

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JNANA SANGAMA”, BELAGAVI-590018, KARNATAKA



A Mini Project Report
On

“ONLINE SHOPPING WEBSITE”

Submitted in the partial fulfillment of the requirement for the completion of **Web Technology Laboratory with Mini Project (17CSL77)** and award of degree of

**BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING**

Submitted By

**SATWIK RAM K
SHAHBAZ KHAN Z
N PAVAN KUMAR**

**1VA17CS047
1VA17CS048
1VA17CS028**

Under the Guidance of

**Mr. Nagamahesh B S
Assistant Professor
Dept. of CSE, SVIT**

**Mrs. Sreelatha P K
Assistant Professor
Dept. of CSE, SVIT**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SAI VIDYA INSTITUTE OF TECHNOLOGY**

(Affiliated to Visvesvaraya Technological University, Belagavi | Recognized by Govt. of Karnataka | Approved
by AICTE, New Delhi)

RAJANUKUNTE, BENGALURU – 560 064

2020-21

SAI VIDYA INSTITUTE OF TECHNOLOGY

Rajanukunte, Bengaluru- 560 064

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

Certified that the Mini project work entitled “*ONLINE SHOPPING WEBSITE*” carried out by **Mr. SATWIK RAM K (1VA17CS047)**, **Mr. SHAHBAZ KHAN Z (1VA17CS048)**, and **N PAVAN KUMAR (1VA17CS028)** bonafide students of **SAI VIDYA INSTITUTE OF TECHNOLOGY**, Bengaluru, in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of **VISVESVARAYA TECHNOLOGICAL UNIVERSITY**, Belagavi during the year **2020-21**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Mini Project work prescribed for the **Web Technology Laboratory with Mini Project (17CSL77)**.

Mr. Nagamahesh B S

Assistant Professor,
Dept. of CS&E, SVIT

Dr. Archana R A

HOD
Dept. of CS&E, SVIT

Dr. H S Ramesh Babu

Principal

External Viva:

Name

Signature

1. _____
2. _____

TABLE OF CONTENTS

Chapters	Page No
ACKNOWLEDGEMENT	i
ABSTRACT	ii
1. INTRODUCTION TO WEB TECHNOLOGY	1
2. SYSTEM REQUIREMENTS SPECIFICATIONS	7
3. SYSTEM DESIGN	9
4. IMPLEMENTATION	12
5. TESTING AND RESULTS	15
6. CONCLUSION AND FUTURE ENHANCEMENT	17
BIBILOGRAPHY	19
APPENDIX - A	
 SNAPSHOTS	20

ACKNOWLEDGEMENT

The completion of Project brings with and sense of satisfaction, but it is never completed without thanking the persons who are all responsible for its successful completion. First and foremost, we wish to express our deep sincere feelings of gratitude to our Institution, **Sai Vidya Institute of Technology**, for providing us an opportunity to do our education.

We would like to thank the **Management, Prof. M R Holla**, Director, Sai Vidya Institute of Technology and **Prof. A M Padma Reddy**, Director (A), Sai Vidya Institute of Technology for providing the facilities.

We extend our deepest sense of sincere gratitude to **Dr. H S Ramesh Babu**, Principal, Sai Vidya Institute of Technology, Bengaluru, for having permitted us to carry out the project work on **ONLINE SHOPPING WEBSITE**.

We express our heartfelt sincere gratitude to **Dr. Archana R A**, Professor and Head, Department of Computer Science and Engineering, Sai Vidya Institute of Technology, Bengaluru, for her valuable suggestions and support.

We express my special in-depth, heartfelt, sincere gratitude to **Mr. Nagamahesh B S**, Assistant Professor, Department of Computer Science and Engineering, and **Mrs. Sreelatha P K**, Assistant Professor, Department of Computer Science and Engineering, Sai Vidya Institute of Technology, Bengaluru for their constant support in completing the project.

Finally, we would like to thank all the Teaching, Technical faculty and supporting staff members of Department of Computer Science and Engineering, Sai Vidya Institute of Technology, Bengaluru, for their support.

SATWIK RAM K	1VA17CS047
SHAHBAZ KHAN Z	1VA17CS048
N PAVAN KUMAR	1VA17CS028

ABSTRACT

The *ONLINE SHOPPING WEBSITE* is a web-based shopping system for an existing shop. The project objective is to deliver the online shopping application into an android platform as well. This project is an attempt to provide the advantages of online shopping to customers of a real shop. It helps to buy the products in the shop anywhere through the internet by using an android device. Thus, the customer will get the service of online shopping and home delivery from his favorite shop. *ONLINE SHOPPING WEBSITE* can be implemented for any shop in the locality or to multinational branded shops having retail outlet chains. If shops are providing an online portal where their customers can enjoy easy shopping from anywhere, the shops won't be losing any more customers to the trending online shops such as Flipkart or amazon. Since the application is available on the Smartphone it is easily accessible and always available.

The *ONLINE SHOPPING WEBSITE* aims to develop an online shopping for customers with the goal so that it is very easy to shop your loved things from an extensive number of online shopping sites available on the web. With the help of this, you can carry out online shopping from your home. Here is no compelling reason to go to crowded stores or shopping centers during festival seasons. To get to this online shopping system all the customers will need to have an email and password to log in and proceed with their shopping. Upon successful login, the customers can purchase a wide range of things such as mobiles, books, apparel, jewelry, infant care, gifts, tools, etc. can be dispatched using an online shopping system. It is extremely secure. Customer service is accessible

CHAPTER 1

INTRODUCTION

1.1 Overview

Web Technology is a general term referring to many languages and multimedia packages that are used in conjunction with one another, to produce dynamic websites. The website has two modules namely, server side and the client side. Growth of Internet community carrying out businesses on the Internet has been very rapid since last few years following a per-defined path or strategy to earn revenue from the net. One such strategy is to create, market and manage a website which attracts huge masses of people.

This project aims to develop an online shopping for customers with the goal so that it is very easy to shop from an extensive number of online shopping sites available on the web. With the help of this, online shopping can be carried from your home. To get to this online shopping system the users will need to have an email and password to log in and proceed with their shopping. Upon successful login, the user can purchase a wide range of things such as mobiles, books, apparel, jewellery, infant care, gifts, tools, etc. can be dispatched using an online shopping system. And of course, requested ordered items at doorstep. It is simple. It has payment making options like net banking, credit card, debit card, or UPI.

