Database Systems Project: LAU Bus Management System

Project Report

Rebecca Bou Zerdan, Razan Houdaifa, and Yara Hamad

Abstract— This report presents the implementation of the database final project which revolves around the current LAU bus management system. The website created allows students to book a seat along with the respective timing and stops. Hence, this will allow retrieving, editing, and adding data to the database.

Index Terms—buses, transportation, bus stops, student

Contents

1 Introduction

THIS document is the report of the LAU Bus Management System project including the design and modeling of the project in addition to the explanation of the software used to build the system and website. The project is based on the frontend and backend development including the database which were connected using the respective web framework. The aim is to provide clear management of the bus transportation system by allowing students to book a seat for every ride on the available buses at the needed timings. Each bus has an assigned route at which there are numerous stops. The website will be accessed only by registered students. The bus management system has a database which has information about students, buses, bus drivers, and locations among others. The database will allow the students to reserve a seat and manage the capacities of every bus. Also, sending requests and messages between the students and bus drivers. Typically, the system will have scanner to be placed at the doors of the bus to detect whether the students who are entering conform with the bookings to keep track of the students who are present and the ones who are missing.

2 BACKGROUND OF THE PROJECT

2.1 Context and Problem Analysis

The project was initially developed because of the problems faced in the bus management at LAU, thus a database to solve the matter was developed. Hence the analysis of what should be included was conducted to solve the problem and decide the functionalities to be included and be provided to the different users including the admin, students, and bus drivers.

The admin will be allowed to:

1. Add registered students to the database
2. Add additional available busses and bus drivers to the database
3. Add additional stops to the database in case they are needed
4. Follow up with the maintenance of the database and the website developed for further updates and edits

The student will be allowed to:

1. Login to the website to view the available booking options available
2. Book a specific seat in a bus and hence automatically presenting the remaining capacity
3. Send messages or complaints to the admin
4. View messages from the bus driver and admin
5. Edit his or her profile including name, number, email, and photo among others

The bus driver will be allowed to:

1. View the different bookings done at the specific time
2. View the names of the students along with their details who booked
3. View the different stops at which the students will be dropped

2.2 Information Needs

The information needed for this project mainly revolve around the students including their ids, names, emails, phone numbers, etc. Other information needed will be regarding the busses available along with their respective descriptions which include seat numbering, capacity, etc. Also, information regarding the drivers includes their phone numbers, names, emails if available, etc. Finally, information regarding the different routes

3 PROPOSED DESIGN

3.1 Conceptual Data Model (CDM)

The following CDM presents the model of our project it includes 9 entities including Student, Seat, Bus_Stop, Route, Destination, Bus, Company, Driver, and Manager. These entities were connected and related using the required relationships which described along with the entities using respective attributes.

