Candidate number:
3097

Investigation:

2.1: Methods of investigation

To effectively try to come up with a programmed solution to the client's problem I must first understand their current system and the problems with it. To do this I have come up with four methods to investigate these problems. These are:

1. An interview with the head of the company. I will ask them direct questions I have on the program and try to find out any of the problems they are facing with the current system. I will try to find out the day to day usage of the system. I aim to find out the issues that I should focus on when developing my new program for the clinic. I will also find out how data is distributed around the company and any of the checks they currently have to make sure people who shouldn't have access to certain pieces of information don't. The benefit of an interview is more detailed and specific answers to specific questions that may not be answered through the other methods that I use
2. I intend on carrying out a survey of all the staff members at the clinic, I intend to ask them on a scale of strongly agree to strongly agree how they feel about various parts of the current system and the idea of the new system as well as for some any possible complaints or suggestions at the end. This should be quick and easy for the staff and shouldn't take too much of their time
3. I will visit the clinic to get a hands-on point of view on how an average day there works. Doing this means I will be able to notice problems with their current system that ordinary staff my not notice
4. I will ask for examples of their current system and copies of what they use. For example, how they store client details and appointments and try see the problems with their system directly

Candidate number:
3097

2.2: Interview

How do you store your records?

We have paper records with booklets for different type of data. These books are stored at reception. We hold paper records for our appointments, finances and all the information about the staff members working at our clinic.

How do you add a new client to you records?

We don't add records for client by themselves, instead we write them in our appointment's booklet where beside the appointment details is the clients name and phone number. It would extremely helpful if we could hold these client records by themselves.

How do you keep the records of staff members?

We have a booklet of all the staff members with their full name, address, postcode and their telephone number. Beside that we also have their salary. The big issue with this list is it has problems adapting to staff members quitting or having some of their information changed as it requires all the information to be taken down again.

How do you create appointments?

When a client wants to create an appointment, they would go over to reception and request one. When an appointment is created a book is opened and a new line is created with the client's name and telephone number as well as the date and time of the appointment. Beside this is the name of the staff member taking that appointment.

How do you look for a certain appointment by date?

It's quite difficult to as the appointment booklet is quite muddled and not in order as not every client is available in each appointment bracket. Usually when a client wants to find the date of their appointment we ask for their name and we search for their name in the booklet, whatever date is beside that name is what we tell them, this has had issues in the past however when a client has had multiple appointments and thus some may have been skipped over.

How do you cancel an appointment?

We cross it out with a pen, this does deal with a quite messy booklet however and makes it much more confusing to see if an appointment is available at a certain date or time. This is an issue many of the staff members have told me about and they would like a much easier method of being able to cancel appointments.

How do you search through your records?

We can't really as there is no set structure of our records, due to this it can take quite a while to find a certain record. This brings finding a record more down to luck then a set structure. It would be ideal if this could be improved in the new database.

Candidate number:
3097

How do you store your finances?

We have a booklet of each finance with its purpose, date and time and amount. The money stored is then put into a safe. When salaries go out I look at the staff booklet and take out the respective amount out of the safe and distribute it among the workers.

Can you see your overall profits per day?

We cannot, we get lots of transactions per day and it would take too long to calculate the overall profits. This also has a large room for error.

How do you store your medical records?

We can't. Due to the sensitivity of the data that we hold, we do not feel that it would be a good decision to hold our medical records in a booklet that any of our staff members could access and has little data security. Instead we have our clients tell us during our appointment or our nurses and physiotherapists remember.

How do you prevent invalid data being added?

We currently have no system in place to add invalid data being added, this has resulted in problems before such as a telephone number of an invalid length which we could not validate automatically. Most of our checks are done by staff but due to human error there are lots of issues with invalid information being added into our program. This issue is most severe at our transactions where incorrect information could have severe consequences for the staff member and our clinic. There are also issues with bad handwriting which can result in some pieces of information being hard to understand.

How do you only allow certain people to have access to certain information?

This is a severe issue with our current system, as we hold all our data in booklets these can be easily stolen and looked at by members of staff who do not have access to it. We currently have CCTV in place, but the CCTV does not automatically check if people are checking information that they shouldn't.

How long are records stored for?

The records are stored for as long as the booklet lasts. After this the booklet is either held in a safe place or thrown away. The booklets held would be transactions and staff whereas we throw away our appointments booklet as we have no longer have a use for it and we don't want to hold potentially sensitive data.

Are there links between records in one booklet and one in another?

Not really. When an appointment is over, it is ticked off and a transaction is added but this has flaws, for example not being automatic and it being hard to find out which appointment belongs to which transaction as it doesn't state when the appointment was paid for in the appointments table.

Candidate number:
3097

How do you remove a record?

We typically cross it out of the booklets, however this has issues with holding data of clients who don't want their information held and can break the GPDR. And issue may be that it is not fully crossed out and it may still be able to be read. However, we cannot remove the booklet or even the page as other vital information on other records should still be held.

How do you edit a record?

We have to remove the old record and create a new one with the new information. This can be a pain for staff members as this method is highly inefficient and ineffective for the staff members and can result in lots of time being wasted. This can also be hard if the old information of that record is hard to read.

How do you share a copy of the same record to other members of staff?

We have to physically give the booklet to them as we have no extra copies to the records and multiple staff members cannot have access to the same record at the same time as you may be able to within a programmed database where multiple people would be able to login and view the file.

What are the issues with your current system?

There is a wide range of issues with our current system. We cannot store client information separately, we are unable to hold medical records, we have no significant security in place to prevent people from accessing and manipulating information they shouldn't have, we have no clean and effective system in place for cancelling records. We have no system to view all records in a neat fashion and search for them based on a specified criteria. We have no automatic links between tables. We aren't able to edit or add records effectively and we have no validation within our system. We are unable to remove a record in a neat fashion. We are unable to get outputs such as the total profits on a specific day or the BMI of a client.

What improvements are you looking for in the new system?

In the new system I'm looking for a program which implements views and has little data redundancy, the system should be in third normalised form. It should be easy to search, edit, view, delete and cancel records and the search should output to a text file and within the program itself. There should be outputs for transactions on a specified date and a client's BMI. In addition, all tables should be linked and update each other automatically. There should be a system of login details so non-staff members cannot access the system.

Candidate number:
3097

Analysis of the interview:

It is clear from the interview that the current system needs a major overhaul. The main issues brought up involved severe issues with data recumbency and access to data sensitivity. As a result, I will focus on these on the development of my program. I also aim to make my program very intuitive and easy to use. From this interview it is clear that there are severe issues with their data security and an overhaul of their current system is necessary. Their current system seems to create a significant amount of issue throughout the company and I believe that a large number of areas at their company needs to be immediately improved. Views are also non-existent in their current system meaning that a staff member could easily access sensitive data and as a result of this lack of data security the company is unable to store information on medical records which forces the staff members at the clinic to either remember the client's medical records or ask them what they are which could result in severe issues if a medical record, like a blood type is remembered incorrectly. Overall, I believe a computer-based system in replacement of their old one is not only important, but necessary and a variety of features and new data security measures must be implemented into their new system to significantly reduce the problems that this clinic faces.

2.3: Survey

Results

I then did a survey with the staff at the clinic. A survey is effective as it is quick and easy to do and gives me a strong indication of how the people at the clinic feel about the new program. I will use the information from this survey to see what to focus on when developing the program. I gave 9 copies of the survey to the staff there. This is the survey that I gave:

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is					
The system should be updated					
You would find a newer system useful					
Data should be held more securely					
Any information held should be backed up					
The current system needs better methods of outputting information					
The current system needs better methods of inputting information					
The current system is slow/inefficient					
There should be more tables held					

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is					✓
The system should be updated	✓				
You would find a newer system useful	✓				
Data should be held more securely	✓				
Any information held should be backed up	✓				
The current system needs better methods of outputting information	✓				
The current system needs better methods of inputting information	✓				
The current system is slow/inefficient	✓				
There should be more tables held	✓				

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is			✓		
The system should be updated				✓	
You would find a newer system useful		✓			
Data should be held more securely		✓			
Any information held should be backed up		✓			
The current system needs better methods of outputting information					✓
The current system needs better methods of inputting information					✓
The current system is slow/inefficient			✓		
There should be more tables held			✓		

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is	✓				
The system should be updated				✓	
You would find a newer system useful				✓	
Data should be held more securely		✓			
Any information held should be backed up		✓			
The current system needs better methods of outputting information				✓	
The current system needs better methods of inputting information				✓	
The current system is slow/inefficient		✓			
There should be more tables held				✓	

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is		✓			
The system should be updated				✓	
You would find a newer system useful				✓	
Data should be held more securely		✓			
Any information held should be backed up		✓			
The current system needs better methods of outputting information				✓	
The current system needs better methods of inputting information				✓	
The current system is slow/inefficient			✓		
There should be more tables held	✓				

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is					✓
The system should be updated	✓	✓			
You would find a newer system useful	✓	✓			
Data should be held more securely	✓	✓			
Any information held should be backed up	✓	✓			
The current system needs better methods of outputting information		✓		✓	
The current system needs better methods of inputting information			✓		
The current system is slow/inefficient	✓	✓			
There should be more tables held	✓				

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is				X	
The system should be updated		X			
You would find a newer system useful		X			
Data should be held more securely		X			
Any information held should be backed up		X			
The current system needs better methods of outputting information		X			
The current system needs better methods of inputting information		X			
The current system is slow/inefficient				X	
There should be more tables held		X			

Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is			✓		
The system should be updated			✓		
You would find a newer system useful			✓		
Data should be held more securely		✓			
Any information held should be backed up		✓			
The current system needs better methods of outputting information				✓	
The current system needs better methods of inputting information		✓			
The current system is slow/inefficient		✓			
There should be more tables held		✓			

Candidate number:
3097

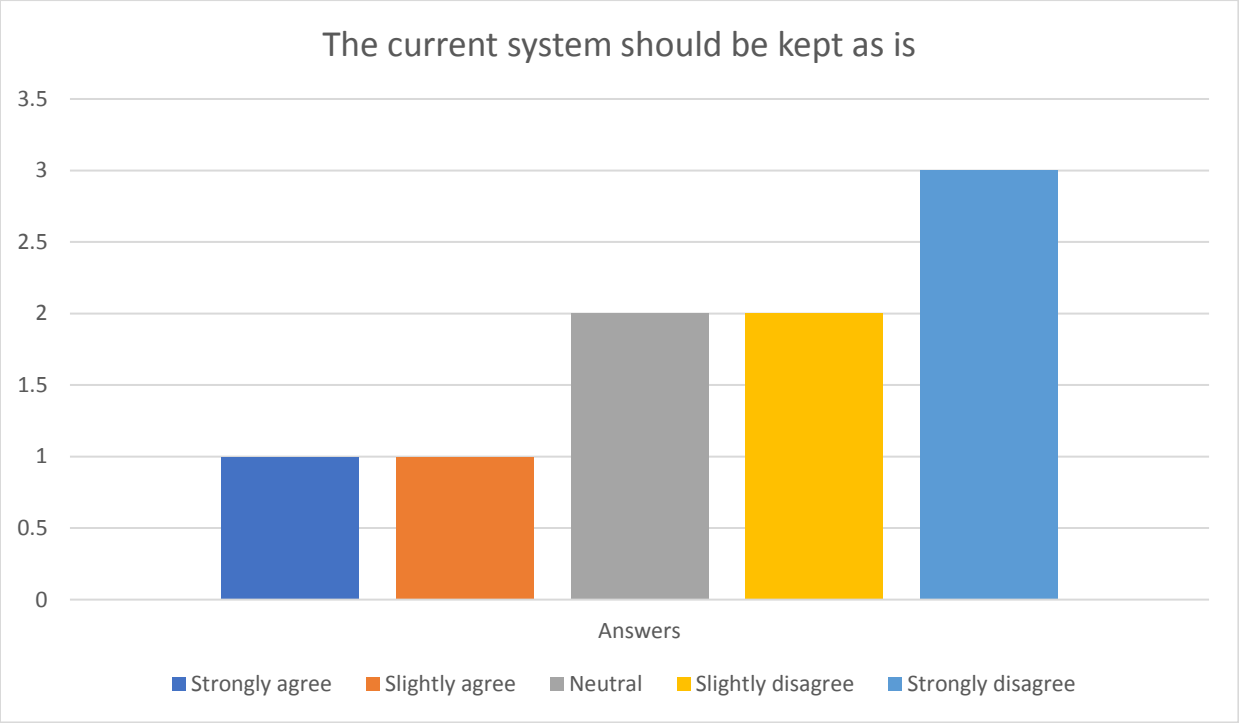
	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is					✓
The system should be updated	✓		✓		
You would find a newer system useful		✓			
Data should be held more securely	✓				
Any information held should be backed up	✓				
The current system needs better methods of outputting information	✓	✓			
The current system needs better methods of inputting information	✓	✓			
The current system is slow/inefficient	✓	✓			
There should be more tables held	✓				

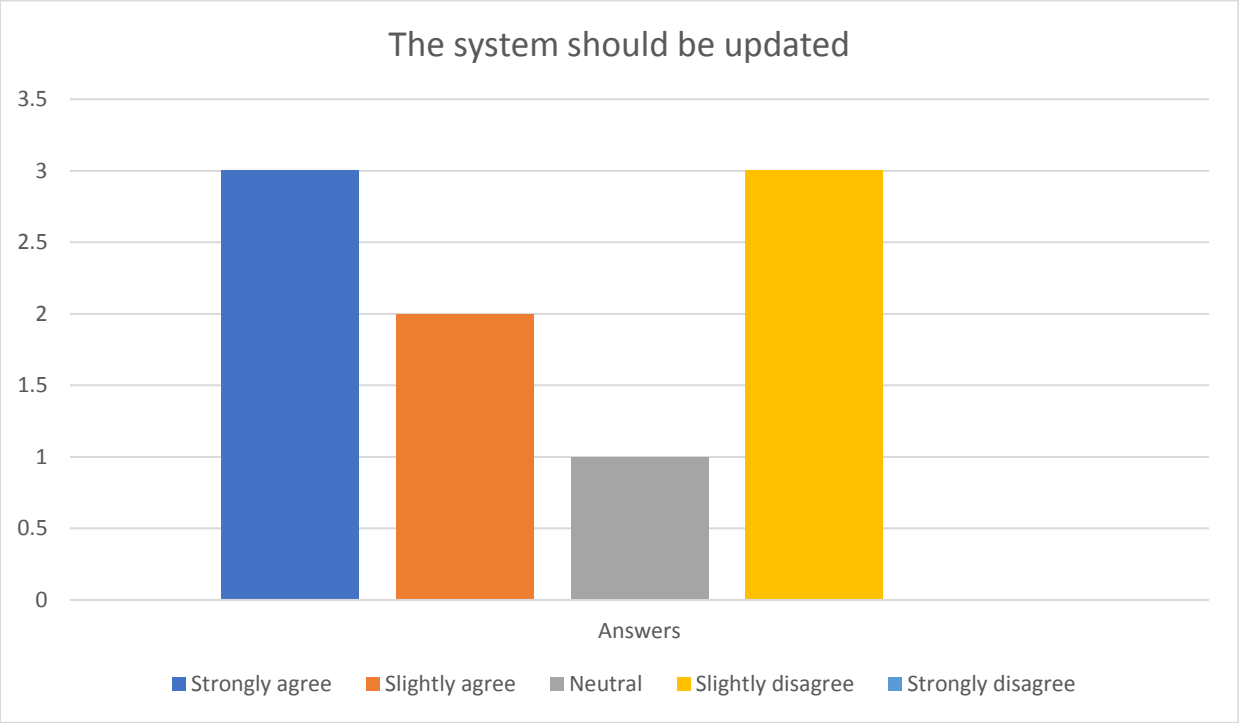
Candidate number:
3097

	Strongly agree	Slightly agree	Neutral	Slightly disagree	Strongly disagree
The current system should be kept as is				✓	
The system should be updated		✓			
You would find a newer system useful		✓			
Data should be held more securely		✓			
Any information held should be backed up		✓			
The current system needs better methods of outputting information		✓			
The current system needs better methods of inputting information			✓		
The current system is slow/inefficient				✓	
There should be more tables held		✓			

