Section 1: Discussion

Table of contents:

Problem 2

Broad aims of the project 2

- Stakeholders 2
- Aims 2
- Discussion of proposed solution 3
- The proposed ER diagram 4

Possible limitations of a solution to the problem 5

Feedback from others 5

- Evidence of consultation 5
  - Blank questionnaires: 5
  - PowerPoint 6
  - Questionnaires 7
  - Discussion of all questions and suggestions 8

Second Draft of sections 1 - 3 12

- Problem 12
- Broad aims of the project 12
  - Stakeholders 12
  - Aims 13
  - Discussion of proposed solution 13
  - The proposed ER diagram 14
  - Possible limitations of a solution to the problem 16

Problem

The problem that I have decided to solve is to create an off the shelf software solution for a reservation system for a hotel. The solution that I will be creating will be desktop based.

Broad aims of the project

Stakeholders

The stakeholders and their requirements are listed below:

- The customer - One requirement for the customer would be to produce a hard copy of the bill to the customer. Another requirement for the customer is to inform them what time the room will be ready for them and what time they will need to book out.
- The receptionist - A requirement for the receptionist would be for them to be able to view what customer is in what room and what rooms are available.
- The management staff - A requirement for the management staff would be to check if there are any problems within the hotel.
- The cleaning staff - The cleaning staff will need to know when customers are leaving the room so they are able to clean it for a new customer to arrive.
- Accountancy staff - A requirement for the accountancy staff is for them to check the customer's bill.

Aims

Broad aims of the project:

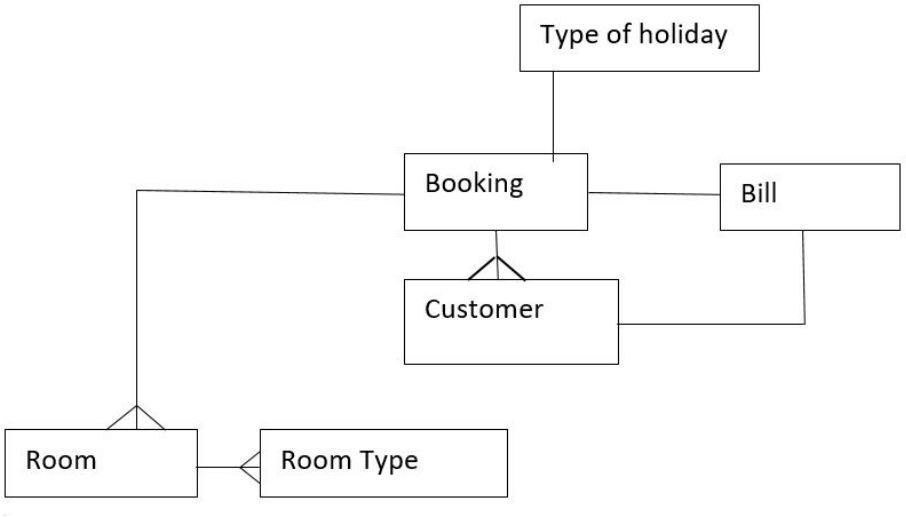
- To produce a hard copy of the bill to the customer.
- Allow the staff at reception to edit, delete or add records to the database.
- Allow the staff to view the database.
- Allow the management staff to see what rooms are booked.
- Stores the customer's details and their booking.

