MINI PROJECT
(2021-2022)

Foodies Hub

MID-TERM REPORT

Institute of Engineering & Technology

Submitted by-

Tarun (181500756)

Naman Gupta (181500410)

Amber Srivastava (181500082)

Deepanshu Gupta (181500207)

Nitin Pratap Singh (181500436)

Supervised by-

Mr. Anand Gupta
Technical Trainer
Department of Computer Engineering & Applications

Contents

Abstract

In this mini project, we are building a website using Django, HTML5, CSS3, JavaScript and Bootstrap. After creating this website it will be hosted.
It will be a non-profit website, which can be used by everyone.

This website will solve the problem of those people who wants to promote their services online and people who wants to order food online, as this website will save their time to find each other and also the deal can be done on a good price.
HTML5, CSS3, JavaScript and Bootstrap makes it possible to design a good UI for the users which is easy to use.
Django is used to implement the backend of the website. It delivers users the functionality to login, logout, register, email verification, searching, list food items, upload items and order them**.**

Introduction

The Foodies Hub is a web-based (website) project, which provides solution to the people who want to sell or order food items. This website is being developed for sellers who wants to promote their business by selling their food items at good price and the buyer who wants to order these food products in their budget. The traditional pattern for selling/ordering food items can become a very complex task and time consuming and totally dependent on manual searching technique. After keep in mind all these problems. We make our site that can solve all those problems of such customers easily in an elegant & efficient manner.

Hardware Requirements

- Memory 4GB Ram or higher
- A decent processor
- Windows 8 or higher

Technology Requirements

- Python 3.6
- Django 3.1
- HTML5
- CSS3
- JavaScript
- Bootstrap
- Font Awesome Icons
- Postgresql
- Django Rest Framework

Objectives

- Provides an interactive interface with the help of Html, CSS, JavaScript

and Bootstrap behaves in a sufficient manner that is easy to use and understand.

- All the information regarding food items and users applied online.
- Provide an advanced feature for order management.
- Provide a secured environment for accessing database wherever necessary.
- Provide a module to update the product information.
- To store details of the buyers and sellers.

Implementation Details

- **Part 1: To build Home page.**

With the use of HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons wewill design the home page which will be having a carousel and navigation bar.

- **Part 2: To design Login and Signup form in Django.**

With the help of Django, HTML5, CSS3, JavaScript and Bootstrap we will design the login and signup form. These forms will also be having the server side validation. After validation the data will be used for further use.

- **Part 3: To implement Login, Signup and Logout.**

This will be implemented with Django only. After validating the data from Signupform we will be saving the data in database and after doing so we will register the user.

After validating data from Login form the user will be logged in to the server.

A logout button will be provided, so whenever user wants to logout, it can be done by pressing the button.

- **Part 4: To implement the e-mail verification of user.**

At the time of signup an e-mail with a link is sent to the user's email address. That link is used to verify the user's mail address. When user will click on that link theuser's mail address will be verified**.**

- **Part 5: To build a Shop page.**

With the use of Django, HTML5, CSS3, JavaScript, Bootstrap and Font awesomeicons we will design the shopping page which will be having search bar to search products, and list of searched products.

- **Part 6: To build Food Items page.**

With the use of Django, HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the item page which will be having all the details of food items uploaded by the sellers, and list of all the items purchased/uploaded by the user.

- **Part 7: To implement comment feature.**

With the help of Django we will implement the comment on products feature.

- **Part 8: To implement cart and checkout feature.**

With the use of Django, HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the cart and checkout feature (checkout feature will not be having any payment partner) which will be having all the details of logged in user, editing the profile detail and list of all the products purchased/uploaded by the user**.**

- **Part 9: Created APIs for different functionality.**

We created five APIs for five different models i.e, Food, Order, Shop, Food Comment, All Food, to deal with them and this is achieved by using Django Rest Framework.

Progress

Part 1 is completed.