Almost a wide range of things can be brought through an online shopping system. The user can purchase goods from anywhere and goods will be delivered to their houses. It is extremely secure. Customer service is accessible. This application is implemented using popular web technologies: HTML, CSS and JavaScript, Django for client and server-side scripting respectively.

1.2 Problem Statement

This project is an attempt to provide the advantages of online shopping to customers of a real shop. It helps buying the products in the shop anywhere through internet by using an android device. Thus, the customer will get the service of online shopping

and home delivery from their favourite shop. This system can be implemented to any shop in the locality or to multinational branded shops having retail outlet chains.

1.3 Motivation

Internet has seen a tremendous growth in terms of coverage and awareness. So, giving the business online presence has become very crucial and important. Consumers of today are attracted to placing an order online and make purchases because it is very convenient and extremely simplified. The main objective is to develop an interface that is user friendly and target a particular audience that is in need for such an application. This has been a key factor of motivation behind this venture. There are several languages in Web Technology which helps in designing user friendly and interactive websites. Each language has its own strengths, which help in styling and designing the client side. The server side includes the connectivity to the database and passing of relevant data to the client side.

1.4 Web technologies

In the IT world, the internet is an essential platform, whether it's for developing or for consumer use. When developing a website, in order to make the websites look and function a certain way, web developers utilize different languages. The Online Green Mart is developed using the following web technologies.

Hypertext Markup Language (HTML)

HTML is the standard Markup language for creating web pages and web applications. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as interactive forms, may be embedded into the rendered page.

HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

HTML elements are delineated by *tags*, written using angle brackets. HTML can embed programs written in a scripting language such as JavaScript which affect the behaviour and content of web pages. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Technology used

by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

Cascading Style Sheets (CSS)

CSS is a style sheet language used for describing the presentation of a document written in a Markup language. CSS defines the look and layout of content. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document.

CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enabling multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Changes to the graphic design of a single document or many can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing Markup in the documents.

JavaScript (JS)

JS is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript is designed for creating network-centric applications and is complementary to and integrated with Java and HTML. Client-side JavaScript is the most common form of the language.

The script should be included in or referenced by an HTML document for the code to be interpreted by the browser. It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content.

It has a lot of advantages. It is platform independent. Coding using Java makes it an object-oriented scripting language.

Django Rest Frameworks

Django REST framework is a powerful and flexible toolkit for building REST Full APIs which can be used in both Web and Other Platforms. The main advantages of Django Rest frameworks are:

- ✓ The Web browsable API is a huge usability win for your developers.
- ✓ Authentication policies including packages for OAuth1a and OAuth2.
- ✓ Serialization that supports both ORM and non-ORM data sources.
- ✓ Customizable all the way.
- ✓ Extensive documentation, and great community support.
- ✓ Used and trusted by internationally recognised companies including Mozilla, Red Hat, Heroku, and Eventbrite.

1.5 Applications of web technologies

Real time web technologies have been around for over 10 years, but it's only relatively recently that people have started seeing them used in the sorts of applications used on a daily basis. Real time web technologies have a number of common use cases, and newer, innovative uses are constantly being discovered.

Simple use cases are things like displaying data, statistics, notifications and news as soon as it becomes available. But the technology shows its real value when interactive experiences are created - when multiple users and systems are instantly communicating with each other. Industries which are benefitting from the technology include social, broadcasting, sports, finance, e-commerce, energy, education, healthcare and gaming.

Applications of real time web technologies are:

Real time web analytics

Google has started pushing into this market with Google Analytics. There is a ton of opportunity here, as there is the ability to capture and transmit interactive user data like never before. From tracking where the mouse floats across the screen, the orientation of a mobile device, and any number of other user interactions. Not only track them, but actually watch them as they happen.

Digital Advertising

The digital advertising world has long been a solid revenue generator for web-based businesses. With real-time web technologies, advertisers can move towards more

interesting advertising paradigms, such as charging for ads based on the amount of time the ad is visible on a user's screen or other real time interaction metrics versus the CPM and CPC metrics that have long been the standard

E-Commerce

E-commerce has always been a hotbed for engaging customers and customer interactions. Showing shoppers what other shoppers are looking at online, or pushing out online deals directly to all connected browsers are the types of real time features that e-commerce platforms are adopting.

Publishing

Keeping eyeballs on the screen is the primary goal on any online publisher, and one way to keep visitors on the site is to keep them engaged.

Realtime data can lead to some very interesting info graphics, and it can also help connect viewers like never before. A great example is some of the real time comments like Disqus.

Backend Services and Messaging

The backend systems have grown in scale and complexity over the last decade, and it is becoming increasingly important to propagate messages across very large systems effectively and efficiently. The real time web is going to be great for these types of functions, Project Management & Collaboration. Google docs and other platforms have already demonstrated the value (and potential complexity) in implementing real time collaborative environments on the Web.

The new era of real time web technologies will make the development of these types of applications simpler and easier to build which is great because most web applications are not built in a vacuum, so having the ability to connect all those users together in constructive and insightful ways should be able to add value in their workflows.

Real time Monitoring Services

The bi-directional communication channel is great for remote devices or servers to stay connected to a central monitoring service. This gives techs and admins the ability to watch what their endpoints are doing in real time without logging into the machine, and also gives the ability to send real time alerts.

Live Charting and Graphing

Charts and graphs have always been a great way to visualize data. Now those graphs and charts can be connected to real time data flows. The possibilities are literally endless, from displaying temperature data measured from a connected home device to streaming stock prices to real-time chart.

Group and Private Chat

Chat has long been the de-facto example for real-time since by its very nature it requires real-time bi-directional communication. Chat is a great use for real-time web technologies, and variations on group chats and private chats leads to more innovation in this area.

CHAPTER 2

SYSTEM AND SOFTWARE REQUIREMENTS AND SPECIFICATIONS

The program works on Desktop PC and is executed using a Django Rest Frameworks interface which interacts with a PostgreSQL database running on localhost.

2.1 FUNCTIONAL REQUIREMENTS

A description of the facility or feature required. Functional requirements deal with what the system should do or provide for users. They include description of the required functions, outlines of associated reports or online queries, and details of data to be held in the system.

2.1.1 Interface Requirements:

- The system shall provide an option to add/delete products to their cart.
- The system should give option for login.
- The system shall provide option to add/delete suppliers.
- The system shall provide option Search a product.
- The system shall provide option to sell a product.

2.2 NON-FUNCTIONAL REQUIREMENTS:

Non-functional requirements define the overall qualities or attributes of the resulting system.

2.2.1 Usability

Usability is the ease with which a user can learn to operate the Online Shopping Website system and get results.

