

ONLINE FOOD ORDERING SYSTEM

Project Report

Author: VAIBHAV GUPTA

Registration Number: 25BAI10267

Date: 24th NOVEMBER 2025

1. Introduction

This document details the development and implementation of an Online Food Ordering System. The system provides a platform for restaurants to showcase their menus and for customers to place orders online. This report outlines the project's objectives, technologies used, installation and usage instructions, key features, customization options, example code snippets, and potential future enhancements.

2. Objectives

The primary objectives of this project were to:

- Develop a user-friendly online food ordering platform.
- Provide efficient order management capabilities for restaurants.
- Enable real-time order tracking for customers.
- Offer customization options to cater to diverse restaurant needs.

3. Technologies Used

- Programming Language: PYTHON

4. Installation

- **Install Python:** Download and install Python 3.x from the official website (<https://www.python.org/>) .

- **Download the project file:** Obtain the project file (for example, project.py) and save it to a folder on the computer.
- **No additional dependencies:** If the project uses only built-in modules like input/output, dictionaries, etc., no other packages are needed.

5. Usage

The system is designed for ease of use for both customers and restaurant administrators.

- **Customers:** Can browse menus, add items to their cart, provide delivery address and contact information, and securely checkout using payment gateway integration. Order tracking is available post-checkout.
- **Restaurant Administrators:** Can manage menu items, view incoming orders, update order status, and manage restaurant information such as opening hours and location.

6. Features

Key features of the Online Food Ordering System:

- **User Authentication:** Secure login/registration for customers and administrators.
- **Menu Management:** Restaurants can add, edit, and delete menu item.
- **Responsive Design:** Compatible with various devices (desktops, tablets, and smartphones).
- **Admin Dashboard:** Centralized management console for restaurant administrators.

7. Example Code

Below is an example of the code:

```
#Ordernow - Online Food Ordering System
```

```
|  
#GREETINGS!
```

```
print("!!: Welcome to Ordernow :-!!")
```

```
|  
#define the menu of restaurant
```

```
menu = {
```

```
    "Pizza":150,
```

```
    "Fried Rice":120,
```

```
    "Chicken":200,
```

```
"Roll":100,
```

```
"Burger":80
```

```
}
```

```
print("Pizza:Rs150\nFried Rice:120\nChicken:Rs200\nRoll:Rs100\nBurger:Rs80")
```

```
#order process
```

```
order_total=0
```

```
item_1=input("Enter the item you want to order:").strip().lower()
```

```
if item_1 in menu:
```

```
    order_total+=menu[item_1]
```

```
    print(f"{item_1} added to your order. Price: Rs{menu[item_1]}")
```

```
else:
```

```
    print(f"ordered item {item_1} is not in the menu")
```

```
another_item=input("Do you want to order something else? (yes/no): ").strip().lower()
```

```
if another_item=="yes" or another_item=="y":
```

```
    another_item=input("Enter the item you want to order:").strip().lower()
```

```
    if another_item in menu:
```

```
        order_total+=menu[another_item]
```

```
        print(f"{another_item} added to your order. Price: Rs {menu[another_item]}")
```

```
    else:
```

```
        print(f"ordered item {another_item} is not in the menu")
```

```
else:
```

```
    print("Thank you for your order!")
```

```
    print("Visit again!")
```

```
#salutation
```

```
print(f"your order total is Rs:{order_total}")
```

```
print("Thank you for ordering from ordernow!")
```

```
print("Visit again!")
```

8. Conclusion

The Online Food Ordering System successfully meets the defined objectives, providing a robust and scalable platform for restaurants and customers. The system incorporates essential features such as user authentication, menu management, and secure payment processing. The modular design and clear documentation facilitate future enhancements and customization.

9. Future Work

Potential future enhancements include:

- **Integration with Delivery Services:** Implement direct integration with third-party delivery services (e.g., Uber Eats, Zomato).
- **Advanced Analytics:** Provide detailed sales and customer data analytics for restaurants.
- **Loyalty Programs:** Implement loyalty programs and rewards for frequent customers.
- **AI-Powered Recommendations:** Incorporate AI algorithms to provide personalized menu recommendations.
- **Mobile Applications:** Develop native iOS and Android mobile applications for enhanced user experience.

10. Summary

This report detailed the development of an Online Food Ordering System, covering its objectives, technologies, installation, usage, features, customization, example code, and future work. The project provides a functional platform for restaurants and customers, with potential for future expansion and improvement.