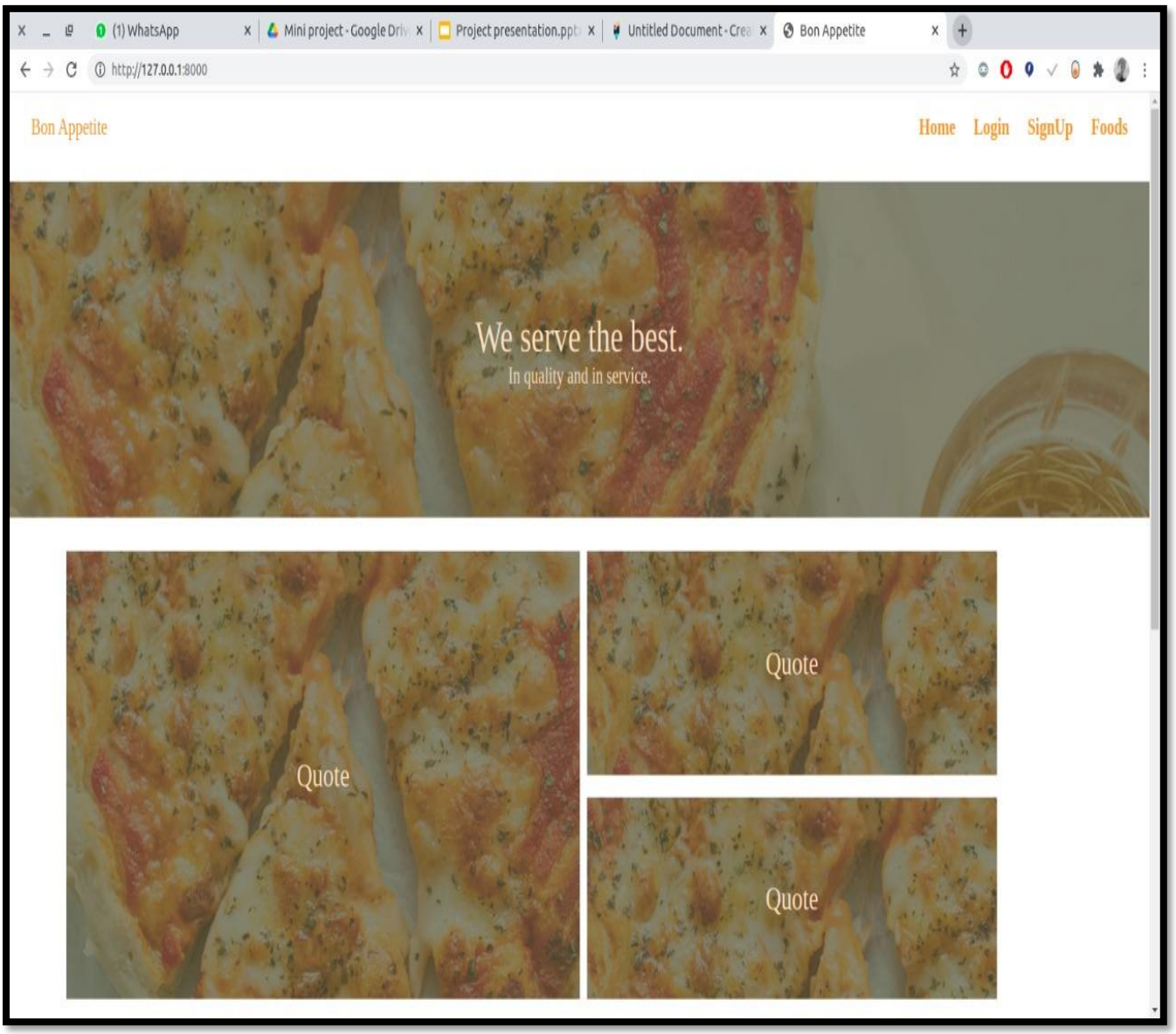
Part 2 is completed.