2.2.2 Security

Security requirements are included in a system to ensure:

- All inventory and supplier information are well secured
- SQL injection is prevented

2.2.3. Reliability

Reliability is the ability of a system to perform its required functions under stated conditions for a specific period of time. Constraints on the run-time behavior of the system can be considered under two separate headings:

- Availability: is the system available for service when requested by end-users.
- Failure rate: how often does the system fail to deliver the service as expected by end-users.

2.2.4. Efficiency

The comparison of what is actually produced or performed with what can be achieved with the same consumption of Clouds (money, time, labor, etc.). It is an important Factor in Determination of Productivity.

2.2 SOFTWARE REQUIREMENTS

Programming language	:	Python
Operating system	:	ANY OS (Recommended: Windows 10, Windows Vista, Windows XP)
Application required	:	Web Application
Coding language	:	Core Python OOPS

2.3 HARDWARE REQUIREMENTS

CPU	:	Pentium IV 2.4 GHz
Memory (Primary)	:	512 MB, 1 GB or above
Hard Disk	:	40 GB, 80GB, 160GB or above
Monitor	:	15 VGA color

CHAPTER 3

SYSTEM DESIGN

System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

3.1 Proposed System

This *ONLINE SHOPPING WEBSITE* is a web-based shopping system for an existing shop. This project aims to develop an online shopping for customers with the goal so that it is very easy to shop from an extensive number of online shopping sites available on the web. With the help of this, online shopping can be carried from your home. To get to this online shopping system the users will need to have an email and password to log in and proceed with their shopping. Upon successful login, the user can purchase a wide range of things such as mobiles, books, apparel, jewellery, infant care, gifts, tools, etc. can be dispatched using an online shopping system. And of course, requested ordered items at doorstep. It is simple. It has payment making options like net banking, credit card, debit card, or UPI. Almost a wide range of things can be brought through an online shopping system. The user can purchase goods from anywhere and goods will be delivered to their houses. It is extremely secure. Customer service is accessible.

The *ONLINE SHOPPING WEBSITE* objective is to deliver the online shopping application into an android platform. This project is an attempt to provide the advantages of online shopping to customers of a real shop. It helps to buy the products in the shop anywhere through the internet by using an android device. Thus, the customer will get the service of online shopping and home delivery from his favourite shop. This system can be implemented for any shop in the locality or to multinational branded shops having retail outlet chains. If shops are providing an online portal where their customers can enjoy easy shopping from anywhere, the shops won't be losing any more customers to the trending online shops such as Flipkart or amazon. Since the application is available on the Smartphone it is easily accessible and always available.

3.2 Flow of web pages

A flowchart is a formalized graphic representation of a logic sequence, work or manufacturing process, organization chart, or similar formalized structure. A website flowchart shows the structure and makeup of any existing or planned website. They are useful for giving a broad overview of a website's content as well.

Flow charts of the webpages are a useful tool in these situations, as they make a process easy to understand at a glance. Using just a few words and some simple symbols, they show clearly what happens at each stage and how this affects other decisions and actions.

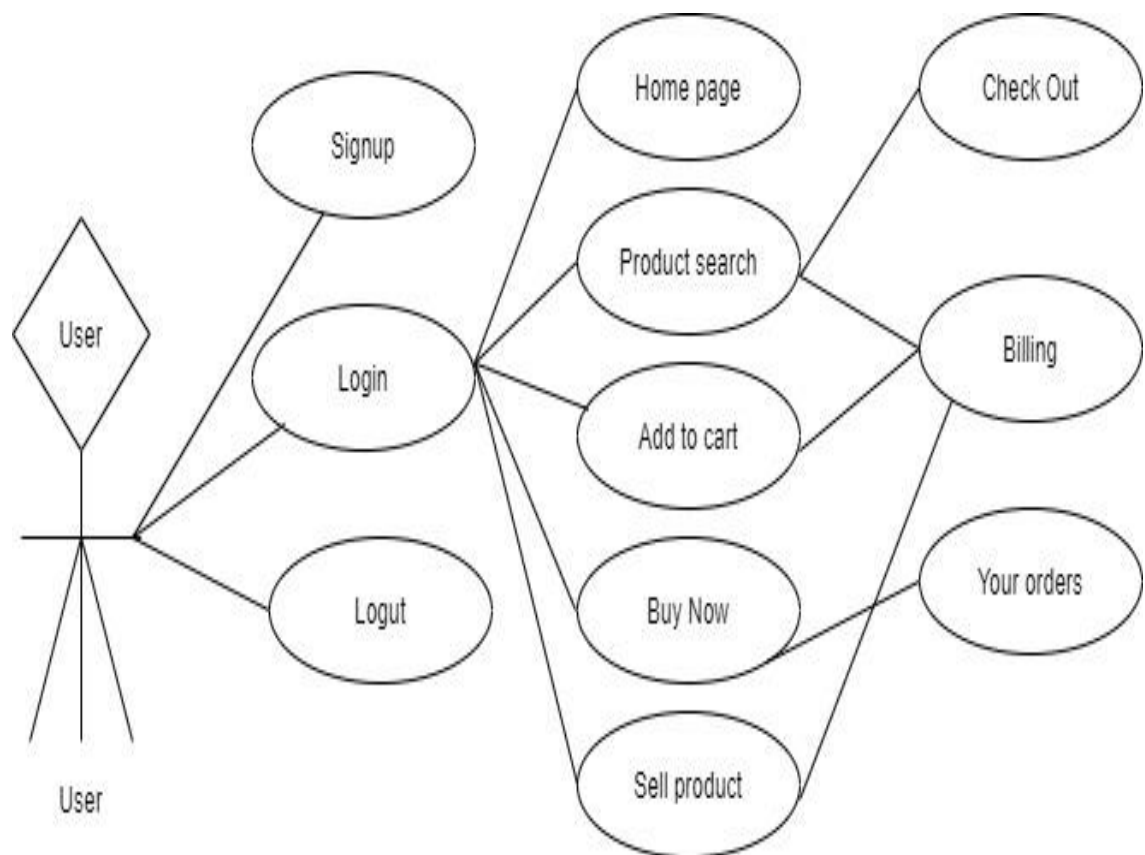


Fig 3.2 Flow of web pages – User

3.3 ER Diagram

The ER diagram below shows the relationship between the many tables that exist in the database for the functioning of Online Shopping Website.

