Assignment 3

Description:
A restaurant chain has reached out to your team to build a reservation system.

Here are the details:

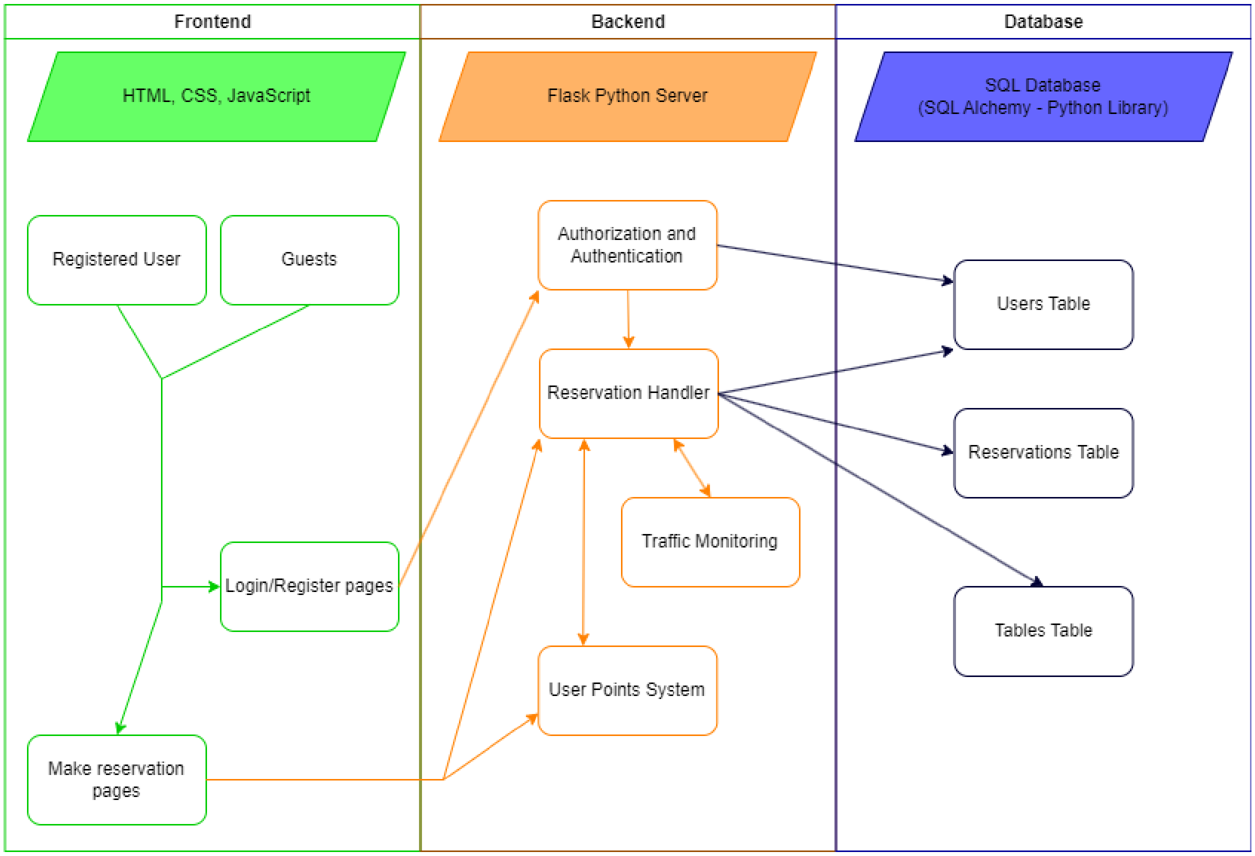
- Two categories of users / customers: guest user or registered user.
- Users should be able to search for a table and reserve.
  - User doesn't need to login to the system to reserve a table. If registered users, they can login.
  - User enters name, phone, email, date and time (date picker), and # of guests for dining and system presents available tables.
  - Tables have maximum capacity limit i.e., 2, 4, 6, or 8.
  - Different combinations are allowed, and owner accommodates the seating, for example: someone requests 8 guests and table for 8 is not available but 2 + 6, or 4+4 is available. System should combine the tables and notify owner they need to combine tables. In this case System reserves both tables.
- If a guest user i.e., not a registered user, system should prompt user to register (Optional) before finalizing the reservation.
- Registered users will have these fields:
  - Name, mailing address, billing address (checkbox if same as mailing address), Preferred Diner # (system generated), Earned points (based on $ spent i.e., $1 is 1 point), preferred payment method (cash, credit, check).
- System should track high traffic days / weekends and a hold fee is required i.e. July 4th will require valid credit card on system to reserve the table.
  - Notify user no show will have minimum $10 charge.

Assumptions:
If you make any assumptions to provide good user experience, please list it.

Answer these questions:

1. Provide detailed software architecture diagram for the proposed solution. Include all modules you will build and any other modules that you will use from other applications / APIs. (15 points)

   Example:
   https://www.edrawsoft.com/software-architecture.html



2. What tools / technologies / frameworks you will use to implement the solution? Discuss in detail. (10 points)
   - Python: we decided to use python to program the backend of the software because it was the best fit so that both team members could contribute efficiently. This is because Caleb and Bryan had previous experience with python.
   - SQL Database: we decided to save all the data for the website on an SQL database. Using a python library called SQL Alchemy
   - Frontend: Javascript, Html, CSS. We decided to create a web application. Therefore the use of javascript HTML and CSS was crucial.
   - Framework: Flask, a python framework. This framework helped set up the backend RESTful infrastructure.

REQUIRED:

Fill in this table, provide as many details as possible:

Group Member Name	What is your contribution?	Discussion Notes
1. Bryan Alvarez	I gave Caleb feedback on the diagram for part 1. And	Divide our software architecture into 3 layers:

	specified the tools/ technologies and frameworks we are using for the project.	1. Frontend - client-side - website interface - achieved by using Javascript, HTML, CSS 2. Backend - server-side - Python framework - Handles reservations 3. Database - user data - reservation and table data - Python
2. Caleb Rogers	Worked with Bryan to construct the software architecture for our project	Divide our software architecture into 3 layers: 1. Frontend - client-side - website interface - achieved by using Javascript, HTML, CSS 2. Backend - server-side - Python framework - Handles reservations 3. Database - user data - reservation and table data - Python
3.		

What to turn in:

- Only soft copy uploaded on or before due date.
- No extensions.
- To get full credit provide details and diagrams (when appropriate).