CANTEEN ORDER AUTOMATION SYSTEM

Group 16 *Hardwired*

Acknowledgments

We would like to thank our dear instructor Dr Indranil Saha for giving us an opportunity to work on this project which was a great learning experience. We would also like to thank our TA, Aman Aryan, who mentored and guided us throughout the course of the project.

We also extend our warm gratitude towards other staffs who indirectly have and are helping us work on this project.

Team members

- ★ Ishan Bawne (200456)
- ★ Sharath Kumar V (200916)
- ★ Jhaansi Reddy (200477)
- ★ Rashmi G R (200772)
- ★ Rathod Preet (200775)
- ★ Mohil (200596)
- ★ Mohd Shadab (200593)
- ★ Kavya Jalan (200503)
- ★ Harshit Kumar Tiwari (200432)
- ★ Alaganji Rohan Satwik (150082)

—

Introduction

Our project, Canteen Order Automation System (COAS), is a software which we have implemented using a web application through which the campus residents can remotely order food from canteens of any hall. Also, the website maintains accounts of students facilitating a credit system, allowing students to pay for their orders later.

Why COAS

➔ Reduce time spent by customer at the canteen to order and avoids crowding
➔ Access to menu is not limited
➔ Credits can be maintained and managed efficiently
➔ Easy menu updation by canteen owner
➔ Efficient feedback system through ratings to helping the owner to improve food quality

Potential users:

1. Student community
2. Canteen owners

- Registration and Login:
    - Register new users
    - Login registered users (students and staff) through user id and password
- Viewing menu by the student:
    - Choice of a particular hall
    - View its menu, and details of each item: name, price, veg/non-veg, average rating
- Placing order by student:
    - Add/delete items to cart and choose quantity.
    - Choice of immediate or later payment
- Rate an order and previous orders by student:
    - Can view previous order and allow rating
    - Display due bill to the student

Major Requirements

- Updation of menu by the owner:
  - Add, delete, change status of availability, and edit any menu item.
- View all orders by the owner:
  - View details of every order: item, quantity, customer.
  - Confirm payments
  - Keep track of orders
- Display due bills
  - View due bills of all users by the owner.

Major Requirements

Key Design and Implementation decisions

We have used layered architecture for the design and working of the website. This works well as certain functionalities are performed by certain users and there is a certain way how controls are managed.

As for the coding up the software we have used Python language and used Django framework. We used HTML5, CSS, bootstrap and javascript for the frontend.

We used SQLite, the default database that django works with.

We have used github as it provides version control through git, distributed accessibility and collaboration.

Demonstration

Future Development Plan

- ❏ Include option for delivery to the resident's address in the campus.
- ❏ Add/Delete orders by owner
- ❏ Add options to change password and implement forgot password
- ❏ Include verification during registration
- ❏ Update profile details of a user
- ❏ Give updates to the customer about the processing of her/his order.
- ❏ Give reviews in form of comments
- ❏ Include payment gateway

The Journey

We are glad that this project has provided with a great learning opportunity. We learnt, improved and were exposed to new challenges and here we have emphasized few of them.

- ❖ Software development concepts (practically)
- ❖ Framework: Django
- ❖ Frontend development
- ❖ Automated testing through python 'unittest'
- ❖ Documentation and presentation techniques
- ❖ Teamwork and responsibility

Thank You !!

We would be pleased to take up any sort of queries and inputs