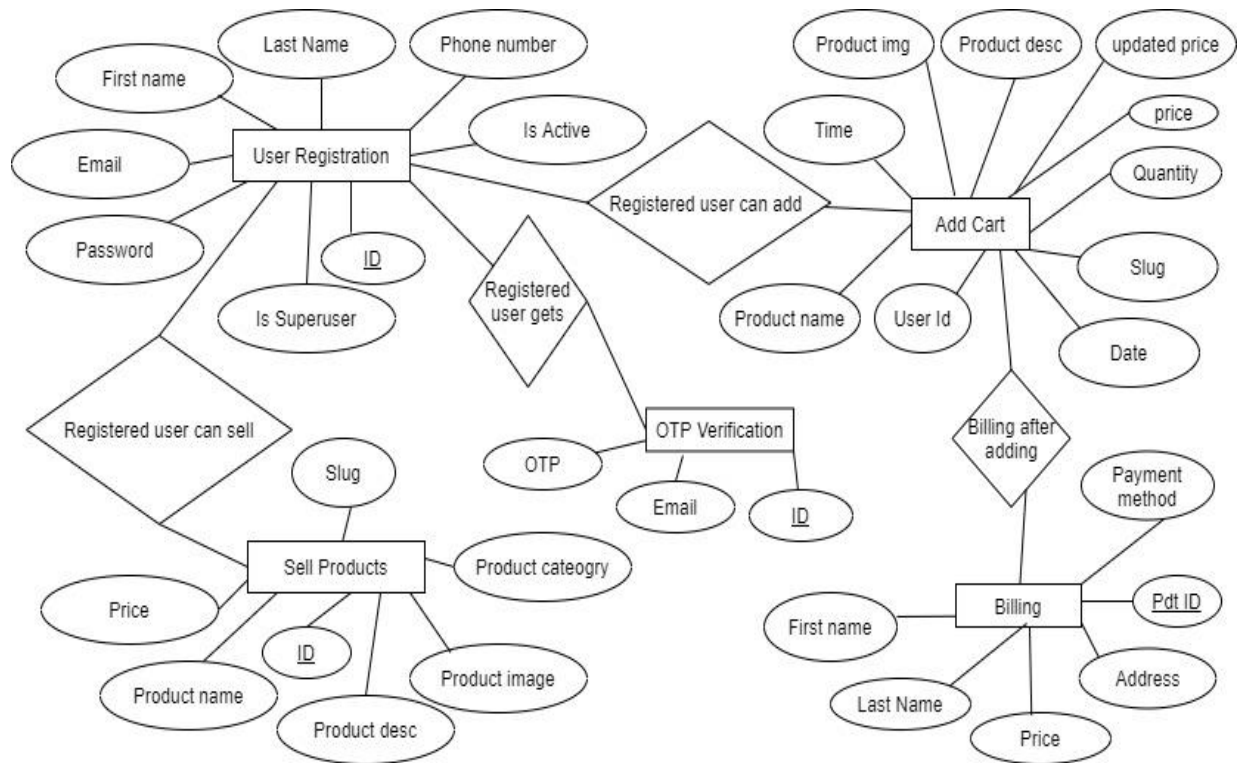


Fig. 3.3 ER Diagram

CHAPTER 4

IMPLEMENTATION

4.1 Module description

4.1.1 User Registration:

This module is used for user registration. The users have to enter the required details and can register to the portal through their email or mobile phone.

4.1.2 OTP:

This module is used for the registered user email verification via OTP that is sent to their respective email ids that were used while registering.

4.1.3 Sell Product:

The register users can sell the products through this module. The user has to enter the required details of the product like name, description and price here and can sell their products.

4.1.4 Add to Cart:

The register users can search for the products and add them to their cart for buying the product later through this module.

4.1.5 Billing Details:

The register user can buy the products which are there on the website. Before proceeding to the checkout the billing details are saved using this module.

4.1.6 Orders :

The register user can view their orders if they have made any orders in the past and can track their delivery status through this module.

4.2 Source Code

4.2.1 Sell Product Model

```
class SellProduct(models.Model):

    slug = models.SlugField(unique = True, max_length=250, null=True,
blank=True, editable = False)
    price = models.FloatField(max_length = 10)
    product_name = models.CharField(max_length = 50)
    product_image = models.CharField(max_length = 100, null = True)
    product_des = models.TextField(max_length = 1000, blank = True)
    product_category = models.CharField(max_length = 50, blank = True)
    date = models.CharField(max_length = 20)
    time = models.CharField(max_length = 20)

    def __str__(self):
        return self.product_name

def rl_pre_save_receiver(sender, instance, *args, **kwargs):
    if not instance.slug:
        instance.slug = unique_slug_generator(instance)

pre_save.connect(rl_pre_save_receiver, sender = SellProduct)
```

4.2.2 Add to Cart Rest API

```
class CartAPIView(APIView):

    def get_object(self, user):
        try:
            details = AddCart.objects.all().filter(user = user).values()
            return details
        except AddCart.DoesNotExist:
            return HttpResponse(status = status.HTTP_404_NOT_FOUND)

    def get(self, request, user):

        details = self.get_object(user)
        serializer = CartSerializer(details, many = True)
        return Response(serializer.data)

    def post(self, request, user):

        serializer = CartSerializer(data = request.data)

        if serializer.is_valid():
            slug = serializer.validated_data['slug']
            price = serializer.validated_data['price']
            print(slug, price)

            if AddCart.objects.filter(user = user).exists():
                if AddCart.objects.filter(slug = slug).exists():
```

```

        quantity = AddCart.objects.filter(user = user, slug = slug).values('quantity')[0]['quantity']
        print(quantity)
        quantity = quantity + 1
        updated_price = quantity * price
        AddCart.objects.filter(user = user, slug = slug).update(quantity = quantity, updated_price = updated_price)
        print("Updated Successfully")
        return Response(serializer.data, status=status.HTTP_201_CREATED)
    else:
        serializer.save()
        print("Added Successfully")
        return Response(serializer.data, status=status.HTTP_201_CREATED)
    else:
        serializer.save()
        print("Added Successfully")
        return Response(serializer.data, status=status.HTTP_201_CREATED)
    return JsonResponse(serializer.errors, status=status.HTTP_400_BAD_REQUEST)

def delete(self, request, slug):
    user = request.user
    slug = self.slug
    if AddCart.objects.filter(user = user, slug = slug).exists():
        AddCart.objects.filter(user = user, slug=slug).delete()
        print("Successfully Deleted!")
        return Response(status=status.HTTP_204_NO_CONTENT)
    return JsonResponse(status=status.HTTP_400_BAD_REQUEST)

```

4.2.3 Consuming Rest API in Django Rest Frameworks

```

def cart(request, user):
    if user == 'AnonymousUser':
        return HttpResponse("Please Login to View the Cart.")
    else:
        url = 'http://127.0.0.1:8000/add-to-cart/'+user
        response = requests.get(url = url)
        results = response.json()
        print("obtained Result is",results)
        total = 0
        for result in results:
            total += result['updated_price']

        subtotal = total
        if total <1000 and total > 0:
            shipping = 100
            total = total + shipping
        else:
            shipping = 0
        return render(request, "cart.html", {"results": results, "total": total, "subtotal": subtotal, "shipping": shipping})

```

CHAPTER 5

TESTING AND RESULTS

After all phase have been perfectly done, the system will be implemented to the server and the system can be used.

