# Gokhale college of Engineering and Management Nashik -5

**A**

**Project Report**

**On**

# "PROJECT TITLE"

**For The Course**

**Database Management System**

**SUBMITTED BY**

**Group member1 (Roll No: )**

**Group member2 (Roll No: )**

**Group member13 (Roll No: )**

**Third Year Computer Engineering.**

**GUIDED BY**

**Name of Mentor**

**SUBMITTED TO**

**Gokhale college of Engineering and Management Nashik -5**

**For Academic Year**

**2017-18**

# CONTENTS

# *INTRODUCTION*

INTRODUCTION : An eCommerce website is an information technology method in which trader, businesses/distributor/marketers can sell products/services and the customer can purchase on that website electronically by using internet on the mobile and computer. It means an e-commerce website is an online shop. e means electronic. Commerce mean business. Website means the group of HTML web pages and that is created to market/sell information/product/services.

In a bigger perspective, every website on the internet is the eCommerce Website. It can be the platform, it can be a marketplace, it can be portal, it can be apps, it can be an entertainment website, shopping website, online courses website and online degree college.

**Ecommerce Website and business:**

- When you purchase a mobile phone /shoes/software/ flowers on any website such as Amazon, Flipkart etc. and pay through credit/debit card and then seller deliver the product through courier or post mail on your location then it's called e-commerce. In this case, Flipkart is an online store website or an e-commerce website.
- When you subscribe to watch a cricket match, movies, and shows on any website such as hotstar through debit card and credit card it is called e-commerce. And in this way, hotstar is a digital and mobile entertainment e-commerce website.
- When you rent or buy movies on YouTube and pay to watch by using the mobile/computer and internet it's called e-commerce. In this method, you have used computer/mobile and internet through electricity and visited youtube website to watch/buy/rent the movie and paid through debit card/credit card/ net banking/payment wallet etc. It means youtube is an eCommerce website in which you can buy/watch/rent the latest movies and shows When you purchase software as a service,

platform as a service, infrastructure as service for your business from cloud computing service providers such as Alibaba, Amazon web service, Microsoft, google cloud etc. on their website then these are the e-commerce websites.

- When you use internet banking then it is e-commerce. You pay bills, transfer money, open RD/FD account, pay installments online, pay for offline products from payment wallets etc. are e-commerce.

**How My Awesome Mart Websites Works:**

Almost all E-Commerce website works similarly. Following is the process of e-commerce website/eCommerce business/online transactions.

- **Internet –** Connecting the people through computers/mobiles and internet.
- **User –** Searching on Google and other search engines for products/services and daily life solutions. **Website –** After

search or research user visits the website that is in top 10 search results.

- **Products / Services –** User (customer) find the product and select it and add it to cart.
- **Purchased –** Now user purchases it through debit and credit card by using third-party payment gateway such as ccavenue, payubiz etc.
- **Payment Gateway – (Payment Gateway, Merchant Accounts, and Online Credit Card Processing Service provider) –** Received the payment and transfer it into website owner or sellers accounts after 1 week or later.
- **Bank Account –** Customer can pay using the debit/credit card and net banking/merchant receive money in the bank account from payment gateway service provider.
- **Delivery –** on spot, While many deliver the product within 8 days after receiving the payment. Some only receive cash on the delivery. above are general explanations of the e-commerce websiteand online store. If you want technical part then please comment. I will cover it in the next article

# *DEVELOPMENT TOOLS:-*

1) **Language: CSS, CSS3, JavaScript, HTML, Django, SQL,.**
2) **We used Pycharm Text as our editor.**
3) **We use Apache server.**
4) **Web-Browser: Mozilla Firefox/Google Chrome/Opera Mini/ Safari /IE11/Microsoft Edge.**
5) **Frameworks and Libraries: Bootstrap and jQuery.**

## ***SYSTEM REQUIREMENTS:***

**On client side**

- Operating System (any)
- Web Browser (IE8, Mozilla Firefox, Google Chrome, Opera, Safari)

**On server side**

- Localhost server
- Django
- Sqllite

## *Hardware Requirements:*

### At Server Side

- P IV or above Processor
- 1 GB RAM
- 120 GB HDD
- *LAN or WAN*

## *FEATURES:-*

- Content management capabilities
- Promotion and discount code tools
- An easy-to-use checkout
- Search engine optimized code and layout
- Reporting tools
- An integrated blog or articles section
- Email marketing integration
- Multiple payment options (Credit card, PayPal, PO, Terms, etc.)
- The ability