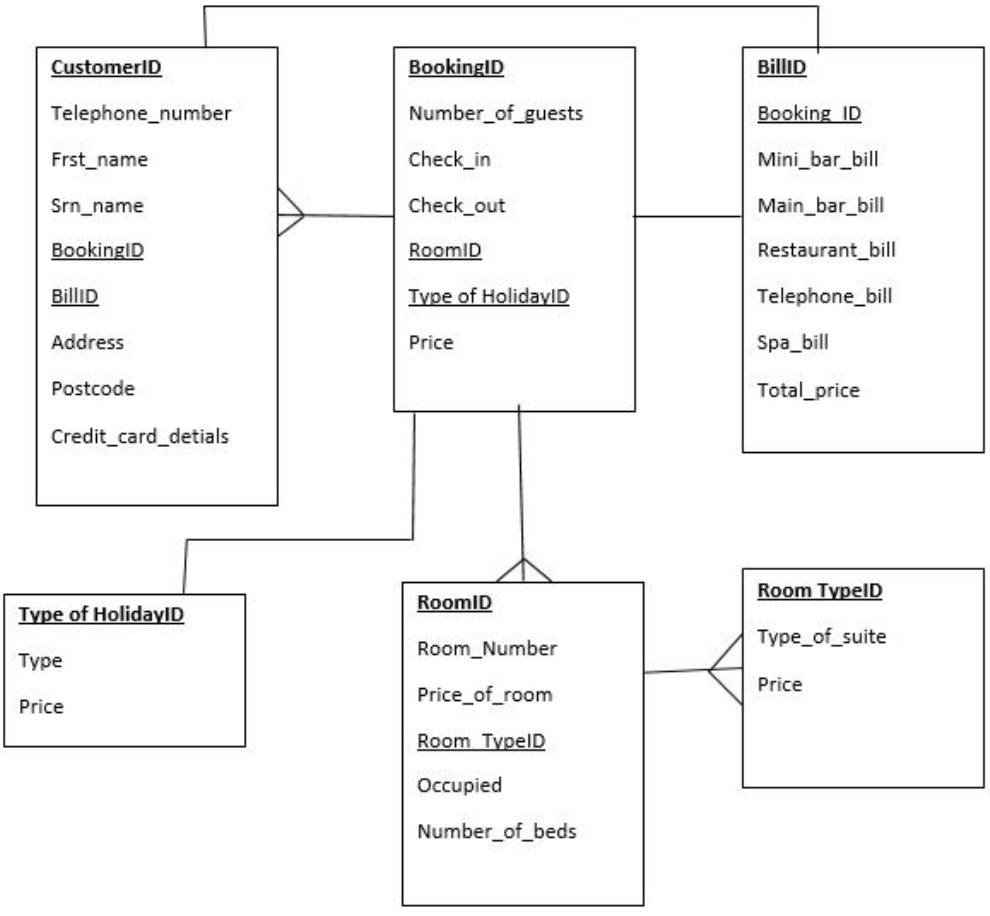
Discussion of proposed solution

My solution will include a relational database and it will be a Graphical User Interface (GUI). A relational database is a database but with many linked tables. The reason why I am using a relational database over a flat file database is because this will reduce data redundancy as in a flat file database the same data will need to be repeated several times. Another benefit of a relational database is that once something is changed it will be updated in all other tables but in a flat file database the same piece of data will need to be updated in every record. This will cause problems as a record may be missed. The reason why I have chosen to produce a GUI for my solution is because they are more intuitive than a Command Line Interface (CLI) making my solution more user friendly.

I will be creating a system that will allow staff to edit, delete and add records from a database. The programming language that I will be using will be Python as it allows me to utilise the SQlite3 library and the tkinter library. This is also the programming language that I have the most experience using. One of the programming tools that I will be using is SQlite3 as this will allow the user to manipulate a database. I will also be using tkinter as this will allow me to create a GUI (Graphical User Interface) which will make my proposed solution easier for staff to use. This will also allow me to utilise the ttk module from tkinter to create a treeview table. The advantage of using this as it allows me to display the database and it allows the user to view the database before manipulating it. The user will also be able to select a record from the database more intuitively.

The proposed ER diagram





Possible limitations of a solution to the problem

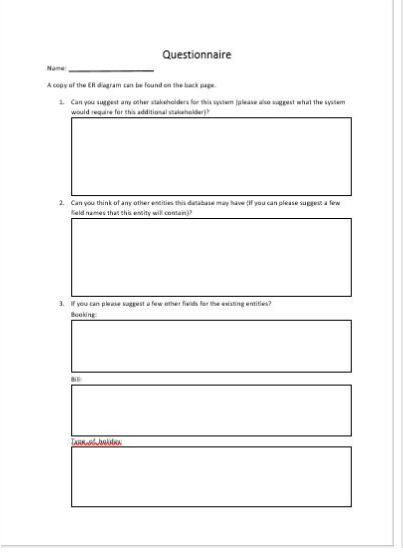
One major problem that I will have will be the time limitations. As I plan to produce a hard copy of the bill and output the database in the form of a treeview I may run out of time and only provide one of these output methods.
Another problem that I may encounter is the staff preferences over the design of the GUI as I may not be able to deliver a well-designed GUI because of my skill as a programmer. As my solution does not have a specific end user I will not be able to query someone about the design of my GUI and if it is easy to use. As a result of not having a specific end user I also do not know the level of expertise the end user will have. This is a problem as I may assume that the user will know certain technical terms and include these in my error messages.
Another problem I may face is the error messages. Due to my expertise and time limitations I may not be able to give precise error message i.e. I may not be able to tell the user the reason for the error. This means the solution may have vague error messages and the staff may waste time trying to locate the issue.
Talk about MAC, Windows, interpreted languages.