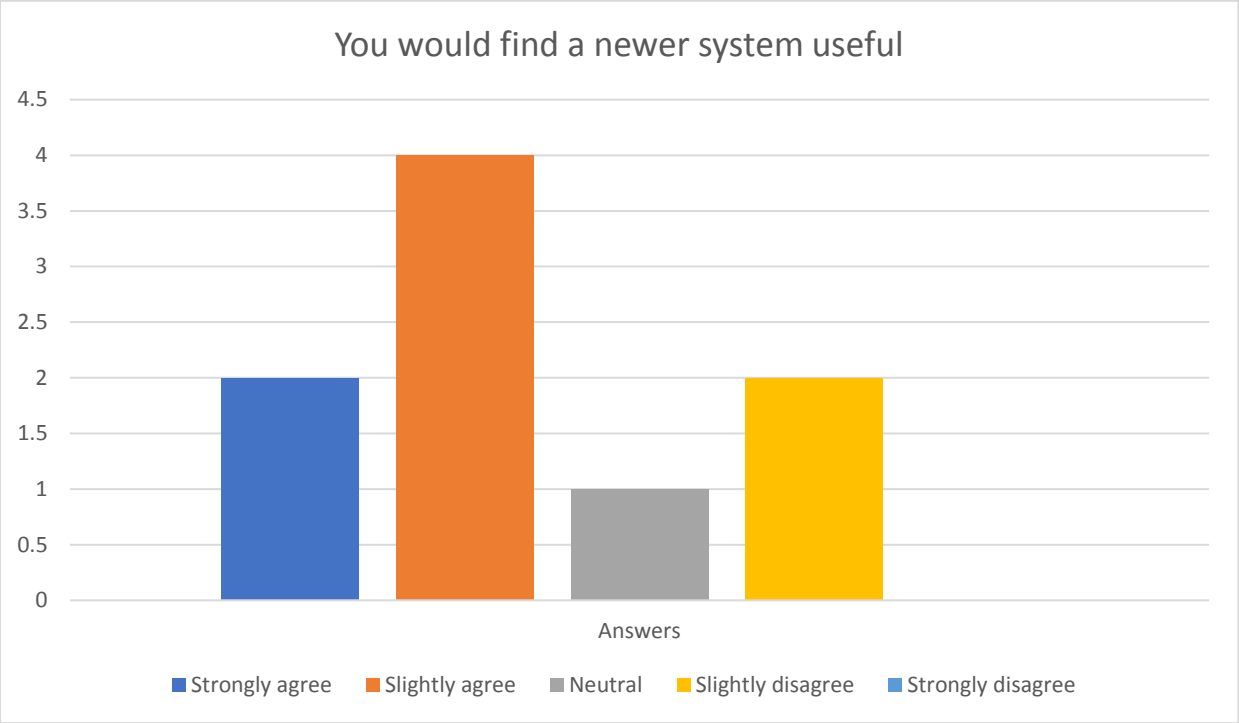
Candidate number:
3097

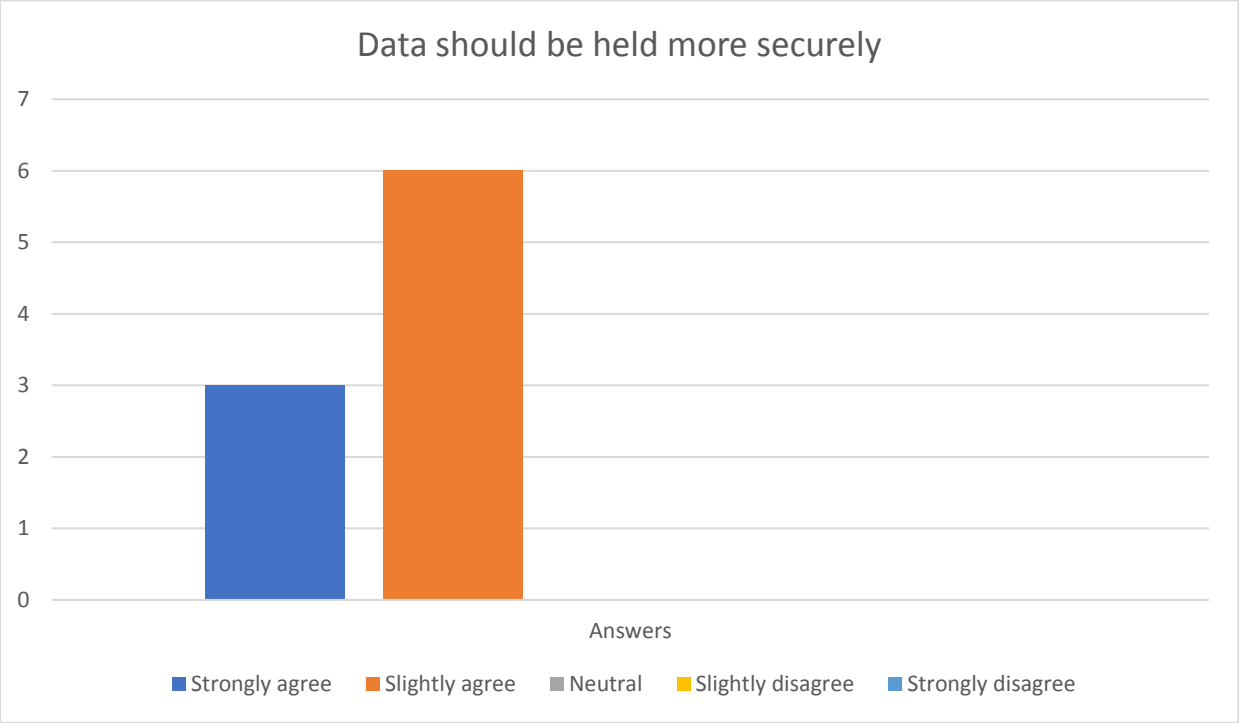
I then converted all the replies to the data into tables to allow a more in-depth analysis :