## *SNAPSHOTS:-*

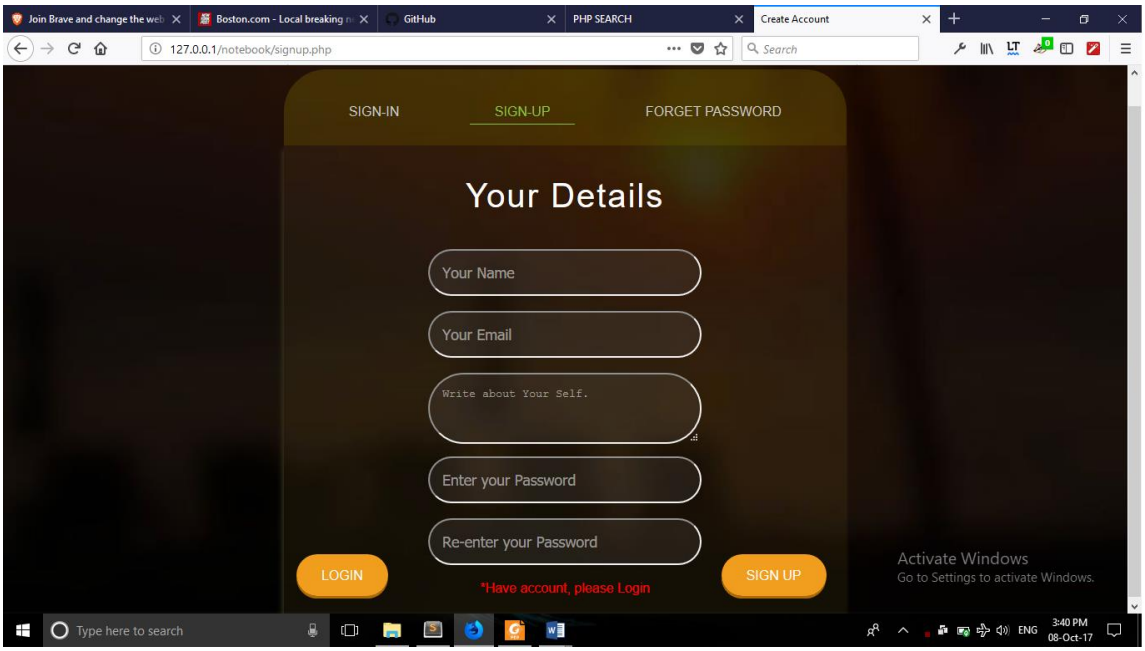
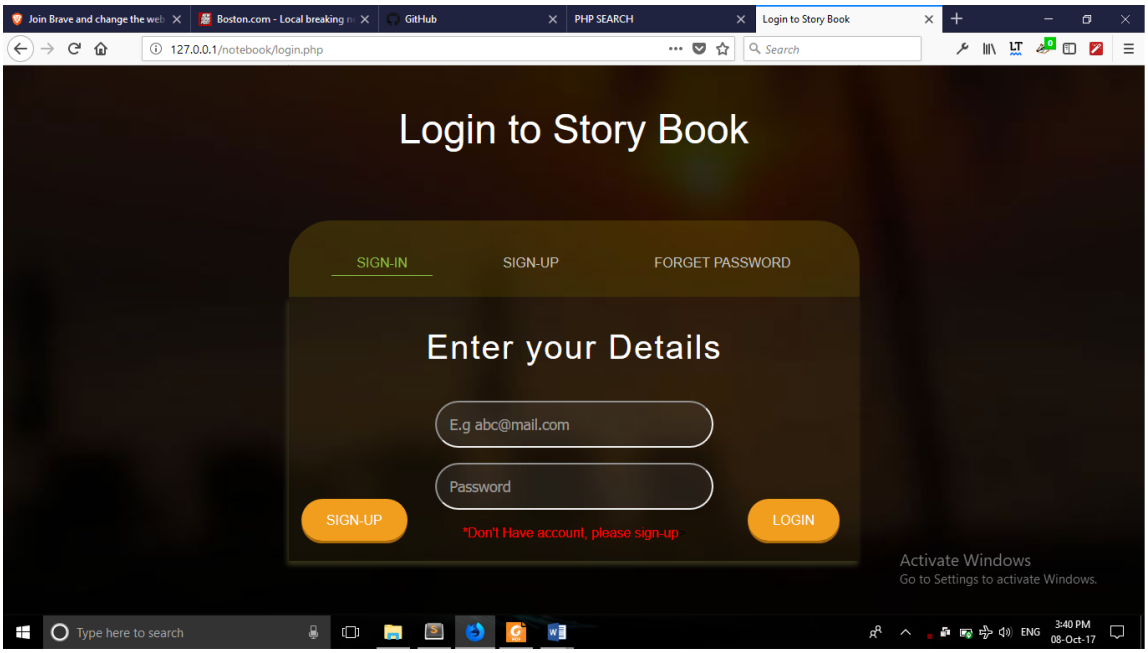