5.1 SYSTEM TESTING

The goal of the system testing process was to determine all faults in the project. The program was subjected to a set of test inputs and many explanations were made and based on these explanations it will be decided whether the program behaves as expected or not. Our project went through three levels of testing.

1. Unit testing
2. Integration testing
3. Validation testing

5.1.1 UNIT TESTING

Unit testing is commenced when a unit has been created and effectively reviewed. In order to test a single module, we need to provide a complete environment i.e., besides the section we would require

- ✓ The procedures belonging to other units that is the unit under test calls.
- ✓ Non local data structures that module accesses.
- ✓ A procedure to call the functions of the unit under test with appropriate parameters.

Test for the admin module

Testing admin login form: This form is used for log in of administrator of the system. In this form we enter the username and password if both are correct administration page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask the details.

5.1.2 INTEGRATION TESTING

In the Integration testing we test various combination of the project module by providing the input. The primary objective is to test the module interfaces in order to confirm that no errors are occurring when one module invokes the other module.

5.1.3 VALIDATION TESTING

The whole of validation testing is to expose latent defects in a software system before the system is delivered. This contrasts with validation which is intended to demonstrate that a system meets its specification. Validation testing requires the system to perform correctly using given acceptance test cases.

CHAPTER 6

CONCLUSION

The development of Shopping website application involved many phases. The approach used is a top-down, concentrating on what first, then how and moving to successive levels of details. The first activity serves as a basis of giving the functional specifications and then successful design of the proposed system. With the growth of internet, netizens prefer to view and purchase items on their fingertips.

The design phase was concerned primarily with the specification of the system elements in manner that best meet the needs. The main objective was to create an interface that best meet the user requirements by providing an easy navigation with user interactions. To implement this design, a computer program was written in HTML, JS and Django. The styling was implemented in CSS.

It is hoped that effective implementation of this software application would eliminate the many problems of the existing system. It is convenient for users as this system provides accurate cost and description of the system. The website is flexible for use. There is human interaction also when some help needed. There can be no fraud and security problems.

CHAPTER 7

FUTURE ENHANCEMENT

Here are the future Enhancements for the Online Shopping Website

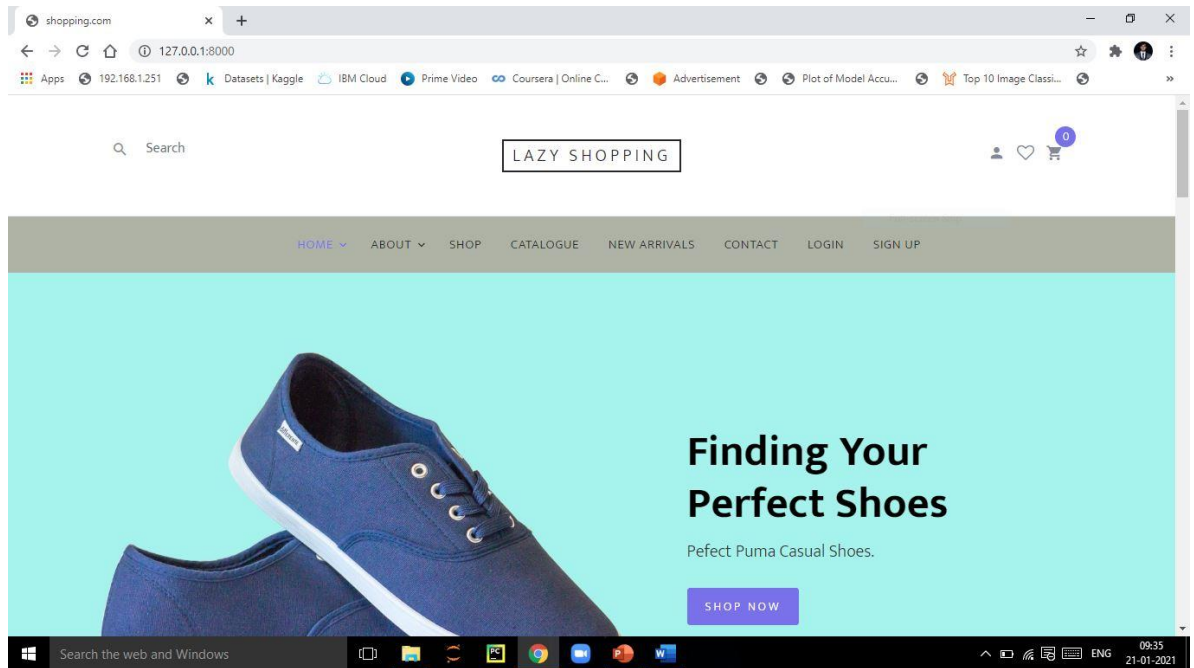
1. Make our website mobile friendly.
2. Provide customer reviews.
3. Give a live chat option.
4. Proofread our content.
5. Offer free shipping.
6. Make our site easy to navigate.
7. Use good photos