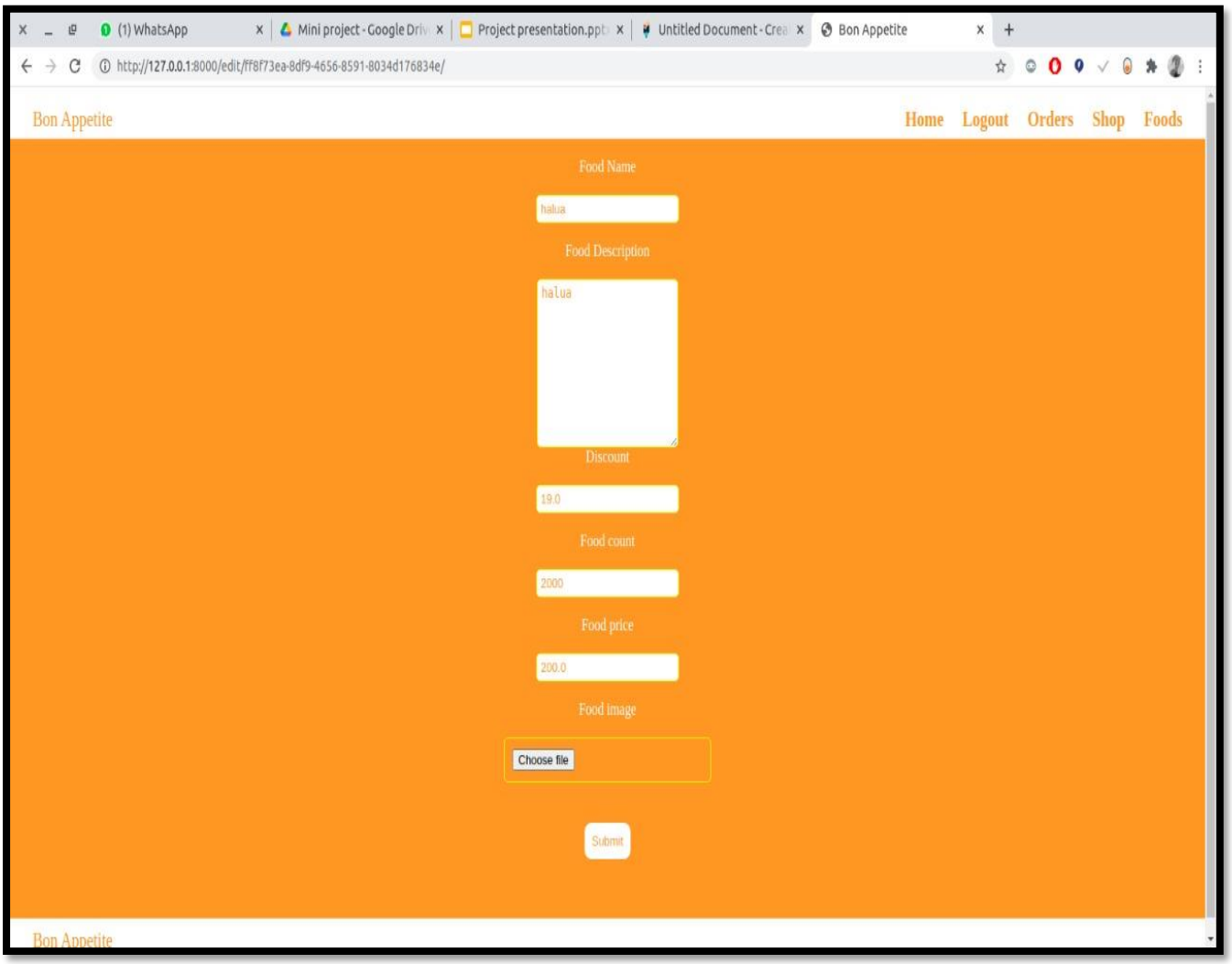
Part 3 is completed.