Feedback from others

Evidence of consultation

For discussing my problem and my proposed solution I used both questionnaires and a powerpoint.

Blank questionnaires:


Questionnaire

Name: _____

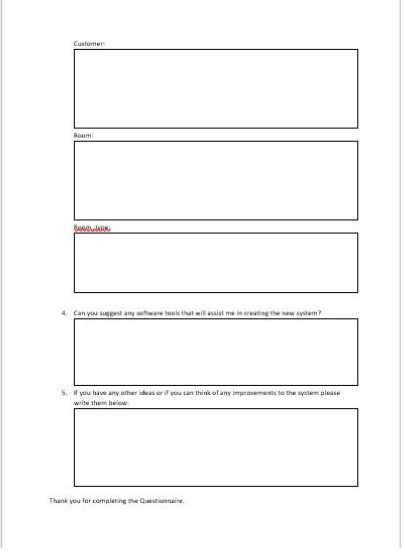
A copy of the ER diagram can be found on the back page.

1. Can you suggest any other stakeholders for this system (please also suggest what the system would require for this additional stakeholder)?
2. Can you think of any other entities this database may have (If you can please suggest a few field names that this entity will contain)?
3. If you can please suggest a few other fields for the existing entities?

Booking:

Bill:

Type_of_holiday:

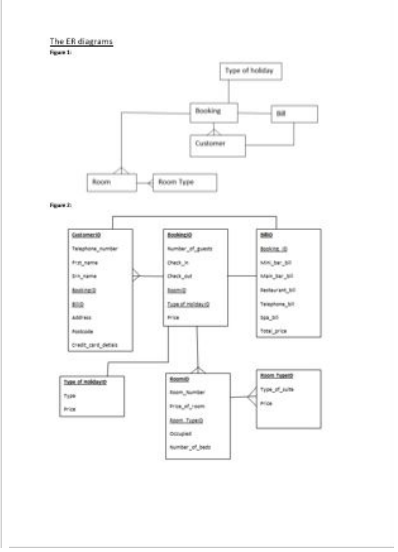

Customer:

Room:

Room_type:

4. Can you suggest any software tools that will assist me in creating the new system?
5. If you have any other ideas or if you can think of any improvements to the system please write them below:

Thank you for completing the Questionnaire.



PowerPoint

Computer Science NEA

By Adam Logan

Introduction

- My idea for the CAT is to update an existing hotel booking system.
- This solution will be desktop based.

Broad aims of the project

- To produce a hard copy of the bill to the customer.
- Allow the staff at reception to edit delete or add records to the database.
- Allow the staff to view the database.
- Allow the management staff to see what rooms are booked.
- Stores the customer's details and their booking.

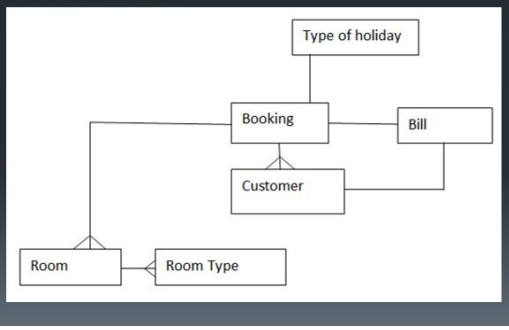
List of stakeholders

- The customer.
- The receptionist.
- The management staff.
- The cleaning staff.
- Accountancy staff.

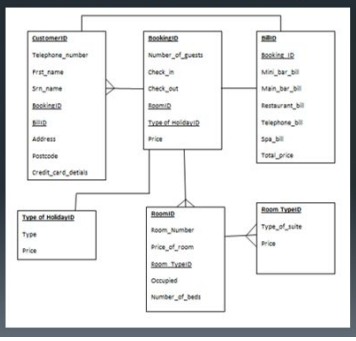
Potential requirements for each stakeholder

- One requirement for the customer would be to produce a hard copy of the bill to the customer. Another requirement for the customer is to inform them what time the room will be ready for them and what time they will need to book out.
- A requirement for the receptionist would be for them to be able to view what customer is in what room and what rooms are available.
- A requirement for the management staff would be to check if there is any problems within the hotel.
- The cleaning staff will need to know when customers are leaving is the room so they are able to clean it for a new customer to arrive.
- A requirement for the accountancy staff is for them to check the customers bill.