***Fig. 2***



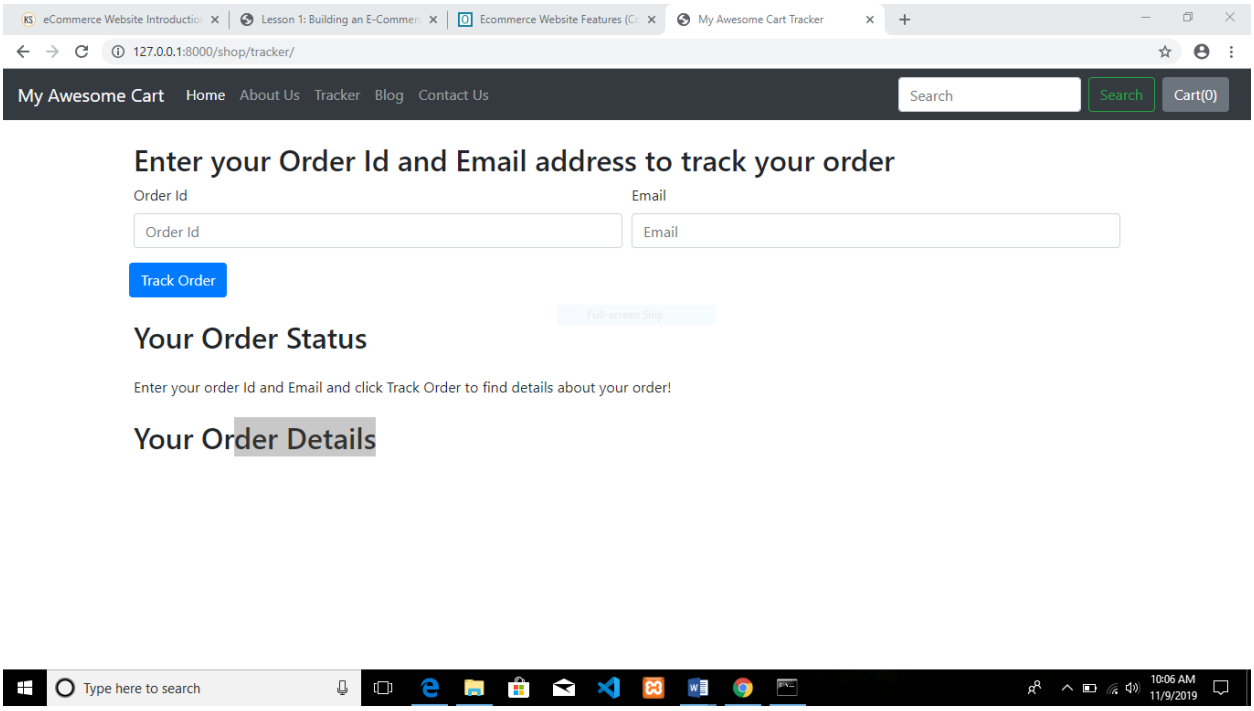



