Software Bid Project Teams

UCS 503- Software Engineering Lab

Team Name: INFERNO

Team ID (will be assigned by Instructor): F

Languages

1. Python

2. Django

3. HTML/CSS, Bootstrap

4. SQLite3

LUNCH ON THE GO

1.1 Project Overview

Students' schedules are jam-packed, and they only have 50 minutes to return to their hostel and eat lunch. Long queues at the mess, the scorching heat, and the monsoon add to the situation.

So, Students usually tend to eat junk food, unnecessarily pay from their pocket, or skip meals.

This not only wastes the food that was already prepared for them but also has a negative impact on their health and finances.

“Lunch on the Go" gives a platform where students can pre-book takeaway meals from their hostels. Instead of waiting in long queues and forgoing lunch due to paucity of time, they can simply pick their packed box from predesignated locations closest to their classroom.

The portal is open for day boarders and faculty. Further, it would allow the user to pick up his/her lunch at the nearest stop.

The interface will be easy to communicate with, listing all the hostels and directly connecting the students to the hostel mess.

1.2 Intended Users

1. College students
2. Day Scholars
3. Faculty

1.3 Functional Requirements

1. **Registration:** The registration function is to register the users into the application. This includes both clients/customers and mess in charge and service providers' employees.
    a. Hostellers,
    b. Day Scholars
    c. Faculty
    d. Mess incharges
    e. Service provider employees

2. **Assigning privileges:** Each of the registered users is assigned privileges as per their role. The categories are:
    a. Day Scholars and Faculty and Students staying in hostels
    b. Service providers in respective hostels- Sodexo or any other

3. **Admin login of each hostel:** Admin login is to perform the following functions-
    a. Mess Manager- to keep inventory.
    b. Mess Secretary- for receiving orders and managing the packing of food.

4. **Authentication:** An authentication using the user id and password is included:

    a. For making sure only authorized students and personnel log in.

    b. It can be upscaled by using the QR functionality.

5. **Predictive location** for closest pickup:

    a. As per the lecture hall location entered by the students, the algorithm suggests the nearest cafe for lunch

    b. Suggestions may be overridden by the user as per convenience.

6. **Payment Gateway :** This module is to make payments and transactions after placing an order.

    a. Hostellers- Nominal charges for delivery and packing, food charges covered under their mess fee.

    b. Day Scholars and Faculty- They will have to pay for both the food and delivery.

    c. May be scaled up to have a credit shell.

7. **Calorie Counter:** To keep track of the nutritional value and calorie intake of the meal consumed.

8. **Daily Menu:** The daily menu gives an overview of the meals prepared in the hostels for the entire week.

    a. As per hostel

    b. Can be edited and managed admin

9. **Feedback and Suggestion portal:** The customers can give their suggestion on the services and meal chosen.

    a. To share grievances about the platform.

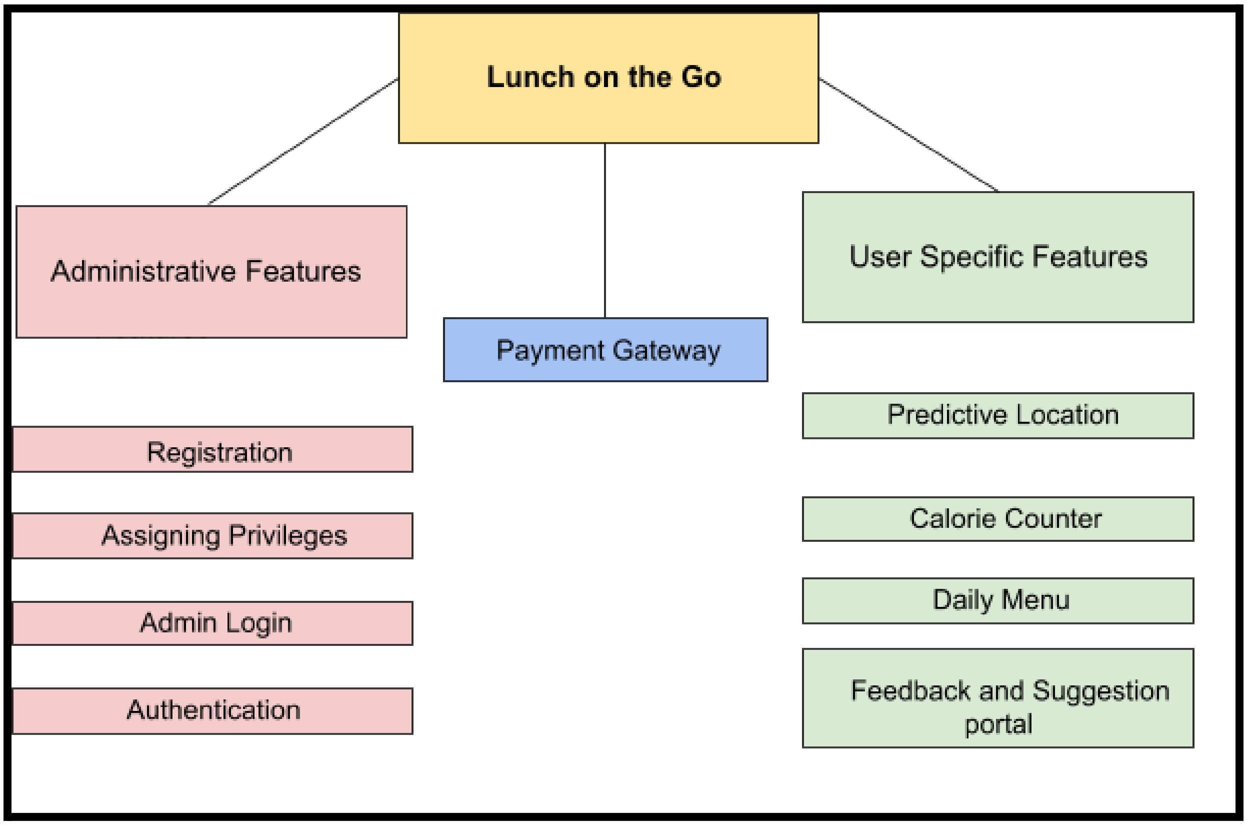
    b. Quality of the food.

    c. Customer care and satisfaction survey.

1.4 Non- Functional Requirements

1. **Availability:** The software should be available 24/7, which means that the user can access the system using the web browser, limited only by the downtime of the server on which the system is running.
2. **Robustness:** Through consistent uptime. This system will be able to stay up and running most of the time. Any downtime would be due to maintenance or upgrades. This downtime also includes any potential failures/crashes.

3. **Ease of use:** The background color for all screens shall be light (such as white) to enhance readability contrast. An option to switch to dark mode will be provided.
4. **Portability:** The application should work on every device and shall have cross-platform compatibility.
5. **Performance Requirements:** The system performance should not be degraded. All web pages shall load within a few seconds. This can be implemented by putting javascript at the bottom of the HTML body. This gives the HTML time to load before any of the JavaScript loads, which can prevent errors and speed up website response time. Processing of each request shall be done within a few seconds.
6. **Security:** User data is not leaked. The data may only be accessed by the admin. The application ensures confidentiality and authentication.
7. **Testability:** The system will be easy to test with the help of test cases described early during the design phase.
8. **Reliability:** The designed software will work smoothly under heavy load. The heavy load is expected during the lunch hours of 1300h to 1500h.


Lunch on the Go
Administrative Features
Payment Gateway
User Specific Features
Predictive Location
Registration
Calorie Counter
Assigning Privileges
Daily Menu
Admin Login
Feedback and Suggestion portal
Authentication