ER diagram



ER diagram (continued)



Possible limitations

- I may not have enough time to produce a system that will display the database in a table format and produce a hard copy of the bill due to time limitations.
- Due to may expertise I may not be able to produce a well-designed GUI.
- I may not be able to produce precise error message so the staff may waste time attempting to locate issues.

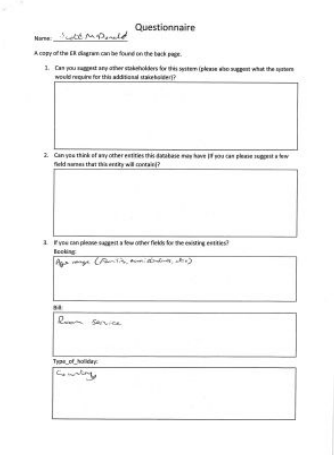
Software tools

- The programming language I intend to use is Python because I have the most experience with this language.
- I will be using the tkinter library (this will allow me to create the GUI) in Python and the sqlite3 library (this will allow me to create and manipulate databases) as well.

Thank you for listening

- Any questions?

Questionnaires


Questionnaire

A copy of the ER diagram can be found on the back page.

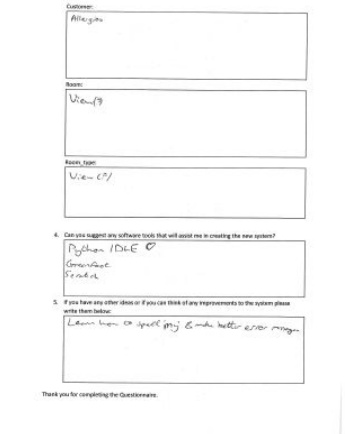
1. Can you suggest any other stakeholders for this system (please also suggest what the system would require for this additional stakeholder)?
2. Can you think of any other entities this database may have (if you can please suggest a few field names that this entity will contain)?
3. If you can please suggest a few other fields for the existing entities?

Booking:

Bill:

Room service

Type_of_holiday:


Customer:

Allergies

Rooms:

View?

Room_type:

View ??

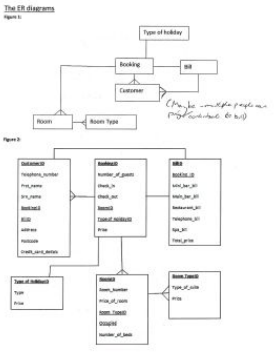
4. Can you suggest any software tools that will assist me in creating the new system?

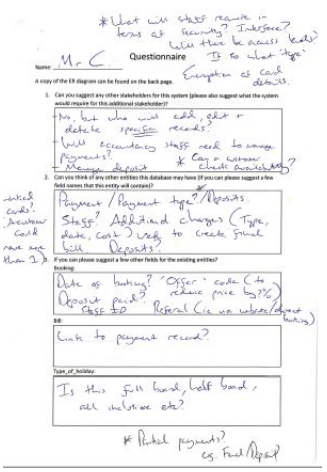
Python IDLE
Sqlite3

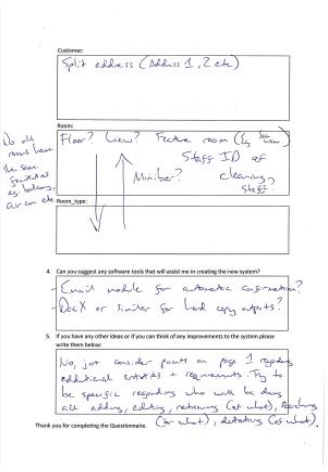
5. If you have any other ideas or if you can think of any improvements to the system please write them below:

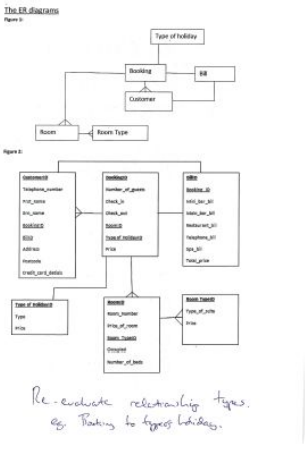
Learn how to specify email address error message

Thank you for completing the Questionnaire.









