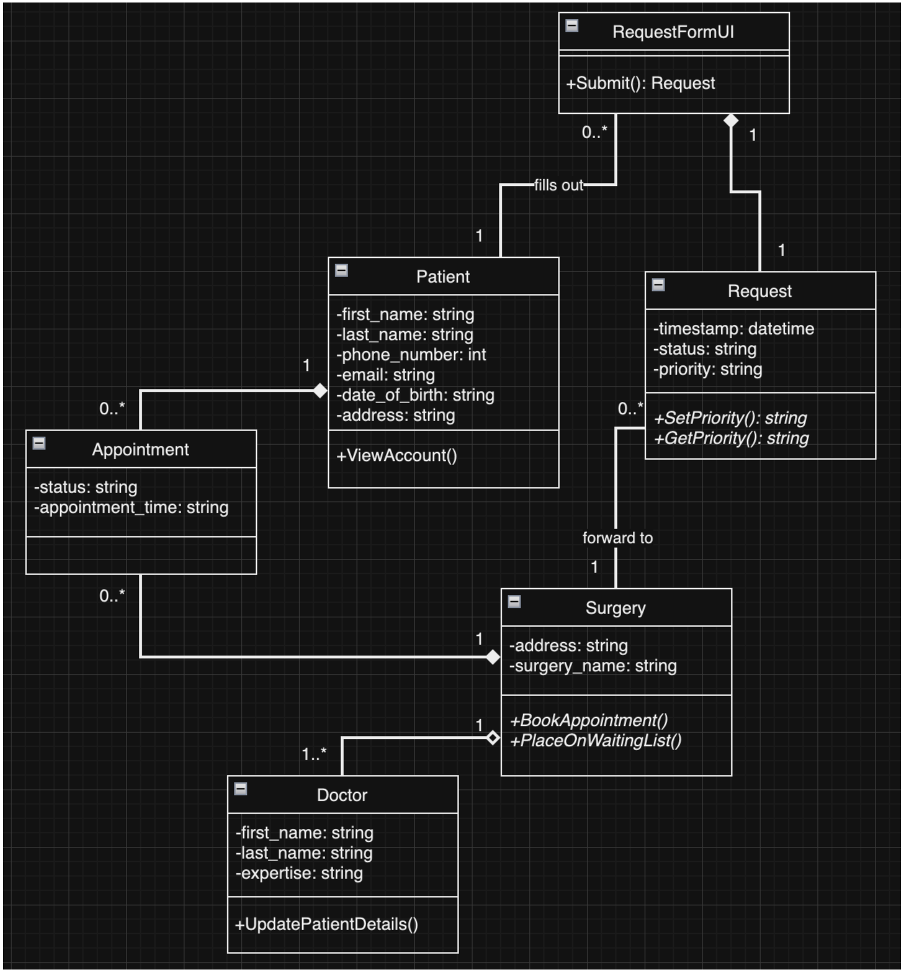

Appointment with Doctor
user
Doctor
GP practice
attends appointment with doctor
updates users personal information if needed
updates medical history
Consult the patient


Viewing patient medical history
user
eConsult System
Gp Practice
View account
Receive information about user
return information
present information to user

User filing a request


User
GP Practice
eConsult System
Doctor
Fill out form/request
Determine priority of request
Request Prioritized
If appointment can be booked and priority is high
Appointment booked for patient
[else]
Put patient on waiting list
Send response to user