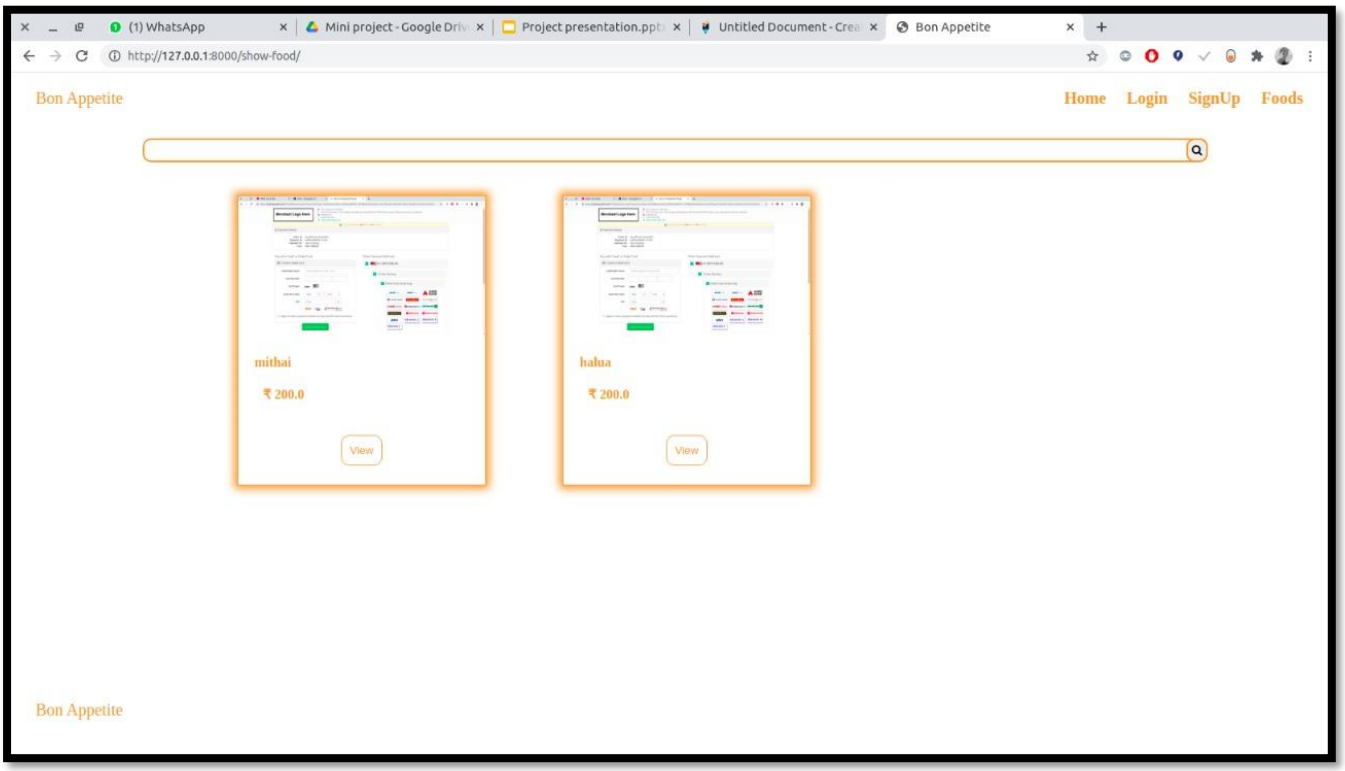
Part 4 is completed.