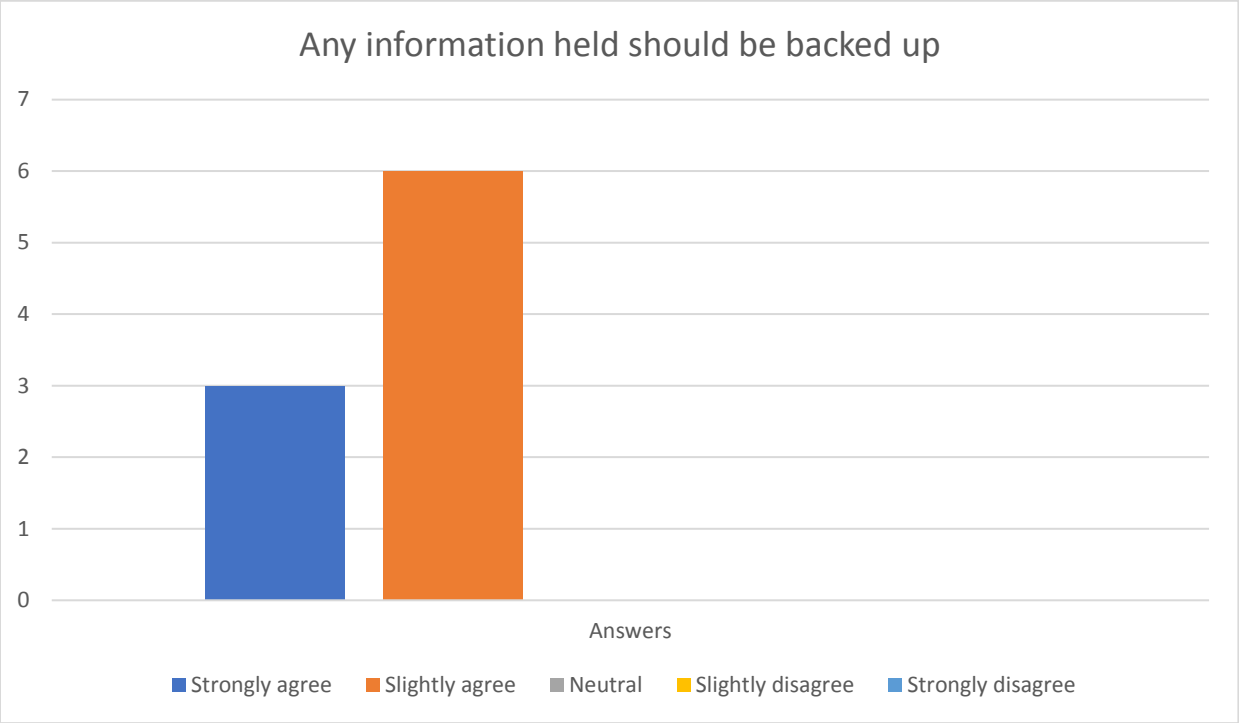


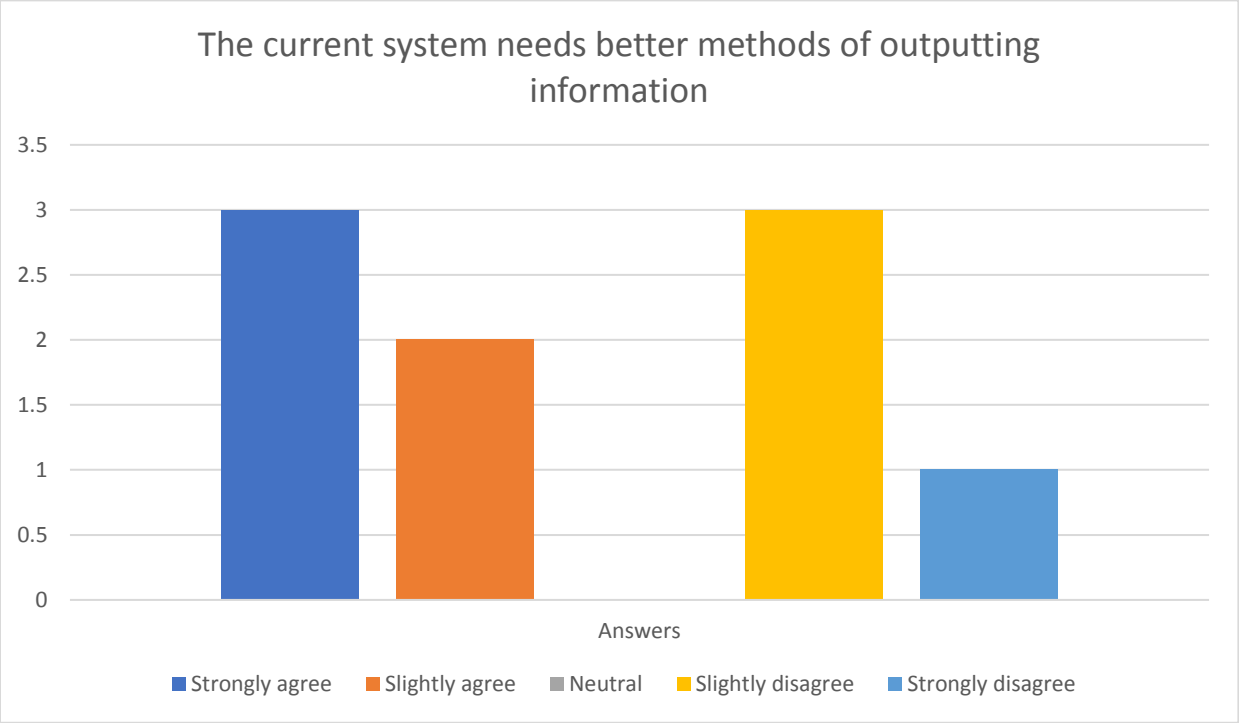
Candidate number:
3097



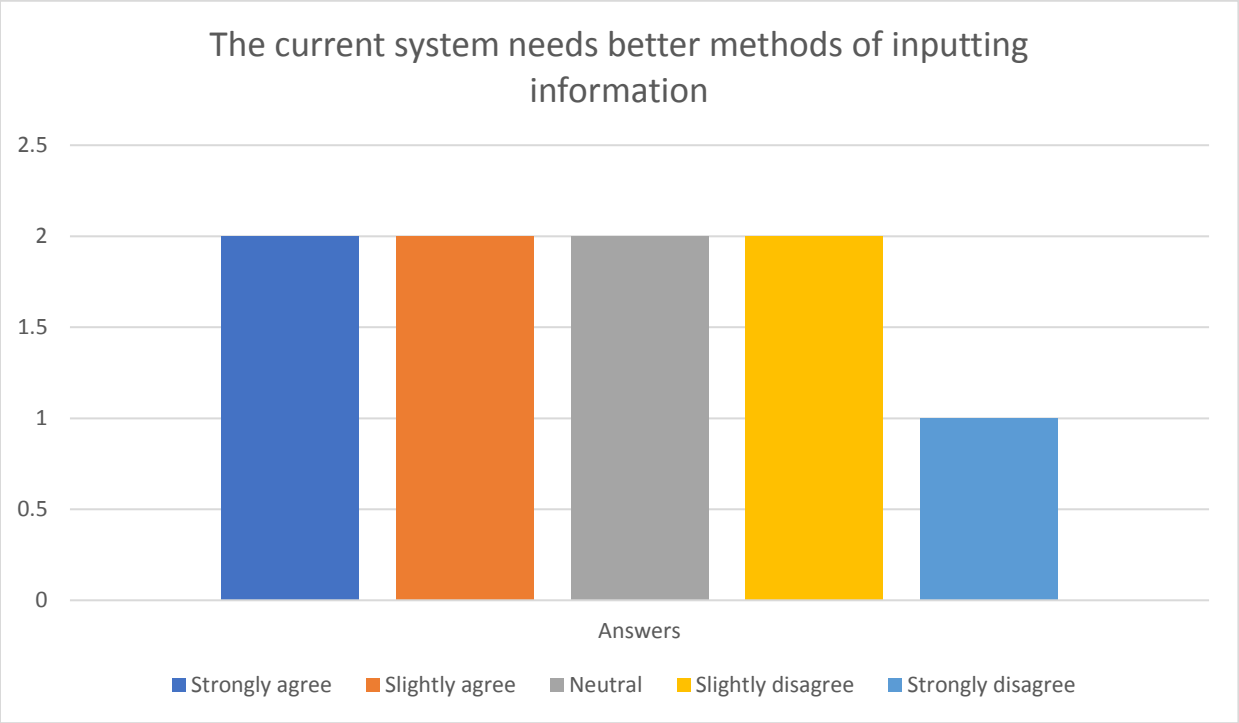


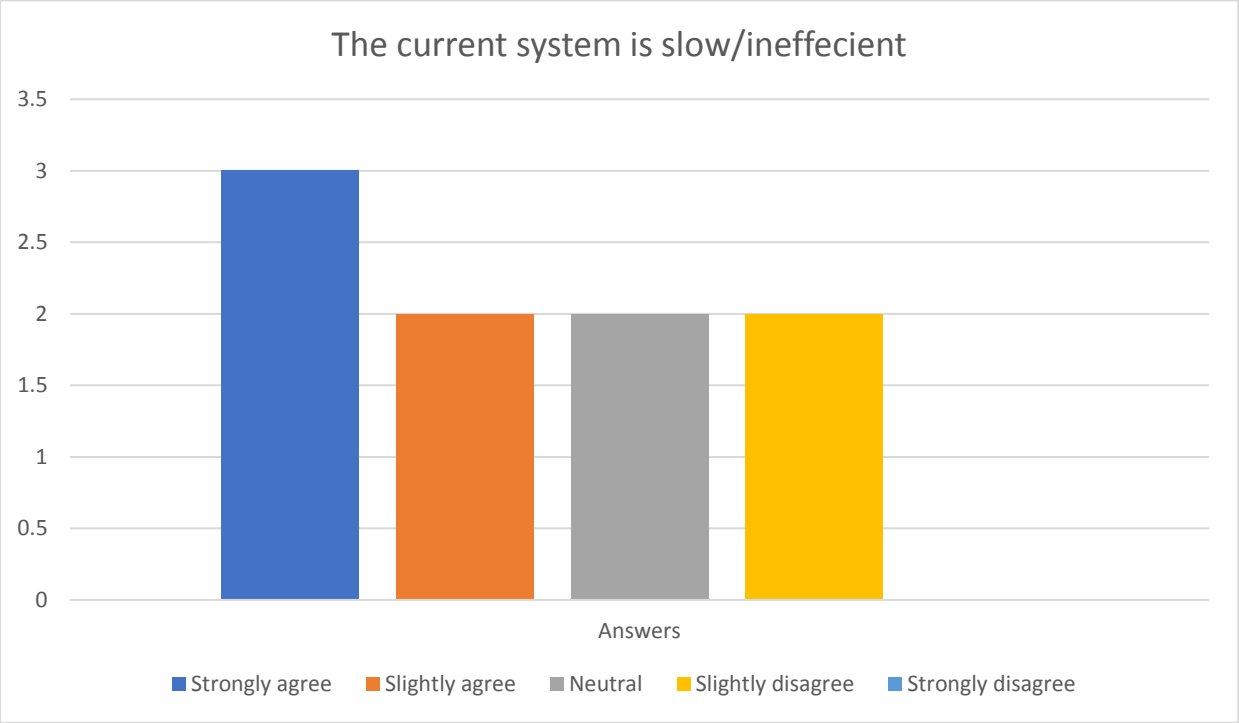
Candidate number:
3097



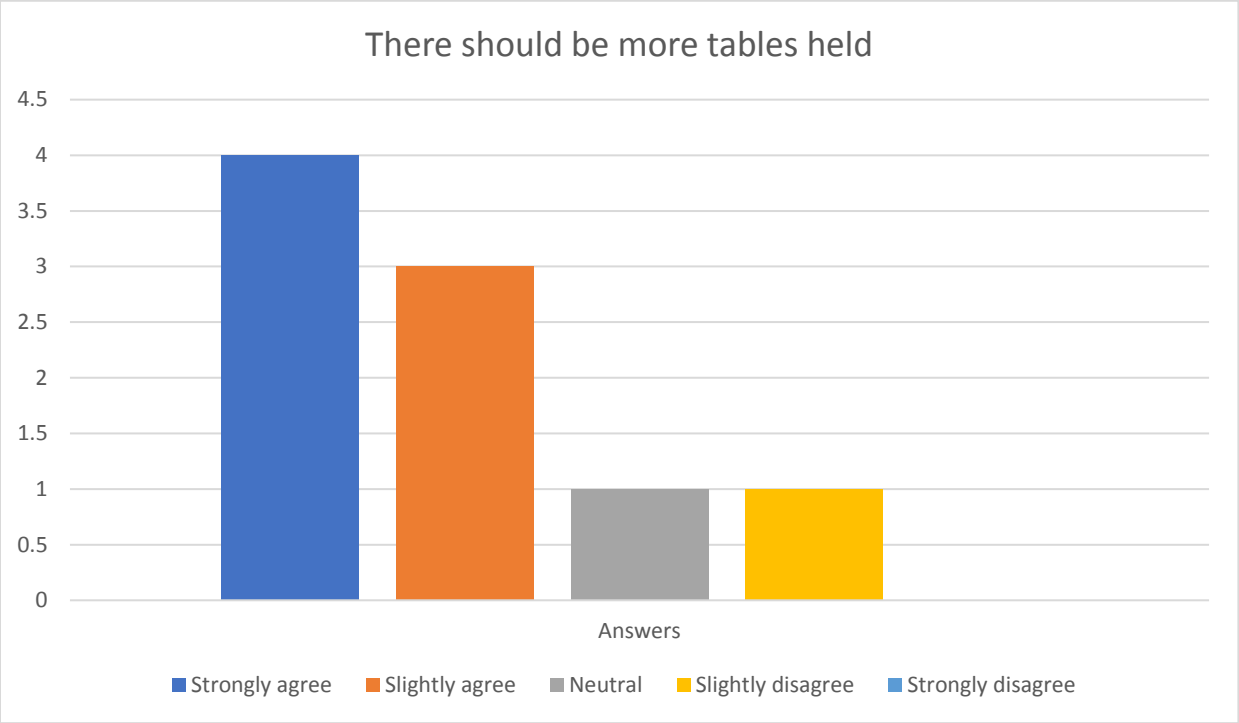


Candidate number:
3097





Candidate number:
3097



Analysis

The current system should be kept as is

There is a clear belief that the system should not be kept as is which you can see from the tables. The staff members want to improve their system and feel like the current one is inefficient which you can see from the tables.

The system should be updated

The general trend is that the current system the staff have should be updated. This shows that the staff members really feel the system should be changed.

You would find a newer system useful

Most of the staff members agree that a newer system would be very useful. This is highly important for the clinic as their old system is starting to break and a newer system could make life significantly easier for the staff members that work at the clinic.

Candidate number:
3097

Data should be held more securely

People clearly want data to be secured. As a result of this I will implement suitable login details and views within my program as this will limit staff members from accessing tables that they shouldn't be allowed to have access to and instead only limit them to the levels of functionality and access that their position at the company allows them to. This will, for example limit receptionists from accessing the medical records of clients and instead only allow nurses, receptionists and the owner of the company to have access to such sensitive information, the login system also means that the database is locked off to anyone who doesn't have suitable details to login.

Any information held should be backed up

Staff members want the data to be backed up and it is clear that data being backed up is important. However, I still believe that due to the limit time given on the program I will be unable to create a backup.

The current system needs better methods of outputting information

Most people are unsure if the new system should have better methods of outputting information. However, in the interview there was a clear emphasis on new methods of outputs being vital.

The current system needs better methods of inputting information

Similar to outputting information, most people are unsure of inputting information. Once again, in the interview the owner requested new methods of inputting data.

The current system is slow/inefficient

There is a slight trend that the current system is slow. This is likely due to the paper system and its problems. For example, if you were to edit in a paper system it would be very inefficient.