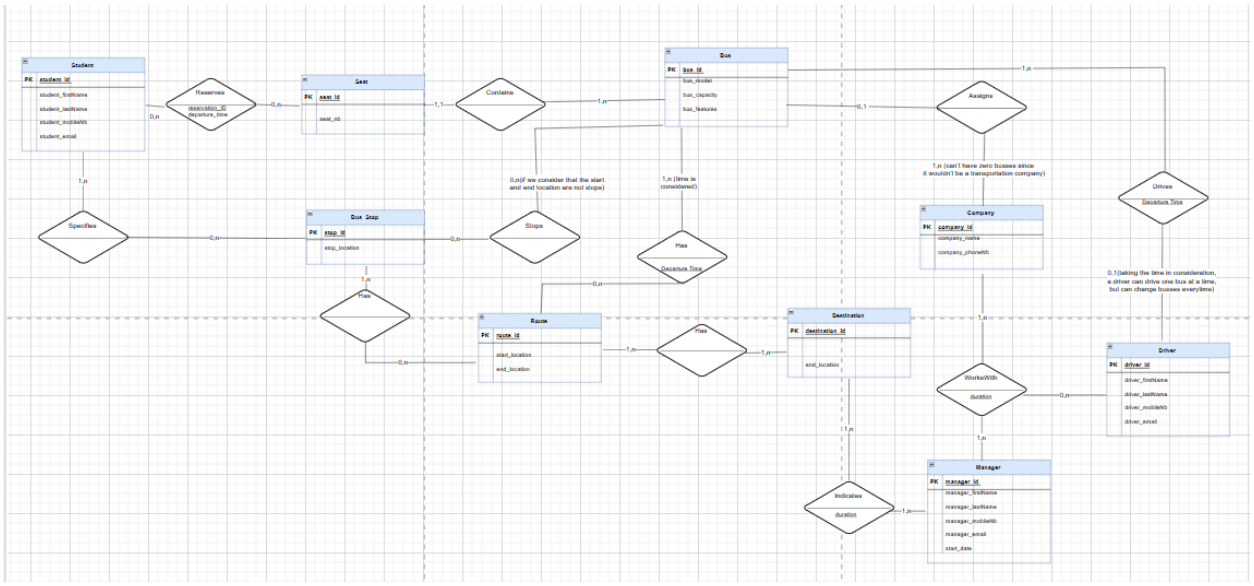

Figure 1: Conceptual Data Model

3.2 Logical Data Model (LDM)

The CDM was converted to obtain the below LDM which was used throughout the analysis, and which followed the rules and regulations discussed in class.

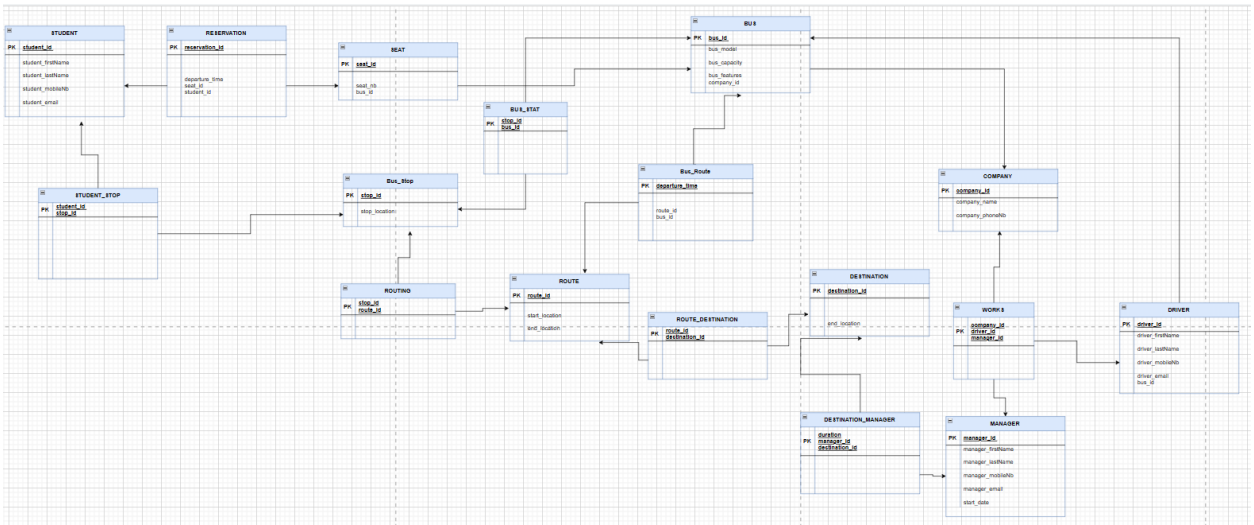

Figure 2: Logical Data Model

3.3 Software and Database Design

The following tools were used in this project:

1. Python
2. Flask
3. MySQL Community Server
4. HTML
5. CSS
6. JavaScript
7. React

Python was used as the main backend language due to its available packages and usefulness in web development. For connecting the developed frontend along with the developed backend, the flask framework was used, some of the packages that were imported and that facilitated our work included SQL Alchemy which helped communicating with the database, flask_login and werkzeug.security to ensure the effectiveness of logging, etc.

The website was designed using HTML and CSS to make the more attractive and systematic. The design includes:

1. Home page
2. Login page
3. Booking page
4. Profile page
5. Confirmation page
6. Complaints page

These pages were linked together using react which served as a library to build a more interactive website.

4 EXPERIMENTAL EVALUATION

In this section, query tests will be included for both the experts and non-expert users. Consequently, non- expert users will be required to specify certain details regarding their spot at which the bus will pick them up at the morning or the spots that the bus will drop them after-noon after bus will depart from LAU Byblos at specific times. In order for the non- expert to provide such data, he is required to choose from multiple drop lists and even filling some textboxes. When it comes to the expert user, the expert user will be the one capable of writing the SQL queries statements in contract to the non-expert individual.