UML Design Class Diagram



Database

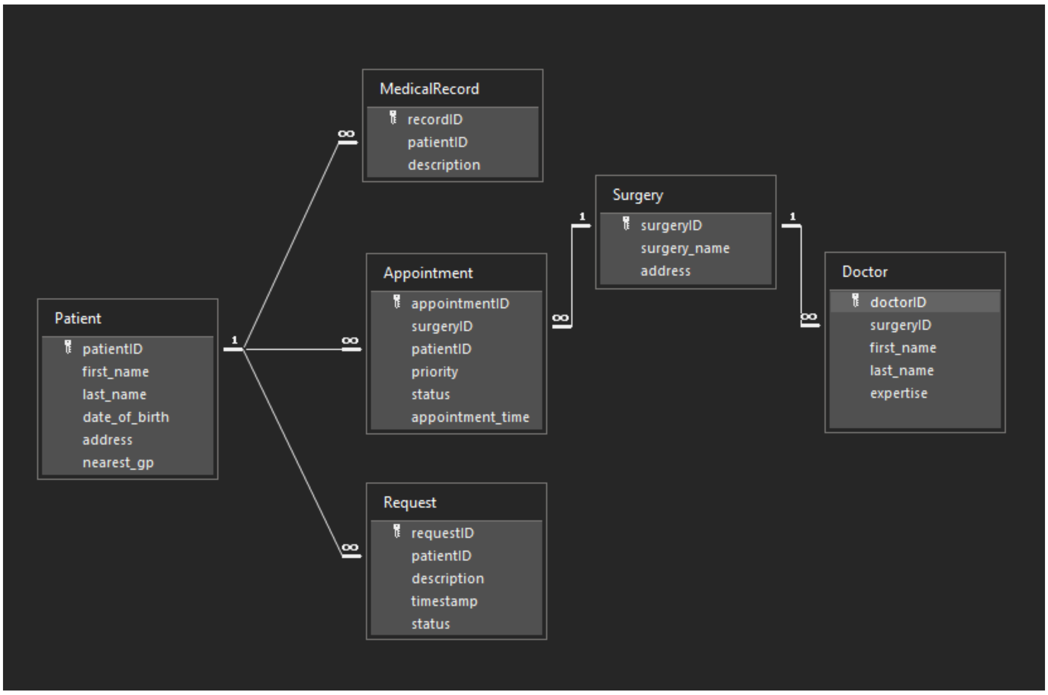
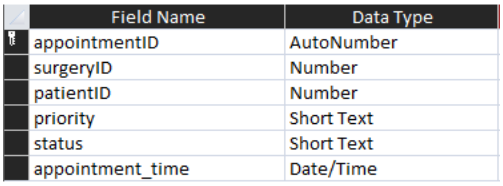
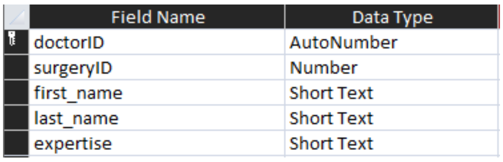


Table Appointment



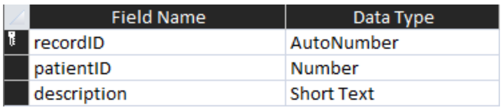
Field Name	Data Type
appointmentID	AutoNumber
surgeryID	Number
patientID	Number
priority	Short Text
status	Short Text
appointment_time	Date/Time

Table Doctor



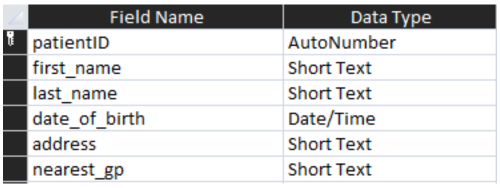
Field Name	Data Type
doctorID	AutoNumber
surgeryID	Number
first_name	Short Text
last_name	Short Text
expertise	Short Text

Table MedicalRecord



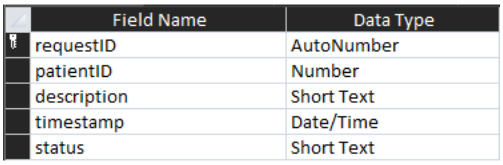
Field Name	Data Type
recordID	AutoNumber
patientID	Number
description	Short Text

Table Patient



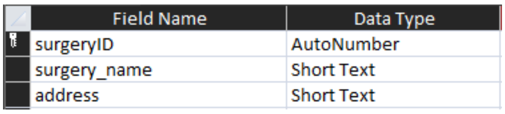
Field Name	Data Type
patientID	AutoNumber
first_name	Short Text
last_name	Short Text
date_of_birth	Date/Time
address	Short Text
nearest_gp	Short Text

Table Request



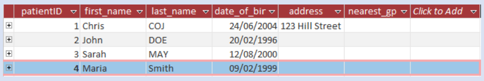
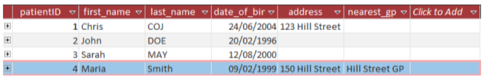
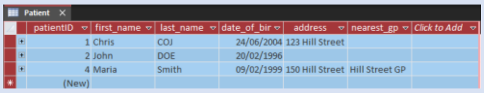
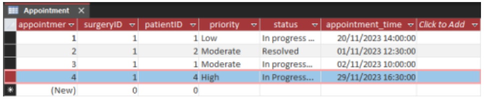
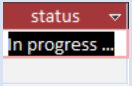
Field Name	Data Type
requestID	AutoNumber
patientID	Number
description	Short Text
timestamp	Date/Time
status	Short Text