There should be more tables held

There is a strong belief more tables should be held, as a result I will include a clients and medical record table which is not currently in the paper system.

Overall

It is clear the majority of the staff members working at the company would like an overhaul of their current system and nearly everyone believed that data should be held more securely is an important specification point for my project. The majority of people on this system believe that their current system inefficient and needs serious overhauls. It is also clear that a lot more features should be added into the program, such as new tables and that the significant majority of staff members at the company believe that the system should be updated and that they would find that this new update would be useful for the company.

2.4: Observation

To understand how the company worked more effectively, I attended the clinic to see how it functions. I called in reception and asked them if I could attend on the day. I called in on the day, so staff wouldn't know I was coming and wouldn't act differently to how they would usually.

Initially I went to the reception, it was clear that the staff were having trouble creating appointments for all the clients and finding the appointments for each one. It was clear there were issues in taking and searching for information within their paper records. As a result, I plan to focus on an ease of use on adding and searching for information within my programmed solution.

I then when to the reception desk and noticed that all the booklets containing records were out in the open and that anyone could access them. This is a severe security result and it is clear that within my program I need to create a login page and suitable views to prevent people having access to information that they shouldn't.

I also noticed that multiple staff members requested the appointments records and as they only had one booklet many staff members had to wait a considerable amount of time to use this. With my programmed solution multiple people will be able to use the program at the same time.

A client then came up and told reception that they are unable to have an appointment at their next date and time and asked to reschedule, I noticed that reception had difficulty finding the last appointment that they had, and it showed that the clinic had a severe inefficiency with their current system.

I then checked them using financial records at the end of the day and it was clear that there was lots of room for error with using their paper system, this is because is extremely easy to miscalculate total amounts. I noticed that because of this the results would have to be calculated multiple times which could result in time being lost on more productive things in the company.

Finally, I looked over at the owner using the staff records to give out each of the staff members salary, it was clear that she found it difficult to read some of the results and some of the writing within the paper-based system was illegible, this resulted in her having to spend a significant amount of time trying to decipher the records to find out how much each of the staff members were owed which I believe is a large problem with their current system.

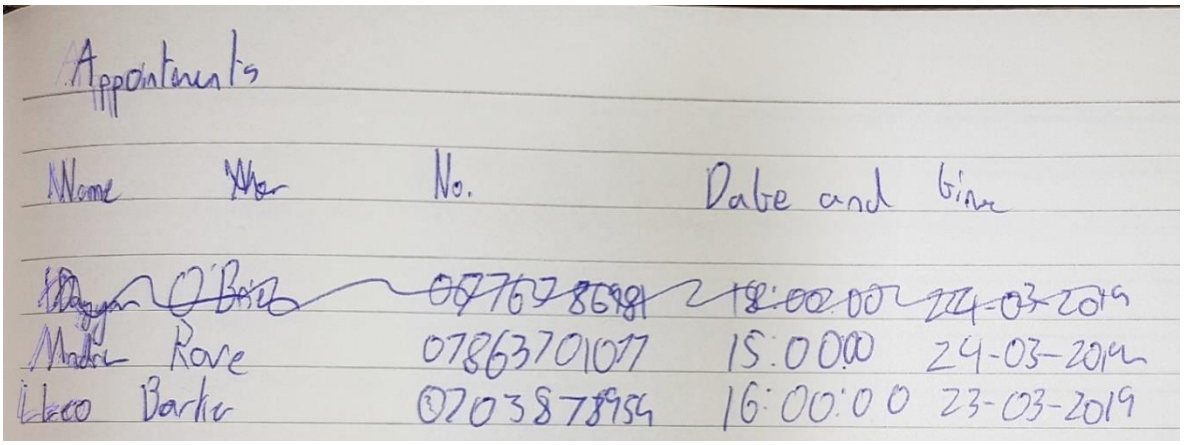
Overall, it is clear that there are severe issues with the use of the system at the clinic, there are many areas which can be significantly improved to new improvements, such as, mathematic calculations for transactions, the ability to instantly search and edit records and the legibility of the database in a programmed solution in comparison to a paper based one.

Candidate number:
3097

2.5: Documentation

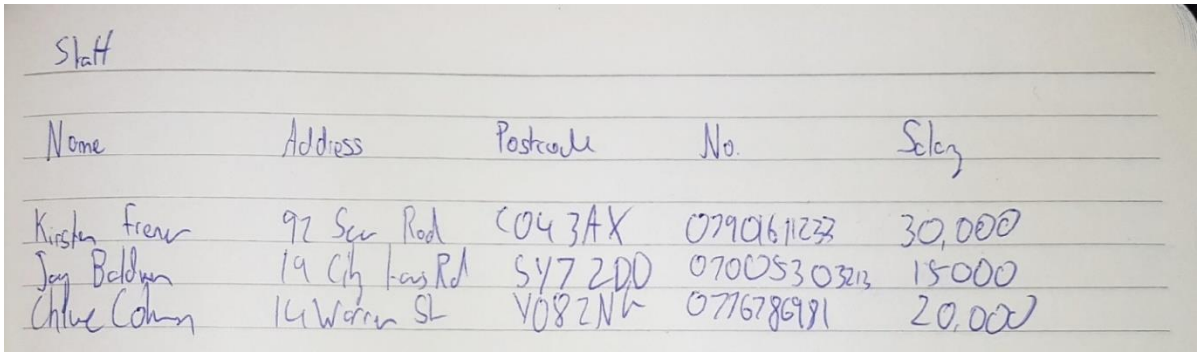
Documentation

I then asked to look at some of their documentation, this was difficult as they did not want to release sensitive information about the company however in the end they complied. It was clear from the documentation that they did not have enough information held and there were lots of scribbles and errors making it hard to read. Below you can see some of the examples of their documentation:



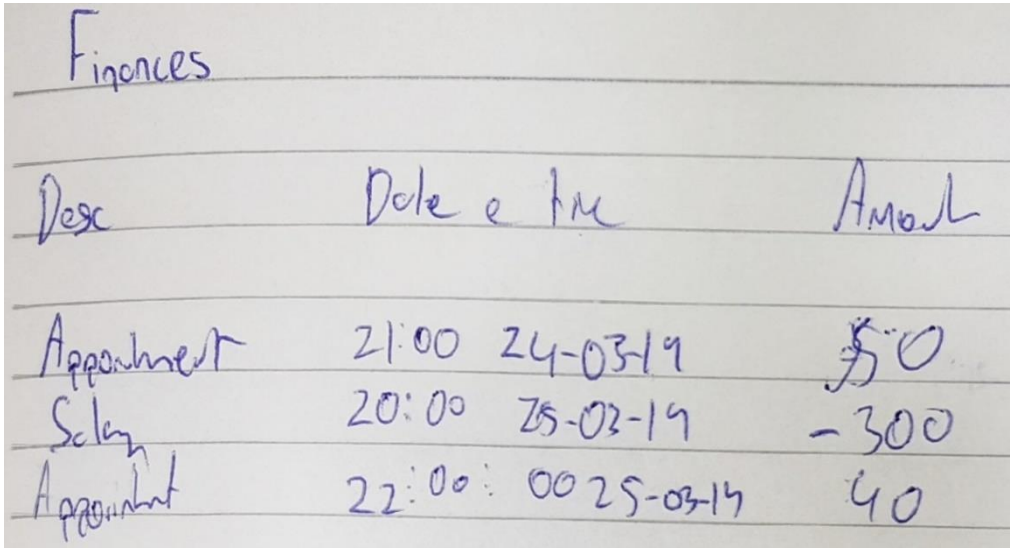
Appointments

Name	No.	Date and time
~~[illegible] O'Brien~~	~~07767869812~~	~~18:00:00 24-03-2019~~
[illegible] Rowe	07863701077	15:00:00 24-03-2019
[illegible] Barker	07038789[illegible]	16:00:00 23-03-2019



Staff

Name	Address	Postcode	No.	Salary
Kirsten Frewer	92 Sea Road	CO43AX	07901611223	30,000
Jay Baldwin	19 City Lees Rd	SY7 2DD	07005303213	15000
Chloe Colman	14 Warren St	V082NR	07767869[illegible]	20,000



Finances

Desc	Date e Time	Amount
Appointment	21:00 24-03-19	£50
Salary	20:00 25-03-19	-300
Appointment	22:00:00 25-03-19	40

As you can see there are clear issues with how the documentation is written down, it is messy there are scribbles and it needs an update to their system urgently. You can also see there is no set area to hold client or medical record information. There is no validation and it would be easy to lose or break and wouldn't be able to be copied and pasted like you would with a programmed database.

Inputs of the current system

The current system allows the company to add the following:

- Customer name
- Customer number
- Appointment date and time
- Staff name
- Staff address
- Staff postcode
- Staff number
- Staff salary
- Transaction description
- Transaction date and time
- Transaction difference

I will attempt to use the majority of these inputs in my final system, however I may omit some of these, such as the staff salary and replace it with their position which can be used instead as each position receives a constant salary, I may also take out the transaction description as the transactions will be linked to the appointments table you will be able to infer the transaction description from there.

Processes of the current system

The current system processes the following information:

- Storage of appointments
- Storage of staff details
- Storage of transactions
- Deletion of a record
- Searching for a record

I will use all of these processes in my system in addition to some extra. My processes will also be a large improvement in comparison to their old system as the methods for storing, deleting and searching for records is currently difficult and inconsistent.

Candidate number:
3097

Outputs of the current system

The current system can output the following:

- A list of all appointments
- A list of all transactions
- A list of all staff members

I will implement all the above outputs in addition to lots of extra ones, such as a list of the other tables that I plan to create, search outputs and some calculations such as the BMI of a selected medical record.