Discussion of all questions and suggestions

Suggestion	Discussion
What will staff require in terms of security? Interface? Will there be access levels? If so what 'type'.	Certain staff will have different access levels i.e. management staff will have CRUD (Create, Retrieve, Update and Delete) functionality for all entities whereas the customer will only be able to view their record in the booking table, room table, room type table, bill table and type of holiday table. They will also be allowed to view and edit their record in the customer table. The interface that I have chosen will be a GUI. I will also create a log in system will users will have to create an account for the system. This will assist me in implementing the access levels, as certain users will have different access levels as others.
Encryption of card details.	This is an idea in which I will certainly implement, as this will improve the security of the system. I may also encrypt the passwords of the accounts for the log in system, which I intend to implement.
Can a customer check availability?	Yes, a user will be able to do this by being able to view the room table but not be able to change it in any way.
Will accountancy staff need to manage payments and/or deposits?	Yes, they will and this is why they will have access to the bill table.

New entities such as payment/payment type (partial payment? E.g. final/deposit), deposits, staff and additional charges (type, date, cost) used to create final bill.	I will be including the payment type entity and it will and linked with the booking entity. This as the name suggests will store the payment type i.e. is it a partial payment, are they paying in advance or if they will pay when they check out? I could add the deposits entity and it will have a 1 to many relationship with the booking table as no matter what room the customer books it will have the same deposit. Another entity that I may add is the staff entity. This will have the field's staffID, salary, jobID. The jobID will be a foreign key in this table. This will be linked (by a 1 to many relationship) to another new entity called job and it will have the field's jobID and job (this is to reduce data redundancy). The staff entity will also be linked to the room table by a many to many relationship. The final new entity, which I will be discussing, is the additional charges entity. I will not be implementing this entity as I believe the bill entity already has suitable field names for all possible charges.
Other field names for the booking entity: age range (families, university students, etc), date of booking, 'Offer' code (to reduce price of bill by %), deposit paid and referral (via website/direct booking).	First, I will address the field name age range. The reason why I will not be implementing this is because this it only refers to the individual doing the booking and not the group as a whole e.g. a family can include a university student. The next field name I will discuss is date of booking which I will implement. The reason for this is because I believe it would be useful for the customer and the hotel staff to be able to check this. The 'offer code' field will only contain a 'Y' or a 'N' as only if there is an offer code will need to be checked. I may have to create a sub-routine to check if an offer code is valid or not. A possible limitation I may have is to link this to the bill entity and actually take the percentage of the total bill. The deposit paid field may be better to go onto the bill entity as before a customer will have needed to actually book the room to pay the deposit. I could include the referral field as it will be useful for management to know where customers are booking the rooms from.

Other field names for the bill entity: link to payment record, room itself and room service.	If I do implement the payment/payment type entity, I will include a link to the bill entity. The room itself entity is referring to how much the room costs. I will not need this entity as bookingID is a foreign key in this entity and this store the price if the room itself. I will be implementing the 'room service' field as this will need to be added on to the total price of the bill.
Another field name for the type of holiday entity is country.	Country cannot be included as a field name as this is for a singular hotel.
Is type of holiday full board, all inclusive, etc?	Yes, this is exactly what is meant by type of holiday.
Other field names for the customer entity: split address (address line 1,2,etc), age, allergies, facilities and passportID.	I will be implementing the split address field and I will be implementing the age field. The allergies field I may implement as the customer entity is the individual booking the room not all the customers that may be staying in the room. This will store the allergies of all individuals that will be in the room. The facilities field I may make into a separate entity, as there will be a lot of repetition. This will also be better suited to link with the room entity or room type entity as the facilities will be related to the room. The passportID field I will also implement but I may rename this to passport as it sounds like a primary key or foreign key.

Other field names for the room entity: floor, view, feature room (e.g. sea view), staffID, minibar.	I will be implementing the field names floor but I believe this will be better suited in the room type entity as many rooms will be on the one floor. I believe view would be better suited as its own entity as many rooms will have the same view i.e. there will not be only one room with an 'ocean view'. I am not going to include the feature room field as this information has already been stored in the view field as discussed above. If I include this and the view field this will increase data redundancy as data is being repeated. If I do implement the staff entity, I will also add the staffID field to link the two entities together. I may implement the 'mini bar' field into this entity but it may be better suited in the facilities entity if I do implement it.
Other field names for the room type entity: view, details about room (balcony, pets)	As discussed above I am going to implement the view field in this entity. I will not implement the 'details about room' field but I may add balcony and pets as separate fields. Both of these fields will only store either a 'Y' or a 'N'.
The use of docx, message boxes and email module for automatic conformation.	I will be using the docx library in python, as this will allow me to produce a hard copy of the bill. I will also be using the message box library, as this will allow me to produce error messages and create dialogue boxes. I may also use the email module for the purpose that was suggested which was automatic conformation.
Add login and user credentials.	I will implement this as this will assist me in creating access levels for each stakeholder.
The booking and type of holiday should be a 1 to many.	This should not be the case as one booking will only have one type of holiday i.e. you cannot book an all exclusive holiday and an half board holiday as the same holiday.
Maybe multiple people could contribute to a bill.	This could be possible but the hotel will only need to store one card as if customers wanted to do this they could transfer the money to the person who booked the holiday.