BIBLIOGRAPHY

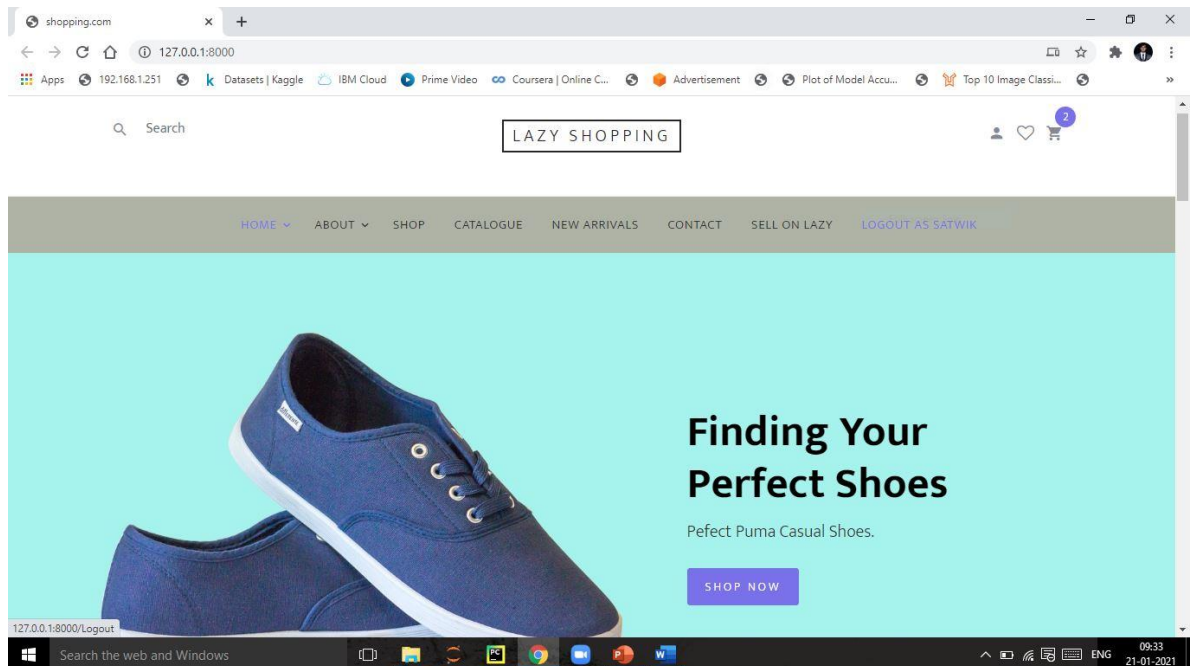
- [1] Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", 1st Edition, Pearson Education India. (ISBN:978-9332575271)
- [2] Robin Nixon, "Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5", 4th Edition, O'Reilly Publications, 2015. (ISBN:978-9352130153)
- [3] Nicholas C Zakas, "Professional JavaScript for Web Developers", 3rd Edition, Wrox/Wiley India, 2012. (ISBN:978-8126535088)
- [4] Zak Ruvalcaba Anne Boehm, "Murach's HTML5 and CSS3", 3rd Edition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016. (ISBN:978-9352133246)
- [5] <https://www.w3schools.com/w3css>
- [6] Django Documentation - <https://docs.djangoproject.com/en/3.1/>
- [7] Django Rest Framework Documentation - <https://www.django-rest-framework.org/>

APPENDIX A

SNAPSHOTS



Snapshots 1 Home Page before login



Snapshots 2 Home Page after login

register

127.0.0.1:8000/register

Apps 192.168.1.251 Datasets | Kaggle IBM Cloud Prime Video Coursera | Online C... Advertisement Plot of Model Accu... Top 10 Image Classi...

First name:
Enter First Name

Last name:
Enter Last Name

Email
Enter a valid Email, an otp will be sent

Password
Enter Password

Repeat Password
Repeat Password

Phone
Phone

SIGN UP

By creating an account you agree to our [Terms & Privacy](#).

Already have an account? [Sign in](#).

Search the web and Windows

09:37 21-01-2021

Snapshots 3 Sign Up Page

register

127.0.0.1:8000/otp

Apps 192.168.1.251 Datasets | Kaggle IBM Cloud Prime Video Coursera | Online C... Advertisement Plot of Model Accu... Top 10 Image Classi...

Email Verification

Please Enter OTP to verify your Email.

Please note that, if your email is not verified you cannot Buy/Sell any products.

Email:
Email

OTP:
Enter OTP

VERIFY EMAIL

By creating an account you agree to our [Terms & Privacy](#).

Already have an account? [Sign in](#).

Search the web and Windows

09:37 21-01-2021

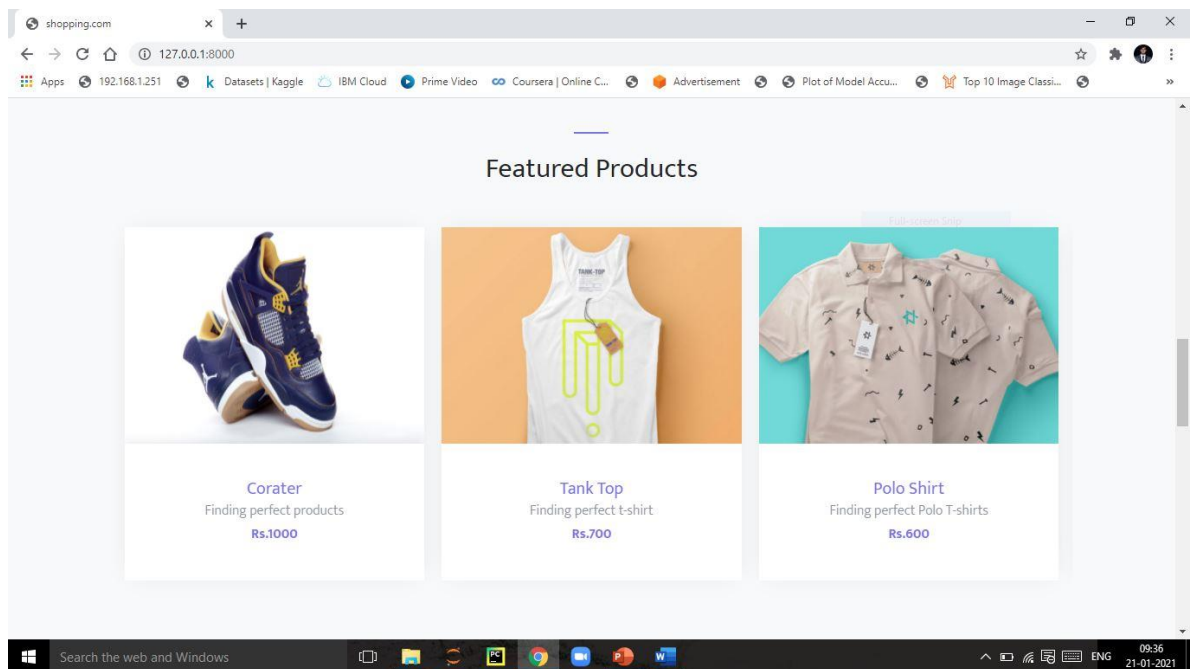
Snapshots 4 Email OTP Verification Page

The screenshot shows a web browser window with the URL `127.0.0.1:8000/login`. The page has a large heading "Welcome Back!" and a section titled "Login to your Account." Below this, there are input fields for "Phone" (containing "6363219400") and "Password" (masked with dots). There are "LOGIN" and "CANCEL" buttons. Below the password field, there is a link "Forgot password?" and a message "Please Don't save password in any google pages". At the bottom, there is a link "Don't Have Account? Sign up." The Windows taskbar at the bottom shows the time as 20:43 on 21-01-2021.