Limitations of the current system

The current system has a large variety of limitations. The summary of which are:

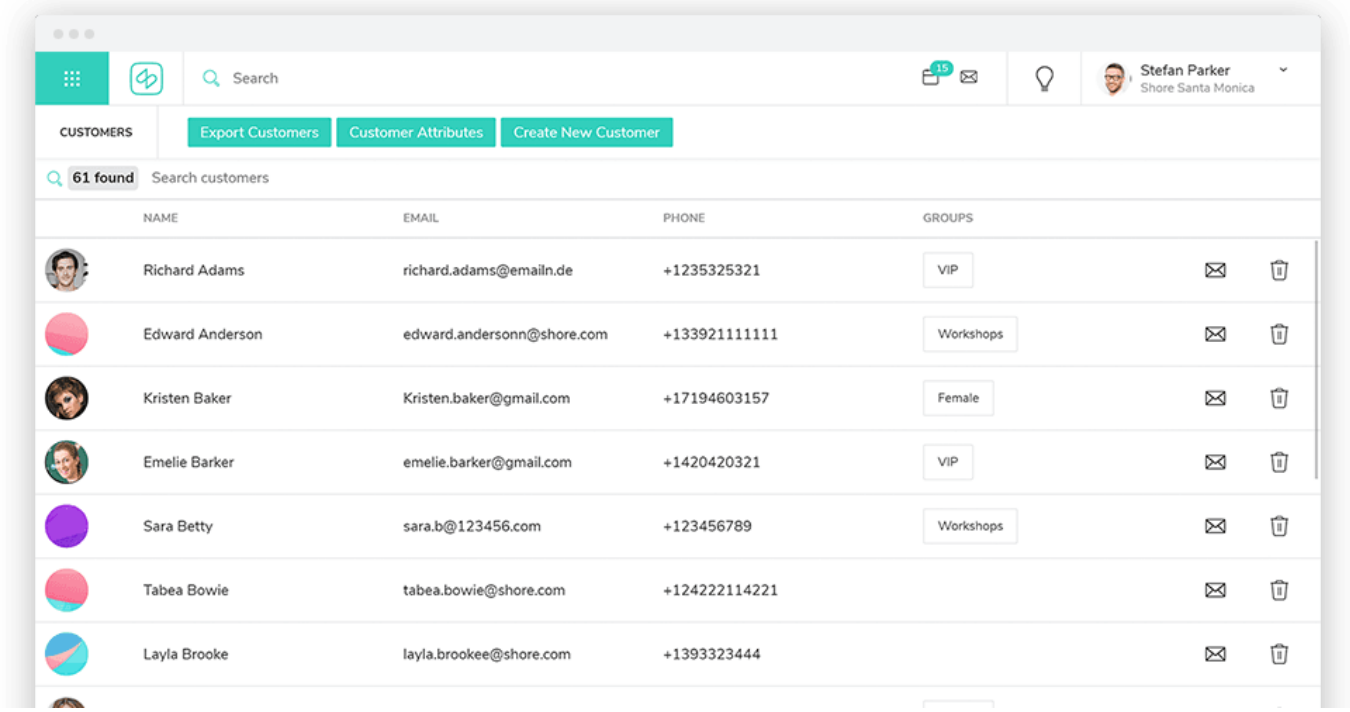
- Lack of tables, there are no tables for clients or staff
- High amounts of data redundancy
- Bad structure of data, this means that attempting to search for a record would be difficult
- Messy and not aesthetically pleasing. It is very complicated for staff members to use and it could be difficult to understand some text if the writing isn't good
- Lack of inputs
- Lack of processes
- Lack of outputs
- Bad data security, no views or login system
- Cannot be used by more than one staff member at the same time