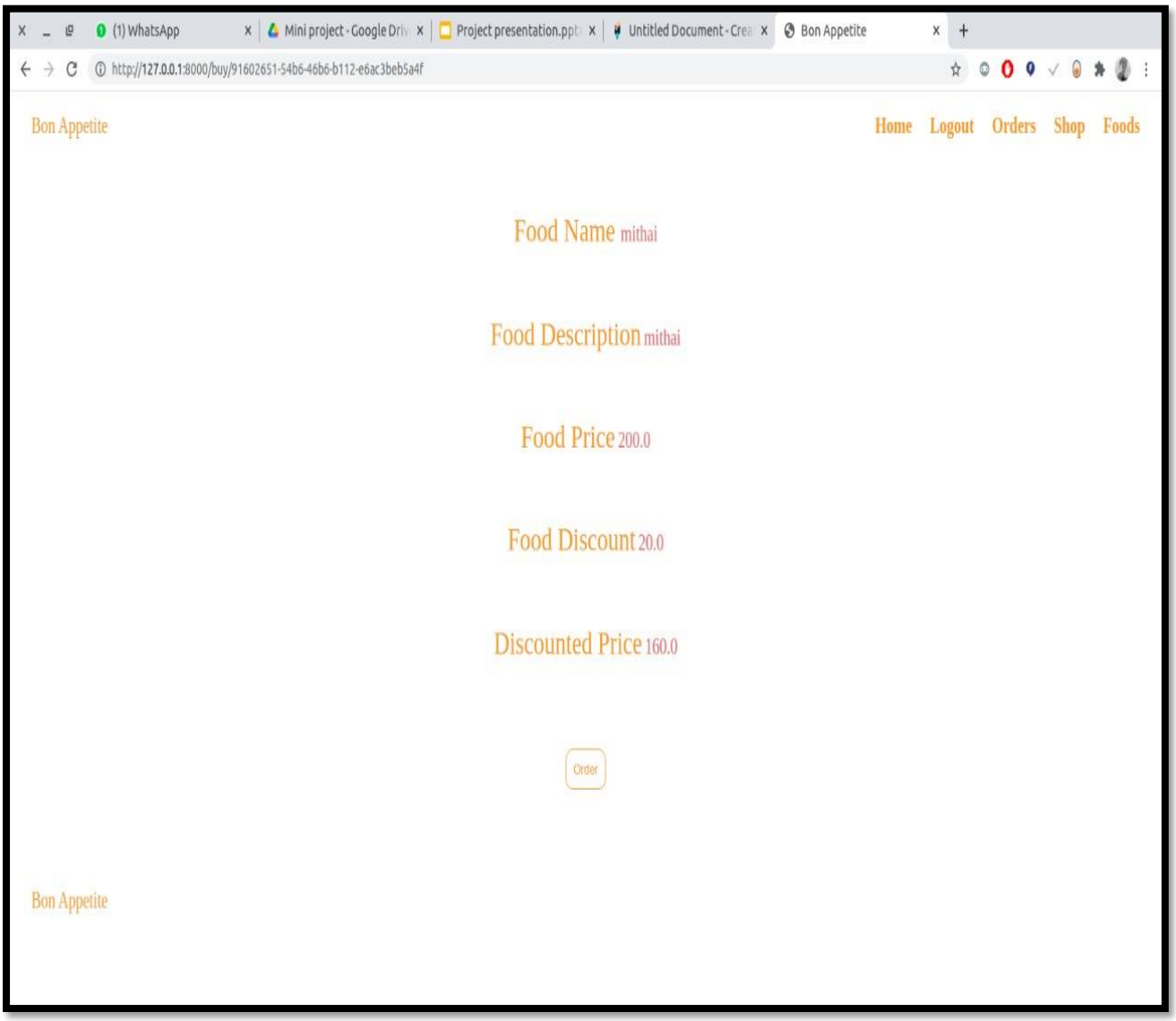
Part 5 is completed.