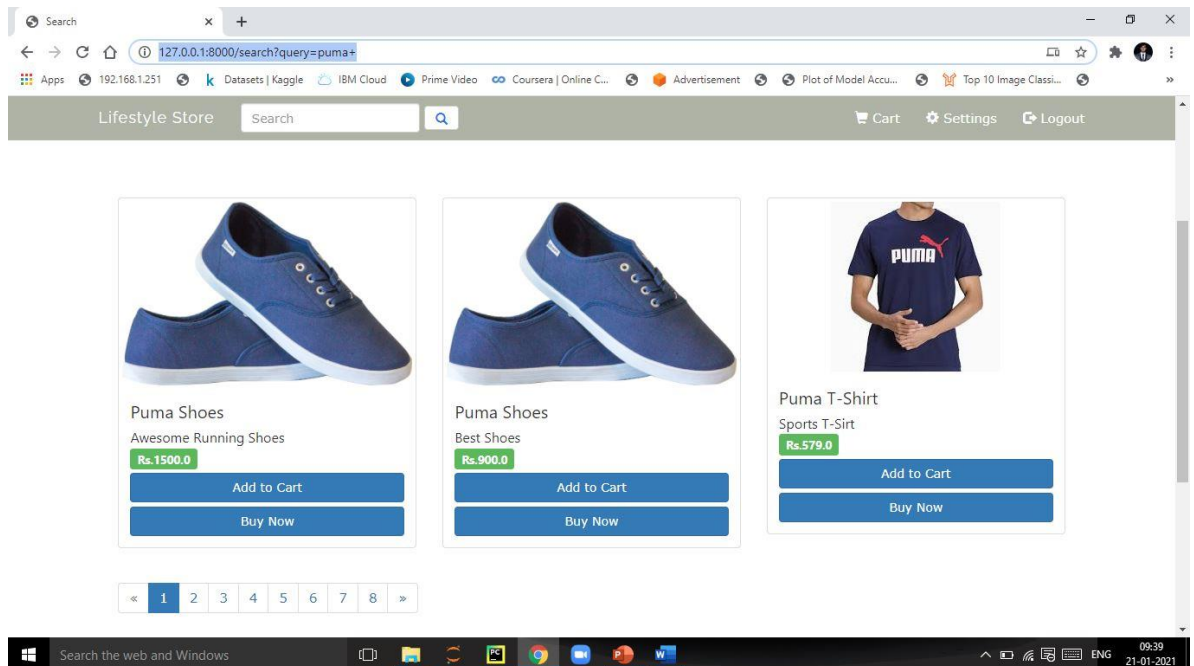
Snapshots 5 Login Page

The screenshot shows a web browser window with the URL `127.0.0.1:8000/sell`. The page is titled "Sell Your Products". It has a form with the following fields: "Product Name" (containing "Ferrari Watch"), "Price" (containing "10000"), and "Product description in 50 - 200 words" (containing a detailed description of a Ferrari watch). There is a "Choose a file to upload Photo of your product." section with a "Choose File" button and a file named "ferrari.jpg". Below this is a "Choose a Category" dropdown menu with "Men Fashion" selected. At the bottom, there is a "Sell" button. The Windows taskbar at the bottom shows the time as 09:44 on 21-01-2021.