Fig. –Tracker System

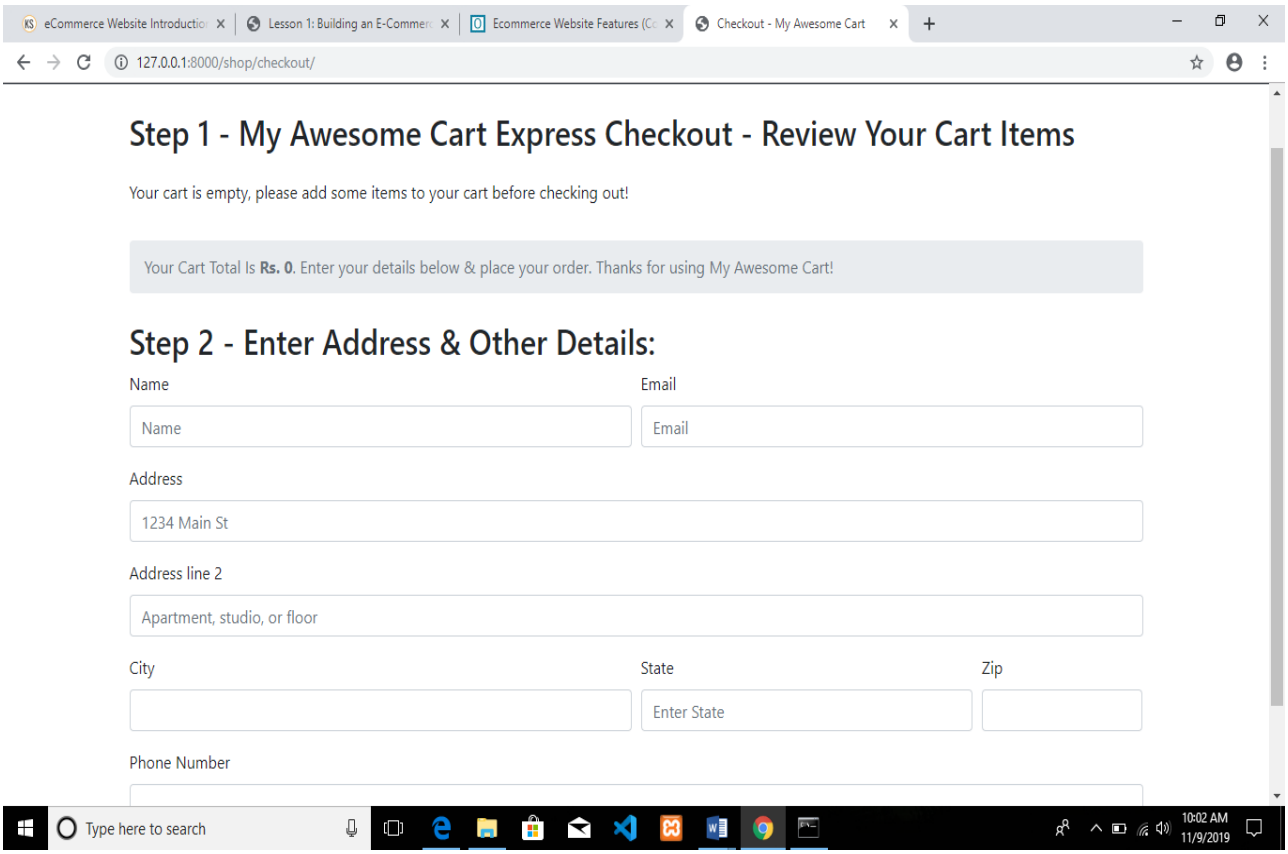