Screenshots



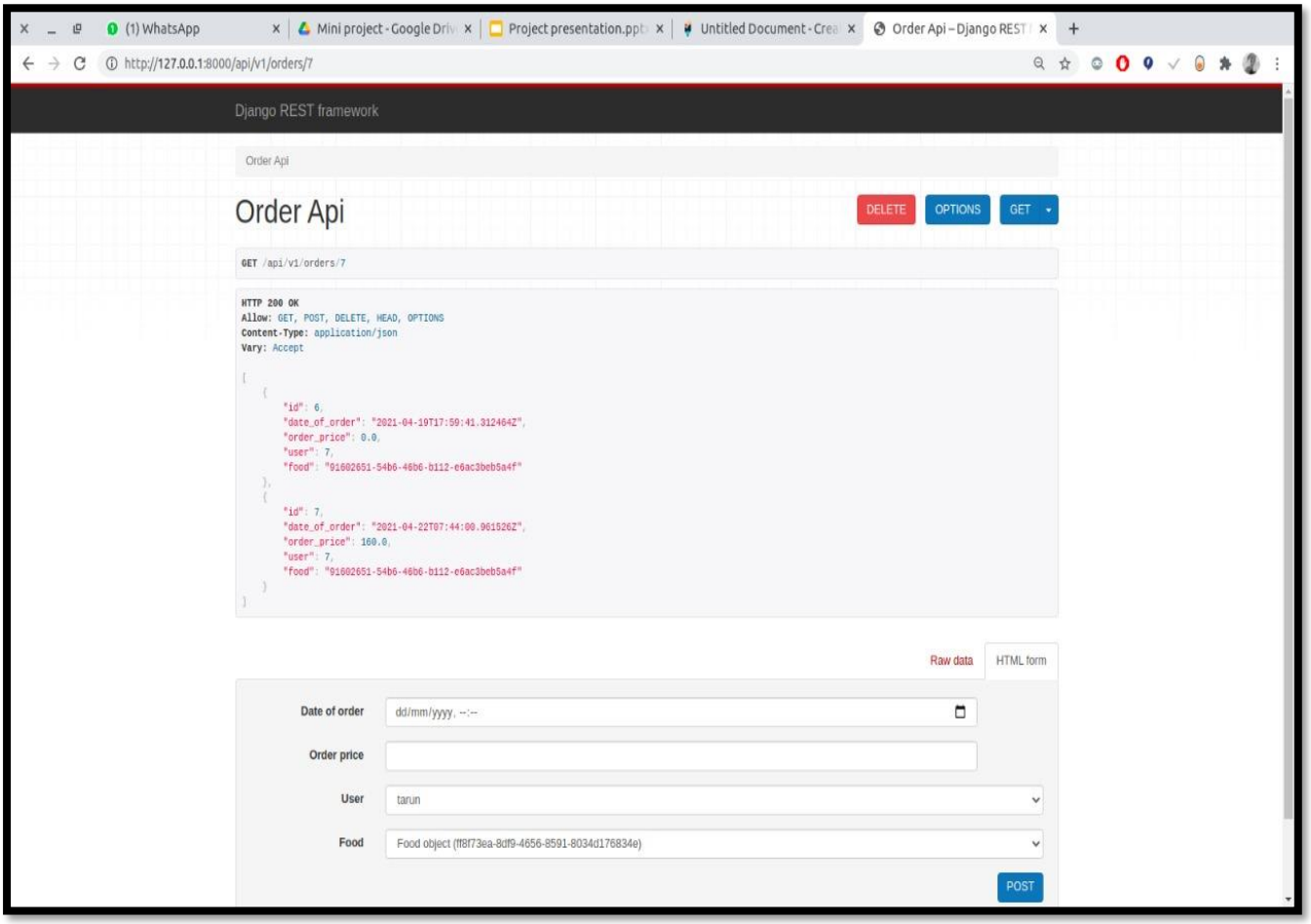

(1) WhatsApp
Mini project - Google Driv
Project presentation.ppt
Untitled Document - Crea
Bon Appetite
http://127.0.0.1:8000/edit/ff8f73ea-8df9-4656-8591-8034d176834e/
Bon Appetite
Home Logout Orders Shop Foods
Food Name
halua
Food Description
halua
Discount
19.0
Food count
2000
Food price
200.0
Food image
Choose file
Submit
Bon Appetite


(1) WhatsApp
Mini project - Google Driv
Project presentation.ppt
Untitled Document - Crea
Bon Appetite
http://127.0.0.1:8000/show-food/
Bon Appetite
Home Login SignUp Foods
mithai
₹ 200.0
View
halua
₹ 200.0
View
Bon Appetite

