**Write a short coherent report (not exceeding 1000 words) summarising the functional requirements of the company, an explanation of your database design and its implementation [Note that the ERD, SQL statements and the PHP code do not give a full explanation]. Finalise your report with a brief appraisal of your application’s features. A URL (and login credentials, if any) for your index Web page should be clearly written in the report.**

URL: https://teach.scam.keele.ac.uk/prin/y1j08/DatabaseAssignment\_23047207/login.php

Username: admin

Password: admin

The functional requirement for the company is to allow customers to book rooms in various branches of the company. A standard customer must only be allowed to book standard rooms, whereas a business customer may book any room, including standard or executive rooms. Customers may book as many rooms as they desire, subject to the restrictions determining whether they are a standard or business customer, as long as there are sufficient rooms available. Standard rooms can be decorated for parties, thus the customer must specify. Business customers must have access to television and the internet, but they also have the option of booking a buffet, and the buffet details must be saved. Business customers also have access to pay by invoice.

My ERD demonstrates the relationships between entities, with the branch having a one-to-many relation with the employees. Branch has a one-to-many relationship with the supplier, just as it does with the item. The branch also has a 5 to 30  relationship with rooms, which is a general entity. The rooms are specialised into basic and executive rooms. The regular room is suitable for parties and may be decorated, whilst the executive room features a television, internet connection, and the ability to order a buffet. The company also has a general customer entity that specialises into business and standard customers. Standard customers can only book standard rooms, therefore they have a zero-to-many relationship and are connected by a booking relation. The business customer can reserve whatever room they desire through the booking relation. The referential integrity of the foreign keys are used in the logical ERD to link each tuple in the child relation to the tuple in the parent relation containing the matching candidate key value. The referential integrity is ensured using ON DELETE CASCADE constraint, ensuring bookings are erased when a customer account is deleted.

My SQL database has tables for branch, room, employee, supplier, item, customer, standard customer, business customer, executive room, standard room, and an associative  entity or relationship known as booking.

My interface begins with a login.php file, which is designed for employees or authorised users. Once logged in, the user gets taken to the index page, which includes a button for logging out of the session. The table from SQL's BRANCH table is shown below. When you click on one of the tuples, i.e. one of the branches, you are taken to a page that displays all of the rooms at that branch; this page also includes a logout button and a home button for returning to the main index page. This page has another table for the ROOMS table. When you click on one of the tuples or the rooms row, it displays the assigned bookings for that room. You still have the home and log out buttons. When you click on a booking, you will be sent to the customer's details, which include their address, email, and phone number. If you click on the customer tuple, you will discover additional information about them. This displays a new table that joins the BUSICUST and STANCUST tables; this table is filled in based on whether a customer is a business customer or a standard customer; if the customer is a business customer, the attributes of the business customer will be filled in, while the rest of the attributes of the new table will be blank; the same is true for standard customer details.

If you click the home button again, you will be transported back to index.php. Below the branch table, there are two buttons labelled suppliers and employees. When you click the employees button, you will be taken to the EMPLOYEES table, which contains attributes for each employee's details. If you click the home button on this page, you will be returned to the home page, and if you click the supplier button, you will be directed to the SUPPLIER table, which contains the supplier's information and the branches to which they are assigned. When you click on a supplier, you can see what items the supplier supplies. Here more than one supplier can supply to a branch. When you click on the logout button you are logged out of the session and redirected to login.php.

While the interface successfully displays data relationships, it lacks create/update functionality for bookings. Future versions could include a booking form with date validation, role-based access (e.g., managers versus employee), and buffet order monitoring for executive rooms.