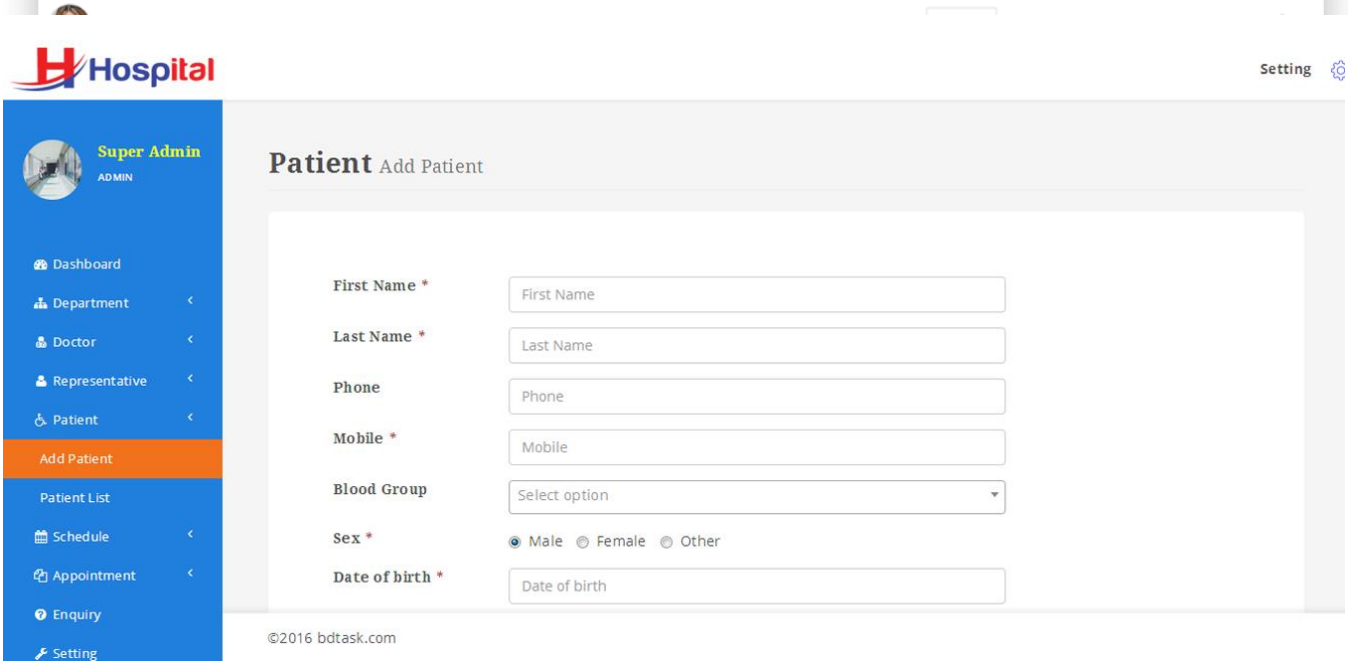
Candidate number:
3097

2.6: Desk based research

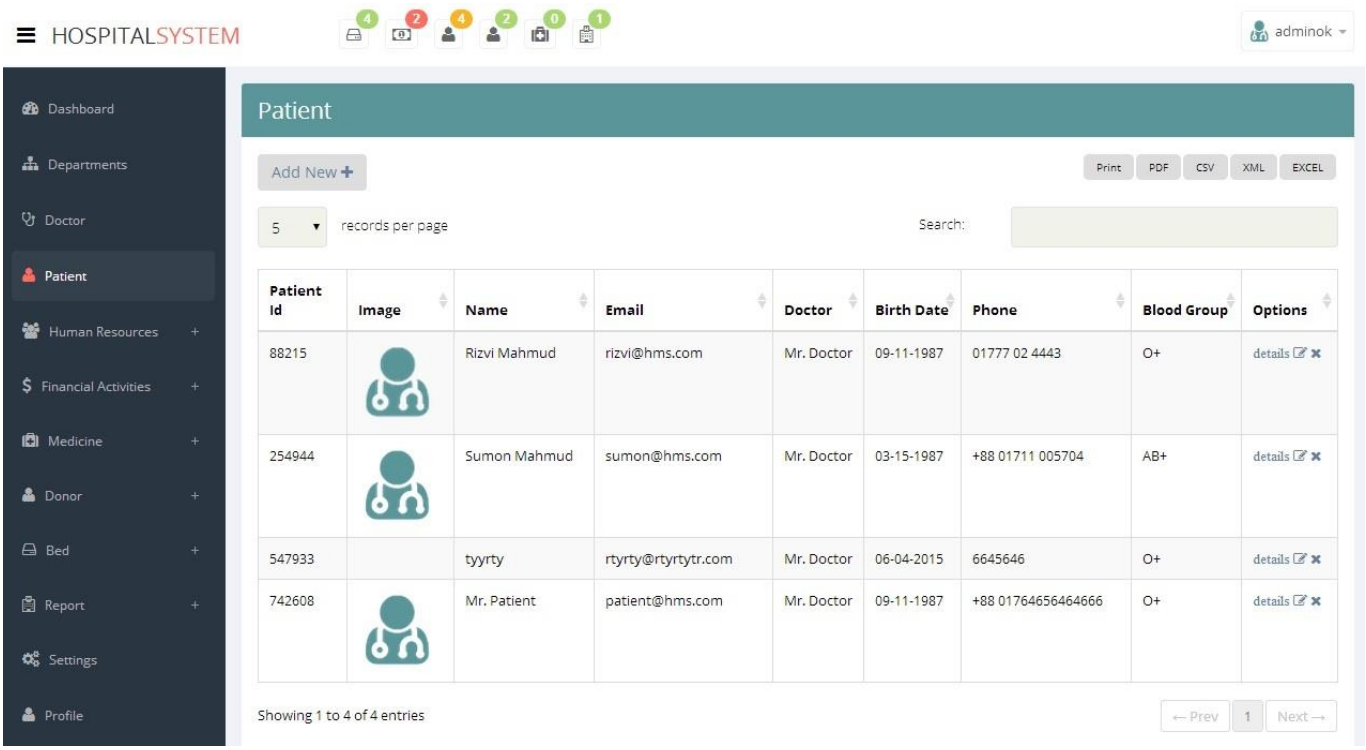
Existing solutions

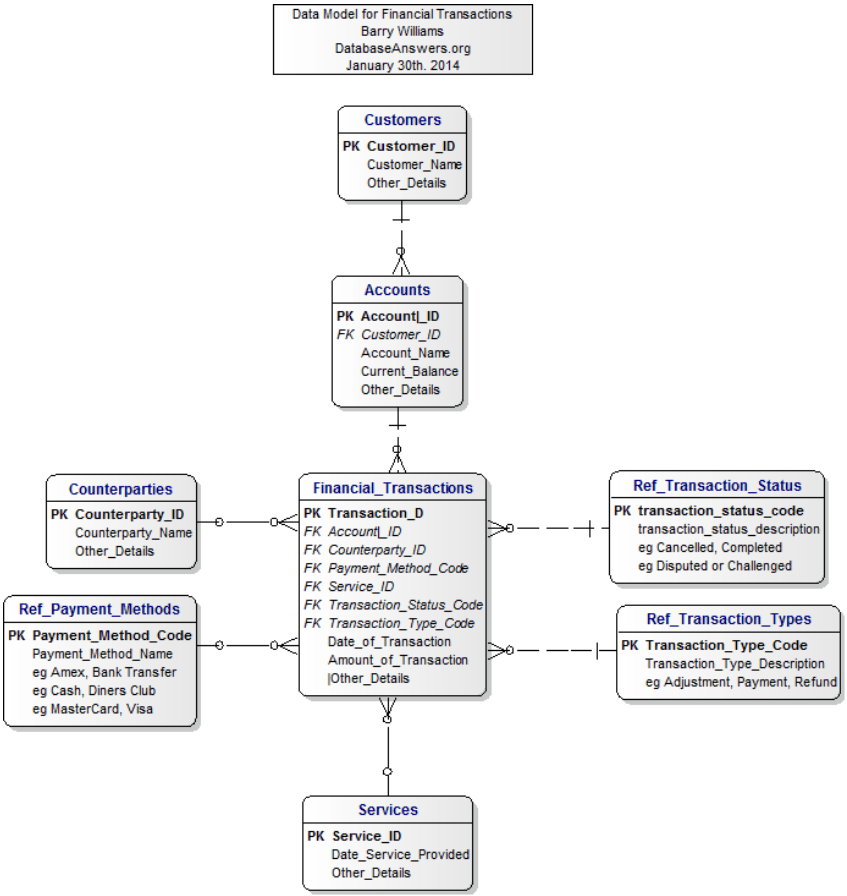
I now have a good idea on what the clinic is looking for in their database and the main improvements that they want to focus on, I will research examples of existing solutions to clinic databases to use as a base for my program and then attempt to replicate these features in the future.



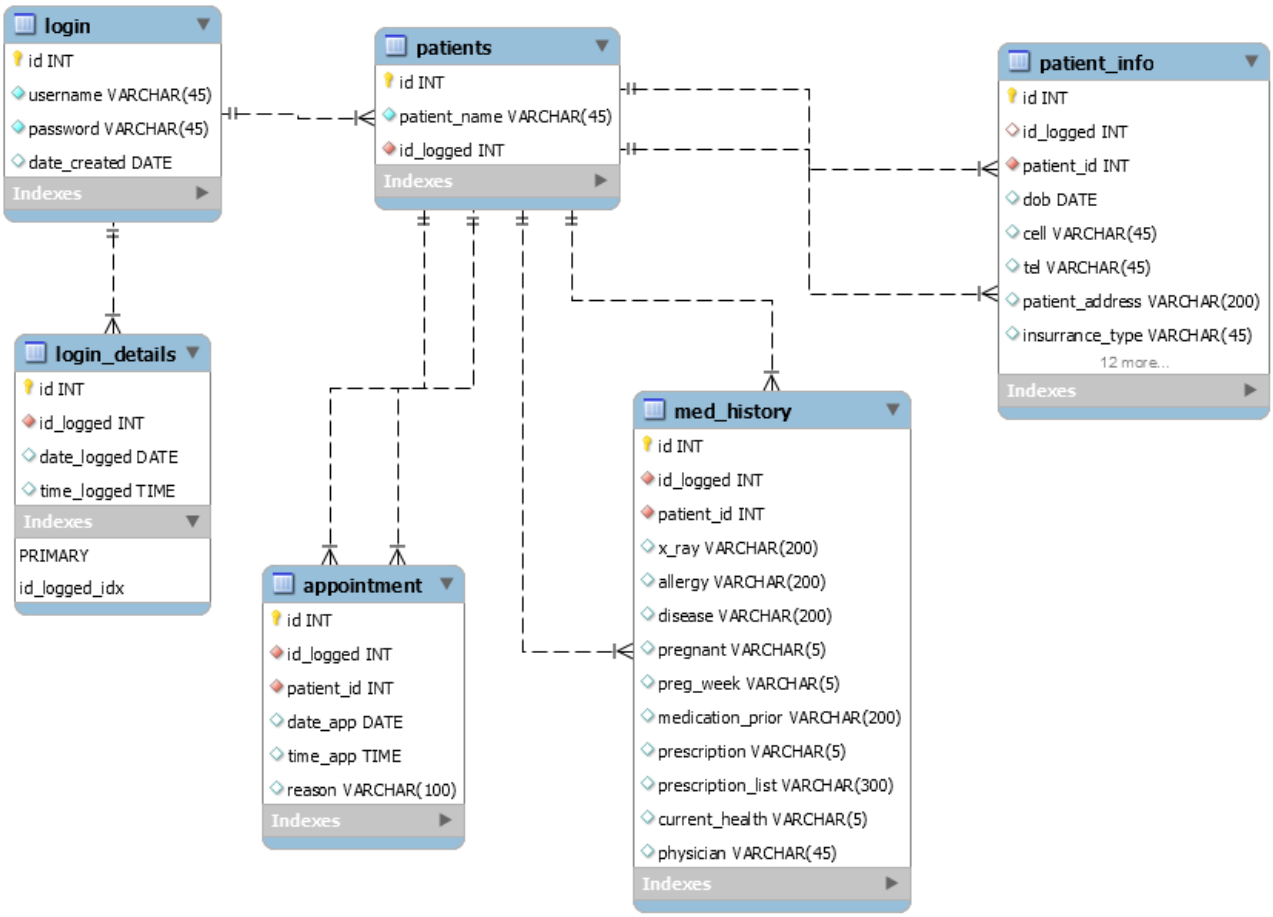


Candidate number:
3097





Candidate number:
3097



Candidate number:
3097

Summary

After looking at some existing clinic systems I have found the main parts I would like to focus on within my program:

In the first image you can see a table in a typical database format. The results are all put in a very easy to understand GUI with the table being displayed at the bottom of the screen and buttons to create and export customers at the top. This is effective as all the table and its controls are within the same page and I hope to take a similar approach in my own program mainly on the idea that this entire GUI is focused on just one page instead of being split over different windows.

In the second image you can see that it is containing a form to add a customer, it is clean and easy to use. To the side of it you can see buttons to switch between different areas of the software. Within my program I plan to develop a similar function which allows the user to switch between different parts of the program with something like tabs. In addition, I hope to create a similar form for adding customers with entry boxes and drop-down menus.

The third image is a combination of the most effective parts of the first and the second. It contains the tab and all its controls within the same window while at the same time allows the staff member to easily switch between the areas of the program in a single click. I hope to make my software very similar to this one in terms of GUI.

The forth image and fifth image is a DFD of the program and will be one very similar to my own, and a connection of the finances in the fourth image and the medical areas in the fifth. I plan to hold similar information in my databases as the ones held in these images and my connection between each of these will be very similar as well.

Finally, on the fifth image I looked at a login system that could be incorporated into my program. I plan to create a similar one with a username above and a password below with an entry button beside the login form. I do not plan to create a registering form as all accounts will be created in the staff tab to prevent a random user from claiming that they are a certain rank they aren't and then being able to access the entire program.

Overall, it is clear that my GUI should be easy to navigate and for the most part within the same page. The tables should be separated by tabs and all the controls of each of the tables should be within the same frame as the tabs. I hope to make my actual GUI as intuitive and aesthetically pleasing as the examples that I have found. In addition to this I found effective examples of DFDs that I could use in my program, I will use these images as the base for my database's format and hope take a similar approach to them. Finally, on this fifth image it contains a login system, I hope to copy a format like this. I hope that my login system will be as easy and intuitive to use as this image and that I will be able to successfully replicate it in my final program.

2.7: Specification

I aim to summarise the specification of the project that I plan to make, this will be a brief idea of what I aim to complete in my program.