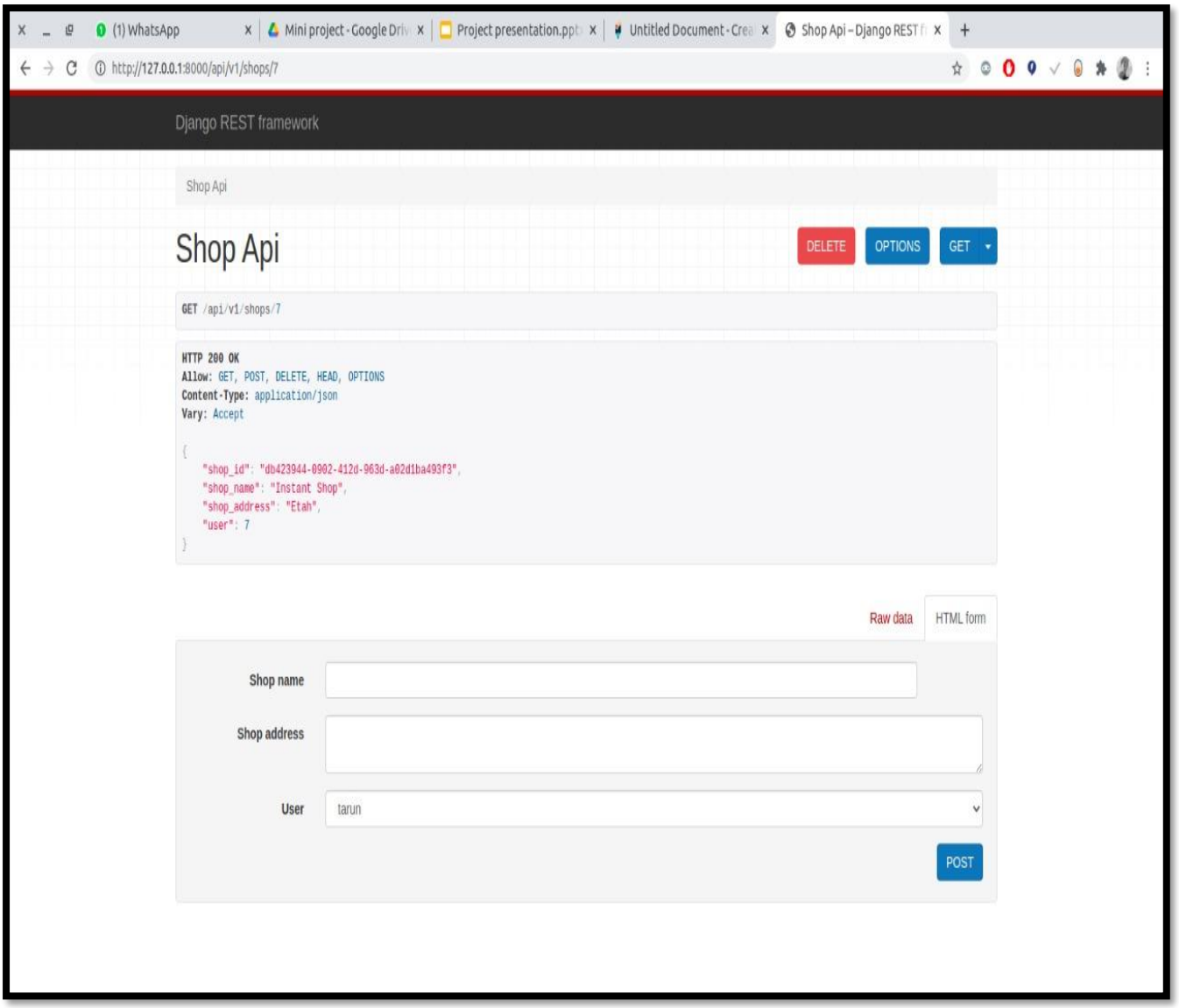

(1) WhatsApp
Mini project - Google Dri
Project presentation.ppt
Untitled Document - Crea
Bon Appetite
http://127.0.0.1:8000/buy/91602651-54b6-46b6-b112-e6ac3beb5a4f
Bon Appetite
Home Logout Orders Shop Foods
Food Name mithai
Food Description mithai
Food Price 200.0
Food Discount 20.0
Discounted Price 160.0
Order
Bon Appetite

APIs

1. **Order API:** This API deals with the Order model.



2. **Shop API:** This API deals with the Shop model.



References

- https://docs.djangoproject.com/en/3.1/topics/db/models/
- https://docs.djangoproject.com/en/3.1/topics/forms/
- https://docs.djangoproject.com/en/3.1/topics/class-based-views/
- https://docs.djangoproject.com/en/3.1/topics/http/urls/
- https://www.w3schools.com/html/
- https://www.w3schools.com/css/
- https://getbootstrap.com/