There should be a many to many relationship between the customer entity and the booking entity.	As the customer entity is only for the individual who booked the room this will still be a 1 to many relationship as a customer can book multiple rooms.
There is no point in the room type table as the ID will need to be repeated.	This is true but it is better to repeat the primary key as this will most likely take up less storage than the data it sores. For example, the primary key 1 takes up less storage than 'presidential suite'. Keeping room type as an entity will therefore reduce data redundancy.

Second Draft of sections 1 - 3

Here is the second draft of sections 1 – 3 including the changes to the proposed solution based on the feedback:

Problem

The problem that I have decided to solve is to create an off the shelf software solution for a reservation system for a hotel. The solution that I will be creating will be desktop based.

Broad aims of the project

Stakeholders

The stakeholders, their requirements and what they are able to access are listed below:

- The customer - One requirement for the customer would be to produce a hard copy of the bill to the customer. Another requirement for the customer is to inform them what time the room will be ready for them and what time they will need to book out.
    - The customer will be able to view, edit and add their record on the customer entity. They will also be able to view their record in the booking entity, room entity, room type entity, the holiday type entity and the bill entity. They will not be able to access any other records except form their own.
- The receptionist - A requirement for the receptionist would be for them to be able to view what customer is in what room and what rooms are available.
    - The receptionist will be able to view the entirety of the customer entity, room entity, room type entity, holiday type entity and booking entity. They will also be able to view the staff entity.

- The management staff - A requirement for the management staff would be to check if there are any problems within the hotel.
    - Management staff will be able to view, add, edit and delete all records in all entities.
- The cleaning staff - The cleaning staff will need to know when customers are leaving the room so they are able to clean it for a new customer to arrive.
    - The cleaning staff will only be able to view the room entity and the staff entity.

- Accountancy staff - A requirement for the accountancy staff is for them to check the customer's bill.
    - Accountancy staff will only be able to view the bill entity.

Aims

Broad aims of the project:

- To create a login system for the booking system.
- To produce a hard copy of the bill to the customer.
- Allow the staff at reception to edit delete or add records to the database.
- Allow the staff to view the database.
- Allow the management staff to see which rooms are booked.
- Stores the customer's details and their booking.