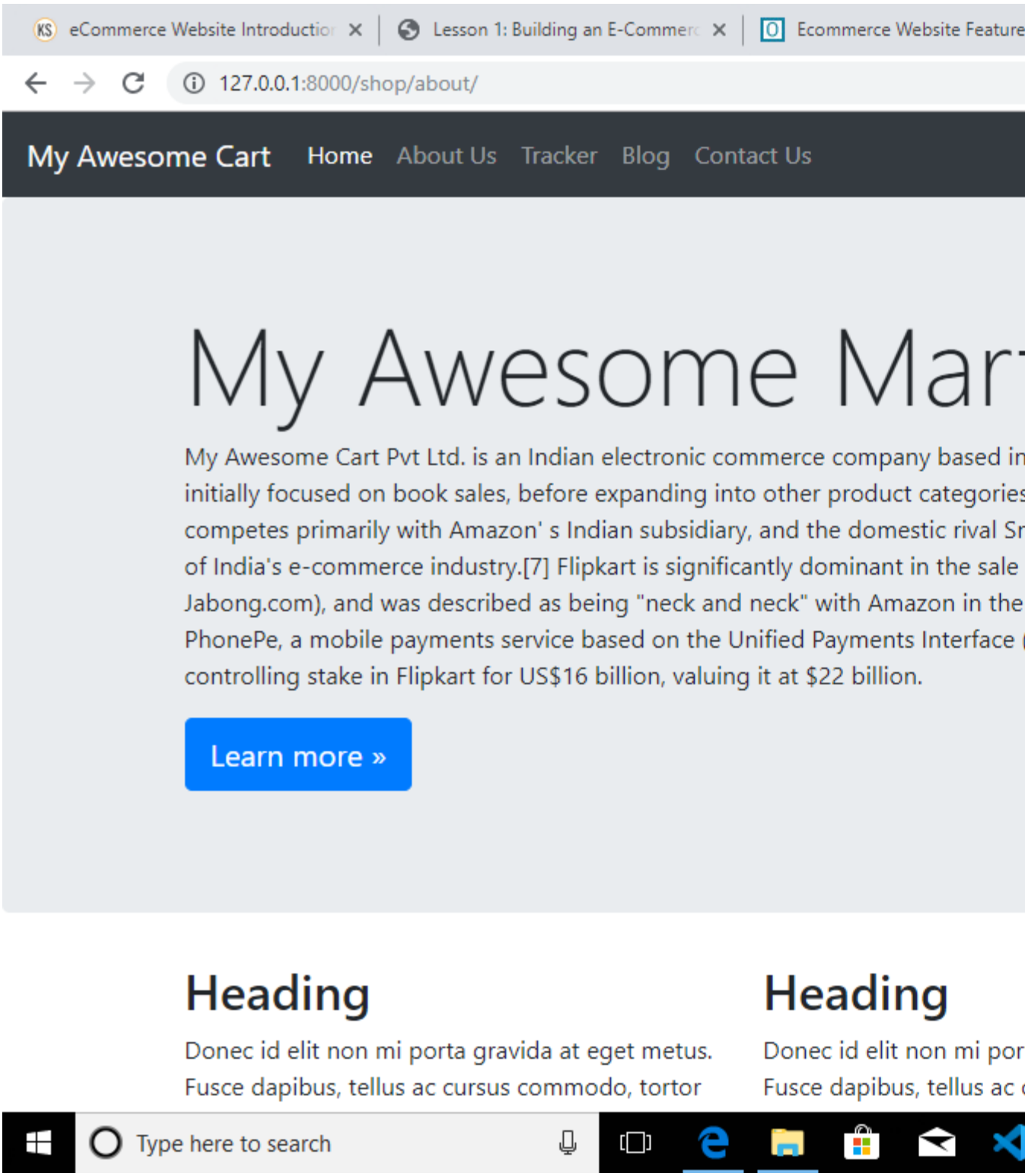
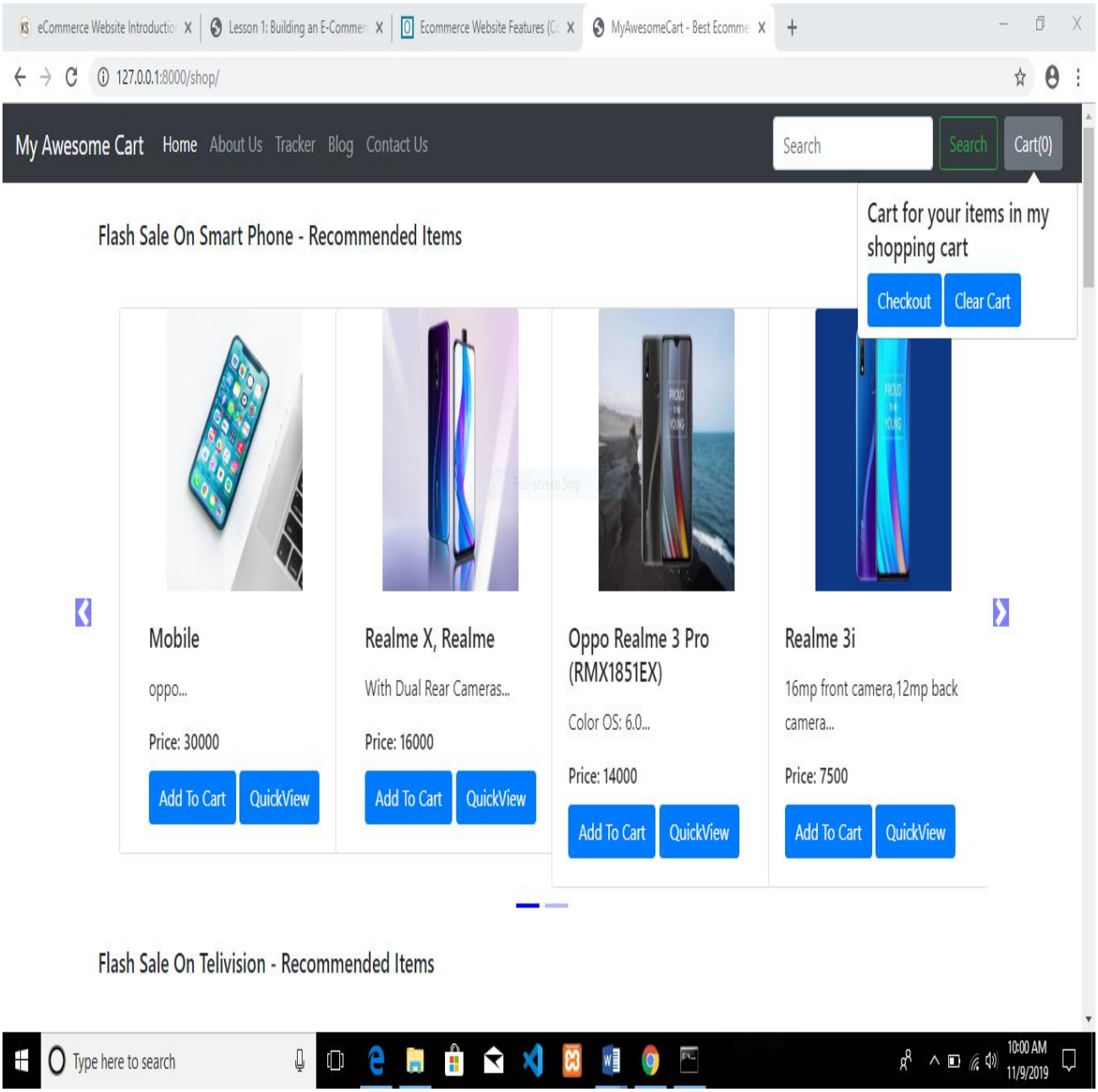