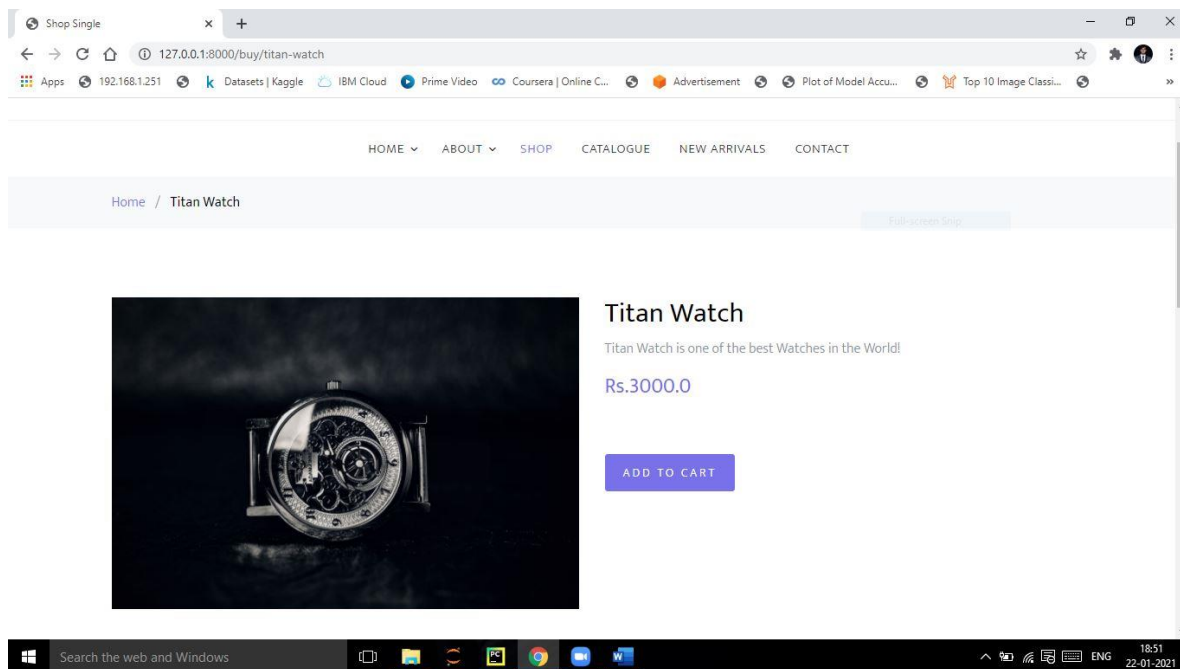
Snapshots 6 Sell Products Page



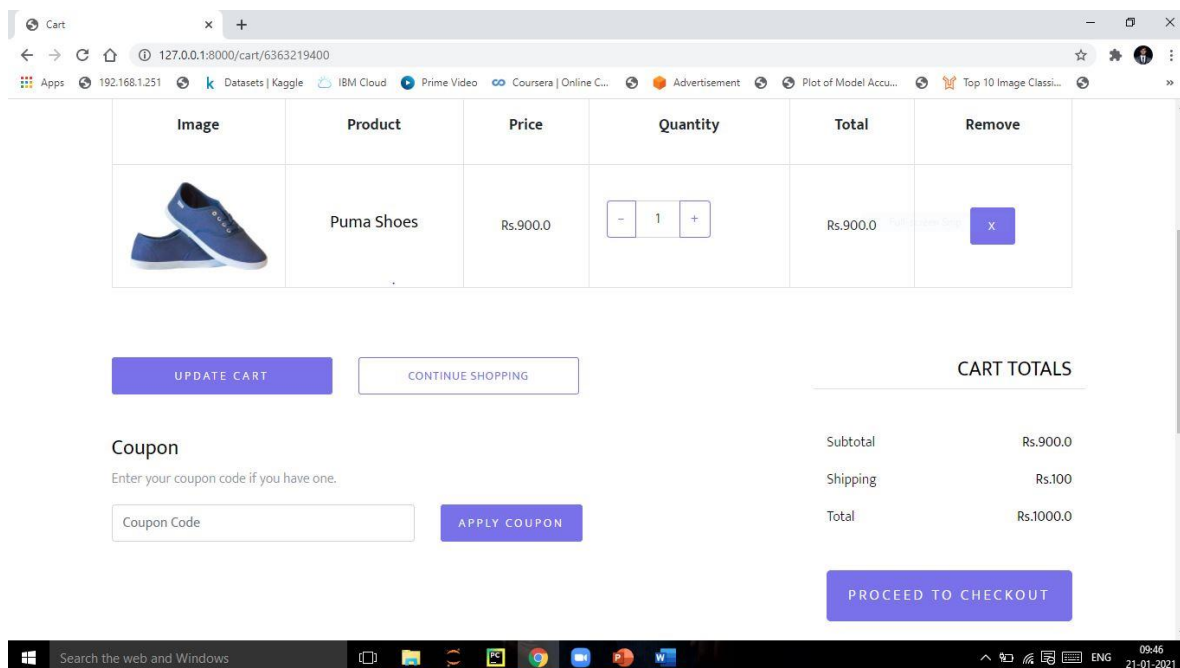
Snapshots 7 Features Page



Snapshots 8 Search Results Page



Snapshots 9 Buy Now Page



Snapshots 10 Cart Page

Checkout

127.0.0.1:8000/checkout

Address*

Street address

Apartment, suite, unit etc. (optional)

State* Posta / Zip*

Email Address* Phone*

Phone Number

Order Notes

Write your notes here...

Your Order

Product	Total
Puma Shoes x 1	Rs.900.0
Cart Subtotal	Rs.900.0
Shipping	Rs.100
Order Total	1000.0

PLACE ORDER

Snapshots 11 Checkout Page

Thank You

127.0.0.1:8000/checkout

✓

Thank you!

You order was successfully completed.

BACK TO SHOP

Navigations

- Sell online
- Mobile commerce
- Point of sale
- Features
- Dropshipping
- Hardware
- Shopping cart
- Website development
- Software

Promo

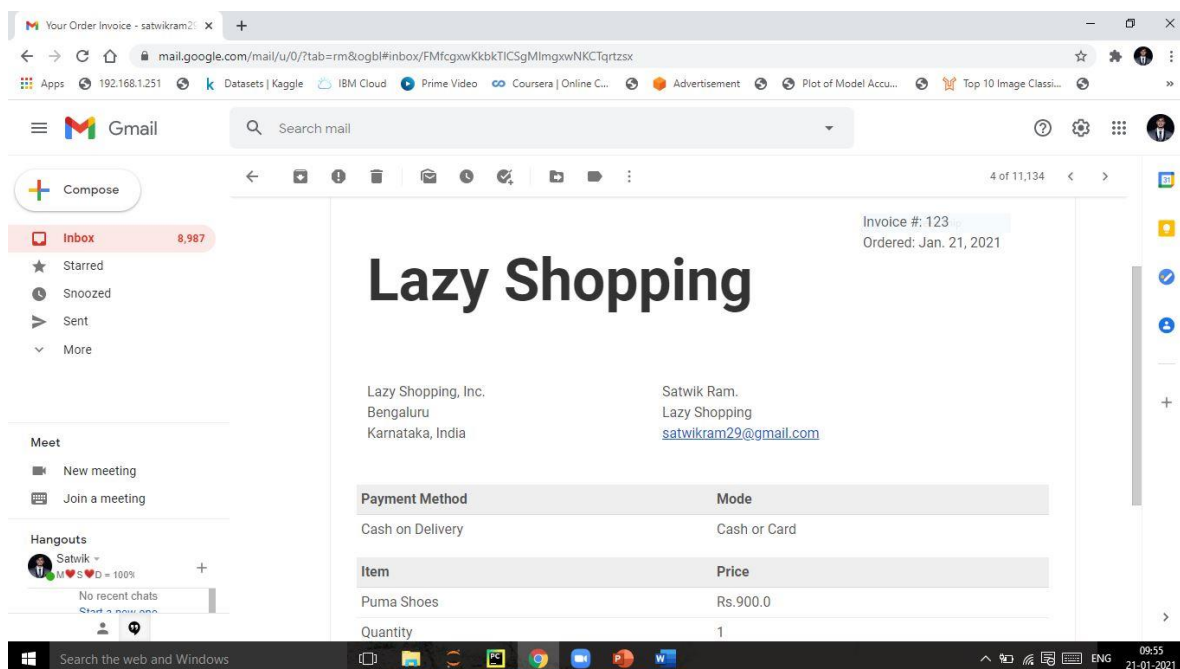
Contact Info

Bengaluru, Karnataka, India





+91 6363219400

satwikram29@gmail.com

Snapshots 12 Thank You Page after buying Products



Snapshots 13 Invoice to Email

Image	Product	Price	Quantity	Total	Shipping Charge	Delivery
	Armani Watch	Rs.1000.0	1	Rs.1000.0	Rs.0.0	No
	Rolex Watch	Rs.2500.0	1	Rs.2500.0	Rs.0.0	No
	Armani Watch	Rs.1000.0	1	Rs.1000.0	Rs.0.0	No
						

Snapshots 14 Orders Page