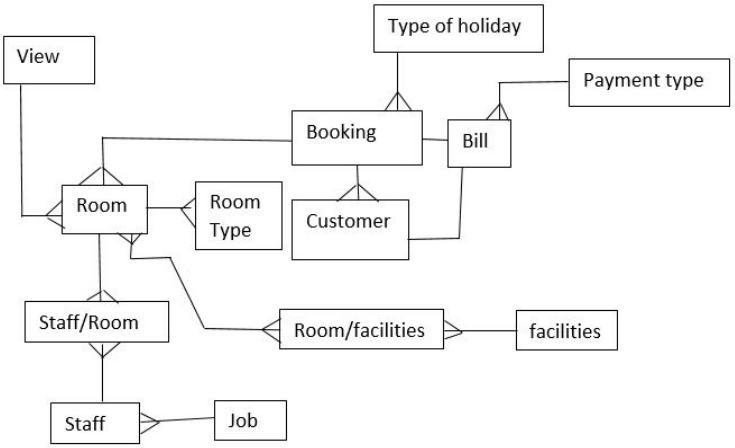
Discussion of proposed solution

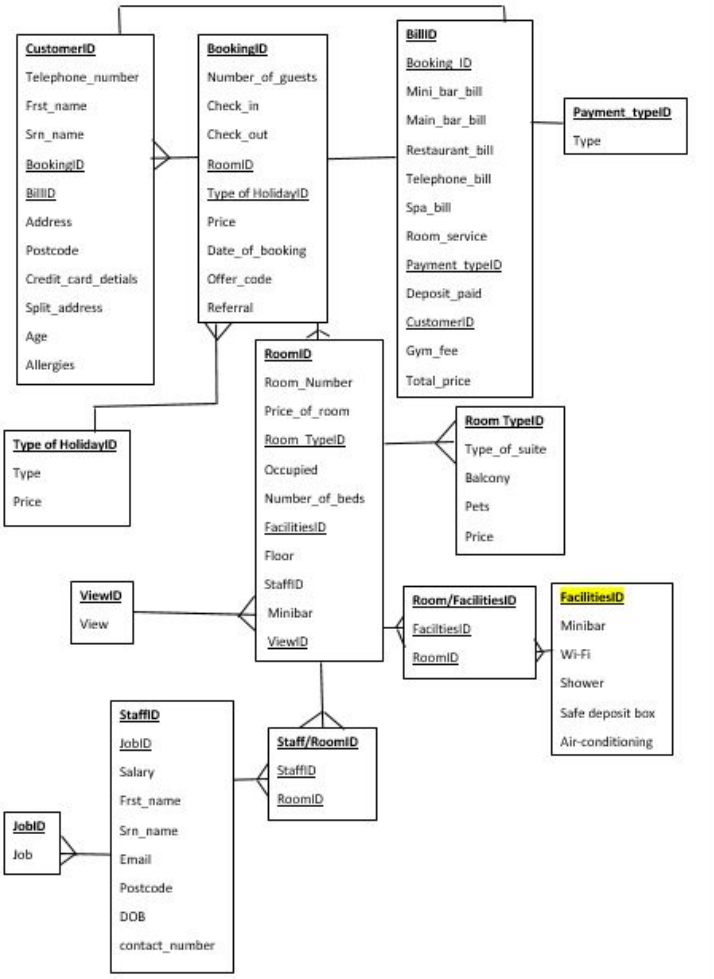
My solution will include a relational database and it will be a Graphical User Interface (GUI). A relational database is a database but with many linked tables. The reason why I am using a relational database over a flat file database is because this will reduce data redundancy as in a flat file database the same data will need to be repeated several times. Another benefit of a relational database is that once something is changed it will be updated in all other tables but in a flat file database the same piece of data will need to be updated in every record. This will cause problems as a record may be missed. The reason why I have chosen to produce a GUI for my solution is because they are more intuitive than a Command Line Interface (CLI) making my solution more user friendly.

I will be creating a system that will allow staff to edit, delete and add records from a database. The programming language that I will be using will be Python as it allows me to utilise the SQlite3 library and the tkinter library. This is also the programming language that I have the most experience using. One of the programming tools that I will be using is SQlite3 as this will allow the user to manipulate a database. I will also be using tkinter as this will allow me to create a GUI (Graphical User Interface) which will make my proposed solution easier for staff to use. This will also allow me to utilise the ttk module from tkinter to create a treeview table. The advantage of using this as it allows me to display the database and it allows the user to view the database before manipulating it. The user will also be able to select a record from the database more intuitively.

