Alexander Varghese, Dominick Arnaldo, Jonathan Bar-Eli, and Sarvesh Soni

Professor Chang

CSAS 4115 BA

13 May 2019

Database Responsibilities

Throughout the entire project, our group had faced many challenges. We decided to split the work up into parts we were comfortable working with. One of our biggest issues when working with the project was with the EER diagram. There would be many times where in order for a form to work, we would have to change a parameter or constraint for a table. If there was even a small change in the table, we would have to delete the current schema, forward engineer the new EER diagram and then work with the new database/schema until there was another parameter or constraint that we needed to change. Another issue that we had that pushed our group back a few days from finishing the project was Github. One of our group members always had trouble with pushing and pulling from Github which prevented him from getting the latest code and pushing all of his changes. This caused for extra wasted time in figuring out how sync up the group’s code with his laptop. Another issue that we had was constantly changing the components and forms for the website. Every time we would provide an input into a form, the form would fetch all of the information but, it would neither update the database nor the database would give the user back any information. One last issue was that we were able to delete an equipment that was already assigned to an employee, which is not supposed to happen. In the project, Sarvesh Soni had created the SQL scripts for our sample data, the documentation and manual of the website. Alexander Varghese had created the entire design of the website, the Home page, the login process (fetching and SQL scripts of a user’s credentials) and connected the database to the website. Jonathan Bar-Eli created forms and components for the Administrators and Employees which were adding, deleting, searching and updating the employees, vendors and equipment. Dominick Arnaldo created the fetch and SQL statements, routing methods, the implementation of the forms, the AdminPage and EmployeePage. One open issue that we could consider problematic is that we created a “Departments table”, but it is only connected to the “Equipments” table. We realized that we should have connected the “Departments” table to the “Employee” and “Office Location” table as well. A feature that we should have had was displaying the floor, room, and department in the employee table so it would be easier for the administrator to reference instead of looking at the database for further information on a specific row. There were some other features we could have added to make the user experience much better including adding a “Welcome, (name of employee)” text next to the “Sign Out” button. We could have also created a faster, more convenient search bar which would not be case sensitive and have the ability to sort the table according to each individual column and making more it more efficient implementing in the server-side instead on the client. The last feature my group could have added was when assigning an equipment, the administrator or employee wouldn’t need to manually erase the old fields when assigning the new ones.

Throughout the entire project, my group had made many changes to the EER diagram and source code. At first, we decided that we didn’t need and ID for Equipment, Room, and Floor. But, we found out very early on that we needed these IDs because it was very important in indicating which rows held what values. The second change that we made to the EER diagram was to make many of the input values in the Equipment table to be either null or not null. My group decided that we would give the user choices when assigning an equipment to another employee or themself. The third change that we made was in the source code where our group, at first, decided that we create a seperate page for each form. But, we decided that instead of having a bunch of files, we would create a pop-up box that would open a new form. The fourth change that my group made was instead of every employee having the ability to do the same functions, my group decided that we should have least of privileges and broke up the users between Administrators and Employees. The last major change that my group made was deciding to not provide the user with a “Personal” page that would show the user’s information. Overall, I believe that my group did an excellent job and met all of the requirements of the project.