Table Surgery

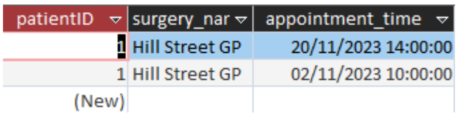


Field Name	Data Type
surgeryID	AutoNumber
surgery_name	Short Text
address	Short Text

SQL queries

<table>
<tr><th>Purpose</th><th>Query</th><th>Result</th></tr>
<tr><td>Add patient</td><td>INSERT INTO Patient(first_name, last_name, date_of_birth)
VALUES ('Maria', 'Smith', '09/02/1999');</td><td>Patient
<table>
<tr><th></th><th>patientID</th><th>first_name</th><th>last_name</th><th>date_of_bir</th><th>address</th><th>nearest_gp</th><th>Click to Add</th></tr>
<tr><td>⊞</td><td>1</td><td>Chris</td><td>COJ</td><td>24/06/2004</td><td>123 Hill Street</td><td></td><td></td></tr>
<tr><td>⊞</td><td>2</td><td>John</td><td>DOE</td><td>20/02/1996</td><td></td><td></td><td></td></tr>
<tr><td>⊞</td><td>3</td><td>Sarah</td><td>MAY</td><td>12/08/2000</td><td></td><td></td><td></td></tr>
<tr><td>⊞</td><td>4</td><td>Maria</td><td>Smith</td><td>09/02/1999</td><td></td><td></td><td></td></tr>
</table></td></tr>
<tr><td>Update patient details</td><td>UPDATE Patient
SET address = '150 Hill Street', nearest_gp = 'Hill Street GP'
WHERE patientID = 4</td><td>Patient
<table>
<tr><th></th><th>patientID</th><th>first_name</th><th>last_name</th><th>date_of_bir</th><th>address</th><th>nearest_gp</th><th>Click to Add</th></tr>
<tr><td>⊞</td><td>1</td><td>Chris</td><td>COJ</td><td>24/06/2004</td><td>123 Hill Street</td><td></td><td></td></tr>
<tr><td>⊞</td><td>2</td><td>John</td><td>DOE</td><td>20/02/1996</td><td></td><td></td><td></td></tr>
<tr><td>⊞</td><td>3</td><td>Sarah</td><td>MAY</td><td>12/08/2000</td><td></td><td></td><td></td></tr>
<tr><td>⊞</td><td>4</td><td>Maria</td><td>Smith</td><td>09/02/1999</td><td>150 Hill Street</td><td>Hill Street GP</td><td></td></tr>
</table></td></tr>
<tr><td>Delete patient details</td><td>DELETE FROM Patient
WHERE patientID = 3;</td><td>Patient
<table>
<tr><th></th><th>patientID</th><th>first_name</th><th>last_name</th><th>date_of_bir</th><th>address</th><th>nearest_gp</th><th>Click to Add</th></tr>
<tr><td>⊞</td><td>1</td><td>Chris</td><td>COJ</td><td>24/06/2004</td><td>123 Hill Street</td><td></td><td></td></tr>
<tr><td>⊞</td><td>2</td><td>John</td><td>DOE</td><td>20/02/1996</td><td></td><td></td><td></td></tr>
<tr><td>⊞</td><td>4</td><td>Maria</td><td>Smith</td><td>09/02/1999</td><td>150 Hill Street</td><td>Hill Street GP</td><td></td></tr>
<tr><td>*</td><td>(New)</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table></td></tr>
<tr><td>Add an appointment in the system</td><td>INSERT INTO Appointment(surgeryID, patientID, priority, status, appointment_time)
VALUES(1, 4, 'High', 'In Progress...', '29/11/2023 16:30:00');</td><td>Appointment
<table>
<tr><th>appointmer</th><th>surgeryID</th><th>patientID</th><th>priority</th><th>status</th><th>appointment_time</th><th>Click to Add</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Low</td><td>In progress ...</td><td>20/11/2023 14:00:00</td><td></td></tr>
<tr><td>2</td><td>1</td><td>2</td><td>Moderate</td><td>Resolved</td><td>01/11/2023 12:30:00</td><td></td></tr>
<tr><td>3</td><td>1</td><td>1</td><td>Moderate</td><td>In progress...</td><td>02/11/2023 10:00:00</td><td></td></tr>
<tr><td>4</td><td>1</td><td>4</td><td>High</td><td>In Progress...</td><td>29/11/2023 16:30:00</td><td></td></tr>
<tr><td>(New)</td><td>0</td><td>0</td><td></td><td></td><td></td><td></td></tr>
</table></td></tr>
<tr><td>View the status of an appointment</td><td>SELECT status
FROM Appointment
WHERE appointmentID=1;</td><td>Appointment
<table>
<tr><th>status</th></tr>
<tr><td>In progress ...</td></tr>
</table></td></tr>
<tr><td>View the history of all the appointments for a particular patient including patient ID, surgery name and appointment time</td><td>SELECT Patient.patientID, Surgery.surgery_name, Appointment.appointment_time