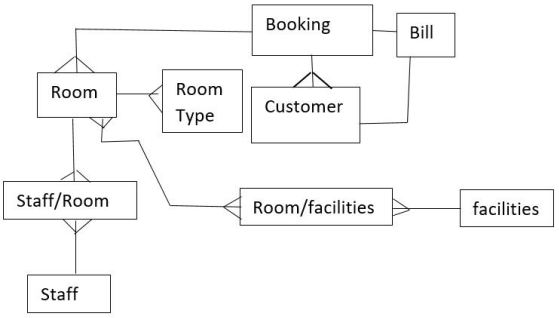
The proposed ER diagram

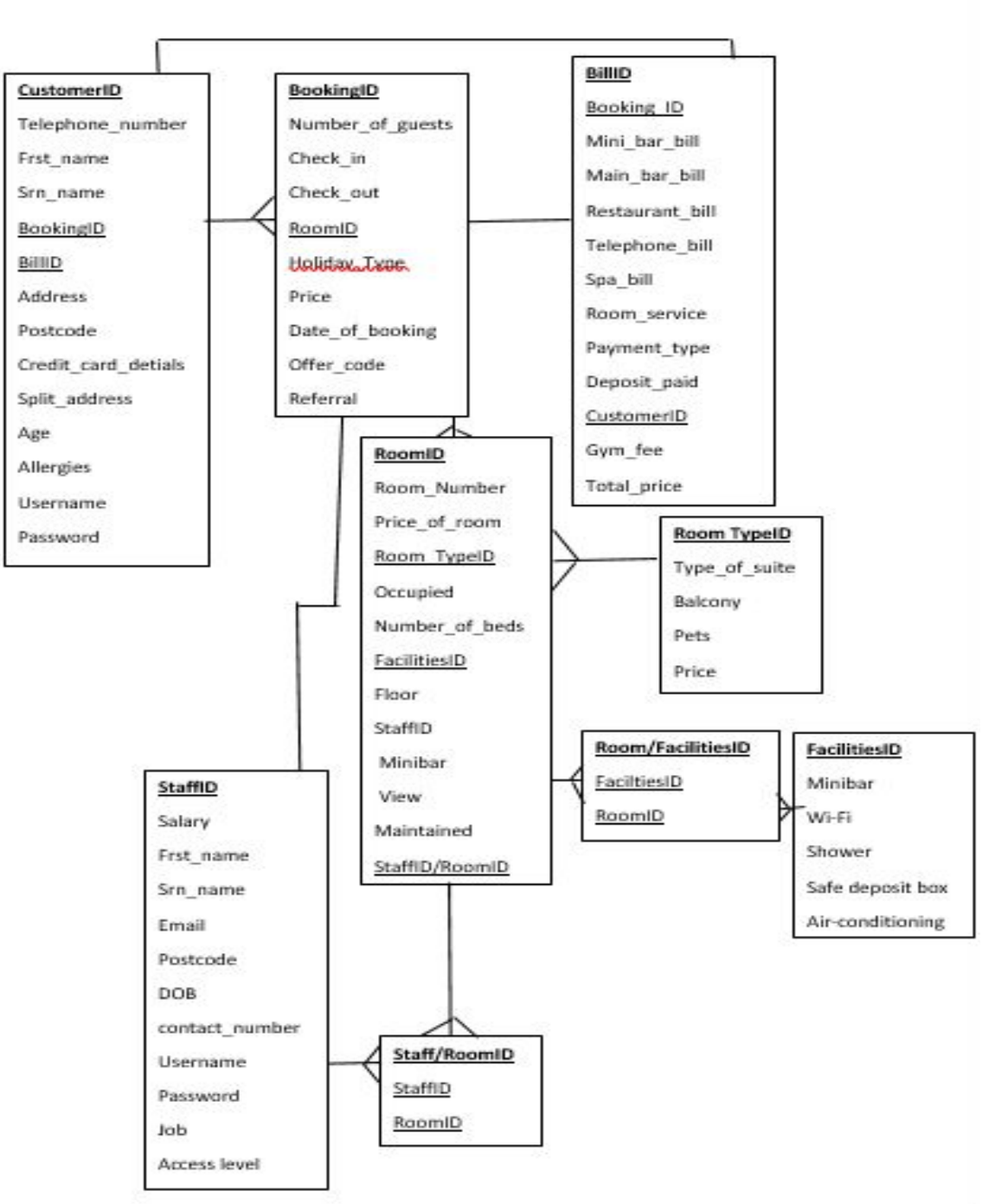
This discussion occurred on the 22nd September 2019





The 3rd draft of the ER diagram:





Possible limitations of a solution to the problem

One major problem that I will have will be the time limitations. As I plan to produce a hard copy of the bill and output the database in the form of a treeview I may run out of time and only provide one of these output methods.
Another problem that I may encounter is the staff preferences over the design of the GUI as I may not be able to deliver a well-designed GUI because of my skill as a programmer. As my solution does not have a specific end user I will not be able to query someone about the design of my GUI and if it is easy to use. As a result of not having a specific end user I also do not know the level of expertise the end user will have. This is a problem as I may assume that the user will know certain technical terms and include these in my error messages.
Another problem I may face is the error messages. Due to my expertise and time limitations I may not be able to give precise error message i.e. I may not be able to tell the user the reason for the error. This means the solution may have vague error messages and the staff may waste time trying to locate the issue.
A possible limitation would be to create a login system for the booking system. The reason why I may struggle to do this is because I may need to create another database or entity that will store all the usernames, passwords and access levels. I will also have to encrypt these passwords to make sure the system is secure.
I may also struggle to implement access levels. As the customer stakeholder should not be able to view all of the customer records in the customer entity but only their own record, this may cause problems. I will need to somehow store what customer is accessing the system so they will only be able to view their record.