Furthermore, the functionalities of the bus management website will be listed below with brief analysis mentioning the non-expert's input and its corresponding SQL translation (Expert):

1- Sign In:

- Non-Expert: The user will first login to the website by providing 2 his username and password into 2 respective text boxes and based on his credentials, will be logged in as if an admin or a university student
- Expert: The above input will be translated into the following SQL statement:
  SELECT * FROM STUDENT WHERE email= currentmail

1- In case the User was a student he/she will be able to perform the following operations:

a) Pick Up Booking:

- Non-Expert: The student will be provided 3 dropdown lists for the respective city from which he wants the bus to pick him up (Beirut or Tripoli). Based on his selection the second dropdown list will hold the regions in that city that the bus would pass by. Similarly, a third dropdown list will hold the stops in that region. The student is requiring choosing the stop from the respective cascaded dropdown lists.

- Expert: The above input will be translated into the following SQL statement:
  SELECT stop_location FROM BUS_STOP

b) Ride Back Booking:
- Non-Expert: The student will be provided 3 dropdown lists for the respective city from which he wants the bus to drop him (Beirut or Tripoli). Based on his selection the second dropdown list will hold the regions in that city that the bus would pass by. Similarly, a third dropdown list will hold the stops in that region. The student is requires to choose the city, region, and spot from the respective cascaded dropdown lists.
- Expert: The above input will be translated into the following SQL statement:
  SELECT end_location FROM DESTINATION
  SELECT stop_location FROM DESTINATION D, ROUTE_DESTINATION R, ROUTE E, ROUTING I, BUS_STOP S WHERE D.destination_id=R.destination_id AND R.route_id=E.route_id= I.route_id AND I.stop_ID = S.stop_ID AND D.endlocation=specified above

c) Profile Editing:
- Non-Expert: The student will be able to update his phone number, address, email address, and his password by simply filling in textboxes with new updated information. The user cannot update his first and last name nor his ID.
- Expert: The above input will be translated into the following SQL statement:
  UPDATE STUDENT SET student_mobileNb= desired value, student_email= desired value, student_address= desired_value WHERE student_id= acc to student

d) Contacting:
- Non-Expert: The student will be able to get to know the contact details of the logistics department (email and extension) by simply clicking on the contact button.
- Expert: The above input will be translated into the following SQL statement:

  SELECT * FROM MANAGER

2- In case the User was an admin he/she will be able to perform the following operations:

a) Pick Up Capacity:
- Non-Expert: The admin will be able to see the number of students that need to be picked up from each spot and their number (number of students) along with their identification info. This information will enable him to know which bus to select based on the capacity needed when picking students up to university.
- Expert: The above input will be translated into the following SQL statement:
  SELECT student_firstname, student_lastname, student_mobileNb, stop_location FROM STUDENT S, STUDENT_STOP R, BUS_STOP B WHERE S.student_id=R.student_id AND R.stop_id= B.stop_id

b) Ride Back Capacity:
- Non-Expert: The admin will be able to see the number of students that need a seat in the bus in the ride back. The admin will be acknowledged about the number of students that requested this service to choose the appropriate bus(capacity).
- Expert: The above input will be translated into the following SQL statement:

  SELECT COUNT(reservation_id), bus_id FROM RSERVATION R, SEAT S WHERE R.seat_id=S.seat_id GROUP BY S.bus_id

5 CONCLUSION

In conclusion, the LAU Bus Management System was implemented to target the problem being faced in the transportation sector on a small scale and could be extended to a larger scale model. The proposed models along with the design were thoroughly discussed throughout this report and a proposed methodology, used in the project, was presented as well. Additional features which can be implemented as addons to this project were also mentioned to ensure high scale efficient software. Finally, this project served as a learning experience which demanded us to get introduced to the different languages, frameworks, libraries for a functional interactive website. These included but were not limited to python, flask, and react among others.

REFERENCES

[1] **"Flask Tutorial," Tutorials Point. [Online]. Available: https://www.tutorialspoint.com/flask/index.htm. [Accessed: Dec-2022].**

[2] **"Python releases for windows," Python.org. [Online]. Available: https://www.python.org/downloads/windows/. [Accessed: Dec-2022].**

[3] **"Welcome to flask¶," Welcome to Flask - Flask Documentation (2.2.x). [Online]. Available: https://flask.palletsprojects.com/en/2.2.x/. [Accessed: Dec-2022].**