Fig.- Checkout System


eCommerce Website Introductio
Lesson 1: Building an E-Commer
Ecommerce Website Feature
127.0.0.1:8000/shop/about/
My Awesome Cart
Home
About Us
Tracker
Blog
Contact Us
My Awesome Mar
My Awesome Cart Pvt Ltd. is an Indian electronic commerce company based in
initially focused on book sales, before expanding into other product categories
competes primarily with Amazon' s Indian subsidiary, and the domestic rival Sn
of India's e-commerce industry.[7] Flipkart is significantly dominant in the sale
Jabong.com), and was described as being "neck and neck" with Amazon in the
PhonePe, a mobile payments service based on the Unified Payments Interface
controlling stake in Flipkart for US$16 billion, valuing it at $22 billion.
Learn more »
Heading
Donec id elit non mi porta gravida at eget metus.
Fusce dapibus, tellus ac cursus commodo, tortor
Heading
Donec id elit non mi por
Fusce dapibus, tellus ac
Type here to search


eCommerce Website Introductio
Lesson 1: Building an E-Commer
Ecommerce Website Features (Co
MyAwesomeCart - Best Ecomme
127.0.0.1:8000/shop/
My Awesome Cart
Home
About Us
Tracker
Blog
Contact Us
Search
Search
Cart(0)
Cart for your items in my shopping cart
Checkout
Clear Cart
Flash Sale On Smart Phone - Recommended Items
Mobile
oppo...
Price: 30000
Add To Cart
QuickView
Realme X, Realme
With Dual Rear Cameras...
Price: 16000
Add To Cart
QuickView
Oppo Realme 3 Pro (RMX1851EX)
Color OS: 6.0...
Price: 14000
Add To Cart
QuickView
Realme 3i
16mp front camera,12mp back camera...
Price: 7500
Add To Cart
QuickView
Flash Sale On Telivision - Recommended Items
Type here to search
10:00 AM
11/9/2019

## SOURCE CODE:-

## Insert in to database:--

```
from django.shortcuts import render
from .models import Product, Contact, Orders, OrderUpdate
from math import ceil
import json
from django.views.decorators.csrf import csrf_exempt
from PayTm import Checksum
# Create your views here.
from django.http import HttpResponse
MERCHANT_KEY = 'Your-Merchant-Key-Here'

def index(request):
    allProds = []
    catprods = Product.objects.values('category', 'id')
    cats = {item['category'] for item in catprods}
    for cat in cats:
        prod = Product.objects.filter(category=cat)
        n = len(prod)
        nSlides = n // 4 + ceil((n / 4) - (n // 4))
        allProds.append([prod, range(1, nSlides), nSlides])
    params = {'allProds':allProds}
    return render(request, 'shop/index.html', params)

def searchMatch(query, item):
    '''return true only if query matches the item'''
    if query in item.desc.lower() or query in item.product_name.lower() or query in
item.category.lower():
        return True
    else:
        return False

def search(request):
    query = request.GET.get('search')
    allProds = []
    catprods = Product.objects.values('category', 'id')
    cats = {item['category'] for item in catprods}
    for cat in cats:
        prodtemp = Product.objects.filter(category=cat)
        prod = [item for item in prodtemp if searchMatch(query, item)]

        n = len(prod)
        nSlides = n // 4 + ceil((n / 4) - (n // 4))
        if len(prod) != 0:
            allProds.append([prod, range(1, nSlides), nSlides])
    params = {'allProds': allProds, "msg": ""}
    if len(allProds) == 0 or len(query)<4:
        params = {'msg': "Please make sure to enter relevant search query"}
    return render(request, 'shop/search.html', params)


def about(request):
```

```
    return render(request, 'shop/about.html')


def contact(request):
    thank = False
    if request.method=="POST":
        name = request.POST.get('name', '')
        email = request.POST.get('email', '')
        phone = request.POST.get('phone', '')
        desc = request.POST.get('desc', '')
        contact = Contact(name=name, email=email, phone=phone, desc=desc)
        contact.save()
        thank = True
    return render(request, 'shop/contact.html', {'thank': thank})


def tracker(request):
    if request.method=="POST":
        orderId = request.POST.get('orderId', '')
        email = request.POST.get('email', '')
        try:
            order = Orders.objects.filter(order_id=orderId, email=email)
            if len(order)>0:
                update = OrderUpdate.objects.filter(order_id=orderId)
                updates = []
                for item in update:
                    updates.append({'text': item.update_desc, 'time': item.timestamp})
                    response = json.dumps([updates, order[0].items_json], default=str)
                return HttpResponse(response)
            else:
                return HttpResponse('{}')
        except Exception as e:
            return HttpResponse('{}')

    return render(request, 'shop/tracker.html')


def productView(request, myid):

    # Fetch the product using the id
    product = Product.objects.filter(id=myid)
    return render(request, 'shop/prodView.html', {'product':product[0]})


def checkout(request):
    if request.method=="POST":
        items_json = request.POST.get('itemsJson', '')
        name = request.POST.get('name', '')
        amount = request.POST.get('amount', '')
        email = request.POST.get('email', '')
        address = request.POST.get('address1', '') + " " +
request.POST.get('address2', '')
        city = request.POST.get('city', '')
        state = request.POST.get('state', '')
        zip_code = request.POST.get('zip_code', '')
        phone = request.POST.get('phone', '')
        order = Orders(items_json=items_json, name=name, email=email, address=address,
city=city,
                       state=state, zip_code=zip_code, phone=phone, amount=amount)
        order.save()
        update = OrderUpdate(order_id=order.order_id, update_desc="The order has been
placed")
        update.save()
```

```
        thank = True
        id = order.order_id
        # return render(request, 'shop/checkout.html', {'thank':thank, 'id': id})
        # Request paytm to transfer the amount to your account after payment by user
        param_dict = {

                'MID': 'Your-Merchant-Id-Here',
                'ORDER_ID': str(order.order_id),
                'TXN_AMOUNT': str(amount),
                'CUST_ID': email,
                'INDUSTRY_TYPE_ID': 'Retail',
                'WEBSITE': 'WEBSTAGING',
                'CHANNEL_ID': 'WEB',
                'CALLBACK_URL':'http://127.0.0.1:8000/shop/handlerequest/',

        }
        param_dict['CHECKSUMHASH'] = Checksum.generate_checksum(param_dict,
MERCHANT_KEY)
        return render(request, 'shop/paytm.html', {'param_dict': param_dict})

    return render(request, 'shop/checkout.html')


@csrf_exempt
def handlerequest(request):
    # paytm will send you post request here
    form = request.POST
    response_dict = {}
    for i in form.keys():
        response_dict[i] = form[i]
        if i == 'CHECKSUMHASH':
            checksum = form[i]

    verify = Checksum.verify_checksum(response_dict, MERCHANT_KEY, checksum)
    if verify:
        if response_dict['RESPCODE'] == '01':
            print('order successful')
        else:
            print('order was not successful because' + response_dict['RESPMSG'])
    return render(request, 'shop/paymentstatus.html', {'response': response_dict})
```