FROM ((Appointment INNER JOIN Patient ON Appointment.patientID=Patient.patientID) INNER JOIN Surgery ON Appointment.surgeryID=Surgery.surgeryID)

WHERE Patient.patientID = 1;</td><td>Appointment, Patient, Surgery
<table>
<tr><th>patientID</th><th>surgery_nar</th><th>appointment_time</th></tr>
<tr><td>1</td><td>Hill Street GP</td><td>20/11/2023 14:00:00</td></tr>
<tr><td>1</td><td>Hill Street GP</td><td>02/11/2023 10:00:00</td></tr>
<tr><td>(New)</td><td></td><td></td></tr>
</table></td></tr>
</table>

References:

[1] Taherdoost (2015) “How to Lead to Sustainable and Successful IT Project Management? Propose 5Ps Guideline.” Advanced Informatics School, Universiti Teknologi Malaysia

[2] PMBOK (2008). "A Guide to the Project Management Body of Knowledge, 4th ed.." Project Management Institute.

[3] Brede Moe (2010) “Information of Software Technology. "Volume 52, Issue 5, Pg 480-491

Marks mapping:

80-100 Exceptional	**70-79 Excellent**	**60-69 Very Good**	**50-59 Good**	**40-49 Satisfactory**	**0-39 Fail**
A	**B**	**C**	**D**	**E**	**F**

	A	B	C	D	E	F	Comments
the 5 Ps 1. Problem:			X				We showed good knowledge of the 5Ps and we explained each one clearly whilst showing evidence to back it up. It is also all relevant to the case study whilst outlining each problem with the 5Ps.
2. Process		X					
3. Project			X				

1. Product			X				
2. People		X					
Functional and non-functional requirements:		X					We explained each functional and non-functional requirement with detail outlining all the advantages and disadvantages related to the case study.

Structured design, 1. Entity Relationship Diagram 2. Data Flow Diagrams. 3. SQL queries	X	X	X				We made each diagram detailed and explained the links between them. We made sure the diagrams were linked to the case study and make it easy to understand for the reader.

UML Design Use Case Diagram			X				With these diagrams we made them very clear and easy to understand.
Design Class diagrams		X					The design class diagram helped understand each object’s relation with one another and the different classes within them.
Sequence Diagrams							Clear sequence diagrams showing the flow of the messages from one object to another.

Group members work contribution form:

Group/Team Name: 2

Team member name	Student ID	individual overall work contribution (%)	Signature
Student1: Catalin Cojocaru	001305051	25%	
Student2: Nabir Malik	001355889	25%	
Student3: Hillary Jackson	001351434	25%	HJackson
Student4: Jamal Akanni	001327518	25%	
		Total 100%	

Describe each task you performed (For example the creation of a Use case diagram)	Student 1: Contribution	Student 2: Contribution	Student 3: Hillary Jackson Contribution	Student 4: Contribution
Task: 5Ps of Software Engineering			90%	
Task: Functional and Non-Functional requirements			80%	
Task: DFD, ERD and SQL			75%%	
Task: Use Case diagram creation			60%	
Task: Design Class Diagram and Sequence Diagram			40%	