- Storage of all the values in all the tables with a suitable data structure and includes third normalised form
- Suitable validation for all values that are entered into the program
- Allows the user to input values into any of the available tables
- Outputs requested values in a variety of methods
- Uses suitable data security, such as views and a login system
- Uses an intuitive GUI with simple navigation throughout the program

2.8: Requirements of stakeholders

Major stakeholders

- The owner
- Receptionists
- Physiotherapists
- Nurses
- IT administrators
- Customers
- Investors

Functional requirements

Inputs

- The system must be able to create a client
- The system must be able to create a transaction
- The system must be able to create an appointment
- The system must be able to create a staff member
- The system must be able to edit a client
- The system must be able to edit a medical record
- The system must be able to edit a transaction
- The system must be able to edit a staff member
- The system must be able to delete a client
- The system must be able to delete a medical record
- The system must be able to delete a transaction
- The system must be able to delete a staff member
- The system must be able to cancel an appointment

Outputs

- The system must be able to output a list of all clients
- The system must be able to output a list of all medical records
- The system must be able to output a list of all transactions
- The system must be able to output a list of all appointments
- The system must be able to output a list of all staff members
- The system must be able to output the BMI of a selected medical record
- The system must be able to output a list of all results of a search of a client table within the program
- The system must be able to output a list of all results of a search of a medical record table within the program
- The system must be able to output a list of all results of a search of a transaction table within the program
- The system must be able to output a list of all results of a search of an appointment table within the program
- The system must be able to output a list of all results of a search of the staff table within the program
- The system must be able to output a list of all results of a search of a client table into a text file
- The system must be able to output a list of all results of a search of a medical record table into a text file
- The system must be able to output a list of all results of a search of a transaction table into a text file
- The system must be able to output a list of all results of a search of an appointment table into a text file
- The system must be able to output a list of all results of a search of the staff table into a text file
- The system must be able to output a list of all transactions on a specific date as well as its total sum into a text file
- The system must be able to output an error when invalid data is inputted into the program

Candidate number:
3097

Process

- The system must be able to store all records in a clean and consistent manner
- The system must be able to store a list of clients
- The system must be able to store a list of medical records
- The system must be able to store a list of transactions
- The system must be able to store a list of appointments
- The system must be able to store a list of staff members
- The system must be able to validate all data added to the program
- The system must be able to calculate the BMI
- The system must be able to calculate the total sum of transactions in a day

Non-functional requirements

- The system must be able to limit access to the program through views
- The system must contain a login page
- The system must be intuitive and easy to use
- The system must be secure
- The system must be quick
- The system must need little to no maintenance
- The system must be use clear self-documenting code with comments

Methods to be used

The program will be developed using Python. This is a powerful language that is easy to use and understand and contains a large array of modules such as SQLite and tkinter. It is a very simple language which should allow any other developers trying to use the program find it easy to use and adapt. The main disadvantage of using Python however is its lack of support for GUI modules and the ones it does have are usually quite messy in comparison to other languages.

To store my data, I will be using SQLite. This is an inbuilt module within Python which allows the program to be connected much more easily to the database in comparison to other data storage methods. SQL is used worldwide and is understood by a large array of developers. SQL allows me to view, search, edit and delete records which should be extremely effective when developing the program. Finally, it allows me to connect multiple tables together which is a requirement if I want to reduce data redundancy and have my program in third normalised form.

For my GUI, I will be using tkinter. This once again is an inbuilt module in Python. Although it is quite messy in its format it is an effective GUI for Python. I will use tkinter for the login page and all forms to add and manipulate data. I will also use tkinter to allow the user to view all records within the program.

I will use a large range of validation throughout my program, the main validation checks I will use will be lookup checks, range checks, length checks, format checks, alphabetical checks, numerical checks and presence checks. Validation checks are vital if I want to prevent the program breaking in addition to keeping the database clean and orderly.

Numerous data types will be used in my program. I will use strings for things such as names, I will use integers for primary keys, I will use date and time data types for any dates and/or times I store throughout my program. I will also use Boolean data types for checks throughout the program, for example if a transaction was successful.

2.9: Success criteria

Clients

- The program must allow the user to add a client to the clients table if all information added is valid
- The program must validate all data inputted into the clients table
- The program must list all clients in a table
- The program must allow the user to search for a clients of a specific keyword in a specific column and output it into the program and into a text file
- The program must allow a user to add a client and create a medical record connected to that client's primary key at the same time
- The program must allow a user to edit a client
- The program must allow a user to delete a client
- The program must list all client information in a clear fashion

Medical records

- The program must validate all data inputted into the medical records table
- The program must list all medical records in a table
- The program must allow the user to search for a medical records of a specific keyword in a specific column and output it into the program and into a text file
- The program must automatically create a medical record when a respective client is added and connect to that client
- The program must allow a user to edit a medical record
- The program must allow a user to delete a medical record
- The program must list all medical record information in a clear fashion
- The program must allow a user to calculate the BMI of a selected medical record

Candidate number:
3097

Appointments

- The program must allow the user to add an appointment to the appointments table if all information added is valid
- The program must validate all data inputted into the appointments table
- The program must list all appointments in a table
- The program must allow the user to search for an appointment of a specific keyword in a specific column and output it into the program and into a text file
- The program must allow a user to add an appointment and create a transaction connected to that appointment's primary key at the same time in addition to saving the date and time of that transaction
- The program must allow a user to cancel an appointment
- The program must list all appointment information in a clear fashion

Transactions

- The program must allow the user to add a transaction to the transactions table if all information added is valid
- The program must validate all data inputted into the transactions table
- The program must list all transactions in a table
- The program must allow the user to search for a transactions of a specific keyword in a specific column and output it into the program and into a text file
- The program must allow a user to add a transaction
- The program must allow a user to edit a transaction
- The program must allow a user to delete a transaction
- The program must allow a user to output all transactions of a particular date and output the total sum of that date
- The program must list all transaction information in a clear fashion

Staff

- The program must allow the user to add a staff to the staff table if all information added is valid
- The program must validate all data inputted into the staff table
- The program must list all staffs in a table
- The program must allow the user to search for a staff of a specific keyword in a specific column and output it into the program and into a text file
- The program must allow a user to add a staff member
- The program must allow a user to edit a staff member
- The program must allow a user to delete a staff member
- The program must list all staff information in a clear fashion
- Usernames and passwords should be stored in this table however passwords will not be able to be seen within the program

Overall

- The program must be in third normalised form
- The database must be stored in SQL
- The program must implement views
- The program must contain a GUI
- The program must be fast
- The program must contain a readme.txt to guide any new users on how to use the program and how to troubleshoot
- The program must be intuitive and easy to understand, and it shouldn't be difficult to use any of the functions

2.10: Methods to be used

I will be using a variety of methods in the development process. These are planned now as it will let me see the plan that I have and the reason I am choosing each method. This is an extension of my success criteria and allows me to further see what the success criteria will be like in developed code.

I will be using Python as the language of development for my code. This is for a variety of reasons, the main reasons are that it contains a variety of modules that are easy to access, this will mean that it would be much easier to create a database or a GUI than in may be with other languages and the installation of these modules should be instant. I am also choosing Python as I already am quite literate in it and understand a large amount of its feature which will allow me to exploit these and use a large amount of techniques in my program to the best of my ability. Python is also a very simple language for other users to understand which means that if a future developer looks through my code and isn't fluent in Python they will likely still be able to understand the good bulk of my code.

The main disadvantage of using Python is its lack of GUI support. I will be using tkinter to create my GUI and I will use treeviews to display the table data. Tkinter is a module that comes installed with Python meaning I won't have to go through any hassle of installing it via pip. It also is a very popular GUI module meaning that it will have a lot of documentation and support for it online. However, I don't have much experience with tkinter and I find the module quite messy to use. In addition to tkinter, I am using treeviews to display my table data. I feel like treeviews are the best way to display table data as they look aesthetically pleasing and very much like a table you would find in a database in addition to being fairly simple to program and connect with Python.

Candidate number:
3097

I will use SQL for the database of my program. SQL is supported worldwide and is a very simple language to understand and it also has a module called sqlite3 preinstalled with Python. I am choosing SQL as I already have a strong amount of experience in it, having had most of it during my GCSE OCR Coursework, in addition to this experience the language is very much like pseudo code and it is easy to understand. SQL also allows me to easily, add, update, delete and search for data and I intend to use all of these features in my programmed solution.

I intend to limit the number of global variables that I use in my program and if possible, always use local variables. The reason for doing this is because global variables can make the program significantly slower. Instead of using global variables I will use local variables throughout my system and use suitable and efficient coding techniques.

The program will be stored using third normalised form. Third normalised form will significantly reduce data redundancy in the program. This will also make the database more efficient and logical. I will not however be normalising my database to a higher form, this is because this would be too complicated, and it would take too much time to program. The database will also be relational in comparison instead of a flat file database as it makes the program much more efficient.

I will implement a large array of validation throughout my program. I will do checks on formats, length, range and type checks in addition to other checks on the logic of the program. Validation is highly important to preventing invalid data being entered into the program and people breaking the program and as a result I will implement as much validation as possible into my program.

I will use self-documenting identifiers and comments throughout my code, this will be used so that future programmers who look at my code will understand the purpose and function of the code and how it works. I will try use suitable variable names so that the self-documenting identifiers and the purpose of each section of the code is clear and if the code is difficult to write in self-documenting identifiers I will use comments.

I will also use a large range of outputs, including the text. It is important to have a variety of outputs as each type of output could have a different purpose. I will use text outputs to allow the users to share and copy and paste parts of the code. I will implement special outputs that calculate values, such as BMI calculators and the sum of transactions of a date. There will also be outputs into the program within the treeview which will allow staff to easily see the data in the program without having to press any buttons.

I will implement a range of algorithms throughout my code, including a number of recursive algorithms. This will make the program much more efficient and will allow code that I right to be much more flexible. I can use this for things such as for outputting a database for one line at a time till the last line.

Finally, I will use a lot of security measures throughout my code. This will mainly focus on the login page and on views throughout the program. I will not however include data encryption and backups as this would take too much time for what it is worth. Views will be useful in preventing access from staff members who shouldn't see or be able to manipulate certain tables and a login form so non-staff users cannot access the database.

Overall, I will use a wide range of methods throughout my program. These will be all be used and will bring the program to the best level it can be.