## *Index Page Code:--*

```
{% extends 'shop/basic.html' %}
{% block title%} MyAwesomeCart - Best Ecommerce Website{% endblock %}
{% block css %}
.col-md-3
{
display: inline-block;
margin-left:-4px;
}
.carousel-indicators .active {
background-color: blue;
}
```

```
.col-md-3 img{
width: 170px;
height: 200px;
}
body .carousel-indicator li{
background-color: blue;
}
body .carousel-indicators{
bottom: -40px;
}
.carousel-indicators li {


    background-color: #7270fc;
}
body .carousel-control-prev-icon,
body .carousel-control-next-icon{
background-color: blue;
}
.carousel-control-prev,
.carousel-control-next{
top: auto;
bottom: auto;
padding-top: 222px;
}
body .no-padding{
padding-left: 0,
padding-right: 0;
}
{% endblock %}
{% block body %}
{% load static %}
<div class="container">
    <!--Slideshow starts here -->
    {% for product, range, nSlides in allProds %}
    <h5 class="my-4">Flash Sale On {{product.0.category}} - Recommended Items</h5>
    <div class="row">
        <div id="demo{{forloop.counter}}" class="col carousel slide my-3" data-
ride="carousel">
            <ul class="carousel-indicators">
                <li data-target="#demo{{forloop.counter}}" data-slide-to="0"
class="active"></li>
                {% for i in range %}
                <li data-target="#demo{{forloop.parentloop.counter}}" data-slide-
to="{{i}}"></li>
                {% endfor %}
            </ul>
            <div class="container carousel-inner no-padding">
                <div class="carousel-item active">
                    {% for i in product %}
                    <div class="col-xs-3 col-sm-3 col-md-3">
                        <div class="card align-items-center" style="width: 18rem;">
                            <img src='/media/{{i.image}}' class="card-img-top"
alt="...">
                            <div class="card-body">
                                <h5 class="card-title"
id="namepr{{i.id}}">{{i.product_name}}</h5>
                                <p class="card-text">{{i.desc|slice:"0:53"}}...</p>
                                <h6 class="card-title" >Price: <span
id="pricepr{{i.id}}">{{i.price}}</span></h6>
                                <span id="divpr{{i.id}}" class="divpr">
                                    <button id="pr{{i.id}}" class="btn btn-primary
cart">Add To Cart</button>
```

```
                                </span>
                                <a href="/shop/products/{{i.id}}"><button
id="qv{{i.id}}" class="btn btn-primary cart">QuickView</button></a>
                            </div>
                        </div>
                    </div>
                    {% if forloop.counter|divisibleby:4 and forloop.counter > 0 and
not forloop.last %}
                </div>
                <div class="carousel-item">
                    {% endif %}
                    {% endfor %}
                </div>
            </div>
        </div>
        <!-- left and right controls for the slide -->
        <a class="carousel-control-prev" href="#demo{{forloop.counter}}" data-
slide="prev">
            <span class="carousel-control-prev-icon"></span>
        </a>
        <a class="carousel-control-next" href="#demo{{forloop.counter}}" data-
slide="next">
            <span class="carousel-control-next-icon"></span>
        </a>
    </div>
    {% endfor %}
</div>
{% endblock %}
{% block js %}
<script>
// Find out the cart items from localStorage
if (localStorage.getItem('cart') == null) {
    var cart = {};
} else {
    cart = JSON.parse(localStorage.getItem('cart'));
    updateCart(cart);
}
// If the add to cart button is clicked, add/increment the item
//$('.cart').click(function() {
    $('.divpr').on('click', 'button.cart', function(){
    var idstr = this.id.toString();
    if (cart[idstr] != undefined) {
        qty = cart[idstr][0] + 1;
    } else {
        qty = 1;
        name = document.getElementById('name'+idstr).innerHTML;
        price = document.getElementById('price'+idstr).innerHTML;
        cart[idstr] = [qty, name, parseInt(price)];
    }
    updateCart(cart);
});
//Add Popover to cart
$('#popcart').popover();
updatePopover(cart);
function updatePopover(cart) {
    console.log('We are inside updatePopover');
    var popStr = "";
    popStr = popStr + "<h5> Cart for your items in my shopping cart </h5><div
class='mx-2 my-2'>";
    var i = 1;
    for (var item in cart) {
        popStr = popStr + "<b>" + i + "</b>. ";
        popStr = popStr + document.getElementById('name' + item).innerHTML.slice(0,
```

```
19) + "... Qty: " + cart[item][0] + '<br>';
        i = i + 1;
    }
    popStr = popStr + "</div> <a href='/shop/checkout'><button class='btn btn-primary'
id ='checkout'>Checkout</button></a> <button class='btn btn-primary'
onclick='clearCart()' id ='clearCart'>Clear Cart</button>     "
    console.log(popStr);
    document.getElementById('popcart').setAttribute('data-content', popStr);
    $('#popcart').popover('show');
}
function clearCart() {
    cart = JSON.parse(localStorage.getItem('cart'));
    for (var item in cart) {
        document.getElementById('div' + item).innerHTML = '<button id="' + item + '"
class="btn btn-primary cart">Add To Cart</button>'
    }
    localStorage.clear();
    cart = {};
    updateCart(cart);
}
function updateCart(cart) {
    var sum = 0;
    for (var item in cart) {
        sum = sum + cart[item][0];
        document.getElementById('div' + item).innerHTML = "<button id='minus" + item +
"' class='btn btn-primary minus'>-</button> <span id='val" + item + "'>" +
cart[item][0] + "</span> <button id='plus" + item + "' class='btn btn-primary plus'> +
</button>";
    }
    localStorage.setItem('cart', JSON.stringify(cart));
    document.getElementById('cart').innerHTML = sum;
    console.log(cart);
    updatePopover(cart);
}
// If plus or minus button is clicked, change the cart as well as the display value
$('.divpr').on("click", "button.minus", function() {
    a = this.id.slice(7, );
    cart['pr' + a][0] = cart['pr' + a][0] - 1;
    cart['pr' + a][0] = Math.max(0, cart['pr' + a][0]);
    if (cart['pr' + a][0] == 0){
        document.getElementById('divpr' + a).innerHTML = '<button id="pr'+a+'"
class="btn btn-primary cart">Add to Cart</button>';
        delete cart['pr'+a];
    }
    else{
        document.getElementById('valpr' + a).innerHTML = cart['pr' + a][0];
    }
    updateCart(cart);
});
$('.divpr').on("click", "button.plus", function() {
    a = this.id.slice(6, );
    cart['pr' + a][0] = cart['pr' + a][0] + 1;
    document.getElementById('valpr' + a).innerHTML = cart['pr' + a][0];
    updateCart(cart);
});
</script>
{% endblock %}
```

**Economic benefits:**

- Banks earning the money due to the increased use of debit/credit cards.
- The government can use that money for the development of own machinery or citizens development.

**Business benefits:**

- Less costly to sell products and services.
- Wide variety of customers on the internet than offline store/shop.
- No credits  or Udhar
- Easy to manage transactions
- Easy to market/sell

**Consumer Benefits:**

- Getting product/service at door.
- Saves time
- Less costly than offline

# Future scope

**Business Scope:** Unlimited. More than half of the population on the earth is on the Internet intentionally and unintentionally. People are looking for the best quality and organic products, they want to feel vow at less cost without wasting time on offline shopping. So if someone think that they can sell online or have any thing that people needs then they can start.

**Job scope:** E Commerce created and increasing the Information Technology related jobs scopes worldwide. More and more people trying to build and run an online store and due to that there is a demand for the following experts:

- *Marketing Automation*

Online shoppers are notorious for browsing around for a long time and leaving the store without buying. Personalized and targeted marketing can help turn them into paying customers.

Marketing automation has been around for a long time, and the trend of using such solutions will continue to rise. More and more steps of lead nurturing, remarketing, email personalization and analytics will be automated, helping even small online store owners get better insights and convert more customers.

Hope this article will clear your all questions regarding the future **scope of ecommerce business in India.**

# *Cinclusion*

I was surprised to discover how many studies and analysis there are behind e-commerce. However, all the fundamental things required for conducting an excellent work do not differ greatly from the "real" ones.

The Internet represents a completely new and profitable way to do business, provided that all the requirements needed are met. For example, regarding a website, it is illegal to discriminate people with disabilities; a company may be sued for it.

The Internet is a parallel online world where all the resources are not fully exploited and where consumers may gain advantages. Here, companies may increase their profits further and users may find the desired product or service easily without the hassle of shopping around