

Introduction

Information systems are applications which are based on software and hardware to provide services. All the services provided are associated with the business entity, users and the business logic involved in it. Now-a-days with lot of changing trends in technology the necessity for information systems is increasing drastically. There are various types of information systems which includes decision making system, system to process the transactions, systems to manage the information or data the company holds and system that carries out all the executive operations.

Out of all the above-mentioned types of systems in this project an information system that processes transactions is studied and a novel system to maintain and process an online bakery shop is proposed and implemented. The rest of the report is arranged as follows, next section describes the research carried out prior to development of the system, all the plan that is made in developing the software is discussed. Later all the details regarding the technology stipe and frameworks that were used during the development are described. UX design section describes how the plan for a user interface is made and what functionalities that are present are also depicted. Webservices section describes the processing of the transaction when a call is made is to store a record or retrieve a record from the database are described. In security threats and measures section what type of measures we have taken for validating the data and also during transaction processing are briefly described. Features section depicts all the features that are included in each and every page of the system. And a brief out of all the topics in the previously discussed sections are included in conclusion, and that closes the report.

Research and Planning

Initially a thorough survey on what an information system does is studied. This survey stated that an information system is a kind of software associated with a machine that

has access to a data base or a repository and it functions according the instructions that are programmed in the software. This can be included in any type of enterprises, in tasks such as management of information, performing transactions and even they support a decision-making system. By this it is understood that an information system is a generic term that can be used for all these kinds of systems which perform previously stated action or similar ones.

Later it was decided that building a transaction processing system would be way beneficial to the enterprise as it can be used in any of type of business. Be it a purchasing website or simply an information providing site. All these systems should be able to perform transactions. These transactions can include a data base call or a server call. After keen observation on the current trends of business, it was found that each and every business be it they are in a food product, clothing or a shoe store need an online interface that enables users to purchase without visiting the store. So, a management system that handles orders of a bakery shop is planned to be designed. This helps both the users and the enterprise to get their work done easier.

Framework and Technologies

With the advent of python, it has become very easy for the developers to code a software. As it offers most flexible features during the development without having complex syntaxes to learn, it is as simple as plain English with programming task assigned to it. Django is used to design the front-end of the system. Coming to why Django it is the latest and easier way that a complex web application can be developed and deployed. Database used in this project is MySQL. To design the webpages Hyper Text Markup Language along with Cascading Style Sheets is used along with JavaScript to perform any validations that are required. Finally, to deploy the application on to the cloud amazon Ec2 instance is employed. To design the web-based online bakery shop following are technologies and frameworks are planned to be used.

- Programming Language: Python
- UX Design: Django
- Database: MySQL
- Design: HTML and CSS
- Validation: JavaScript
- Cloud deployment: AWS EC2, RDS
- API: Google Map API
- Server: Tomcat

Implementation

The project development is divided into different modules based on the requirements.

They are as follows:

1. Database Creation
2. Creation of User-Interface
3. Implementation of business logic

1. Database Creation

In this module all the required tables for login, products that are available, and so on are created and whichever tables require a pre-information to be used to display on the pages of the application are filled in. This includes whenever the customer selects few of the items and add to the card the table that is maintain this cart values should be updated based on the choice that the customer has made.

2. Creation of User-Interface

This is the main module that needs highest care to be taken as it is the interface that is visible to the user or the customer and through which they need to proceed further. It is always a norm that a user interface should be simple, well organized so that the user feels ease during the usage of the application. In this module a pre-planned and well

designed so that all the features implemented are easy to use and by the user. Django is the framework that is used throughout the project to implement such user interface.

3. Implementation of Business Logic

Business logic module of a project is the key as it contains all the coding part that implements the desired functionalities. In this project python programming language is chosen and all the functionalities described in the next section are implemented using the same.

Cloud Deployment

The application developed is deployed on to the amazon webservices cloud using an EC2 instance and is up and running. To do so an EC2 instance is launched and logged in using SSL and a node is installed and deployed the code on to the instance and set an option to start the server to run. When we want to deploy into the local machine tomcat server is used to run the application. The following are the steps involved in launching ec2 instance and configuring the RDS using AWS:

Instance Launch

Initially ec2 service is selected from the available AWS services. Figure 1 depicts the launch of the instance.

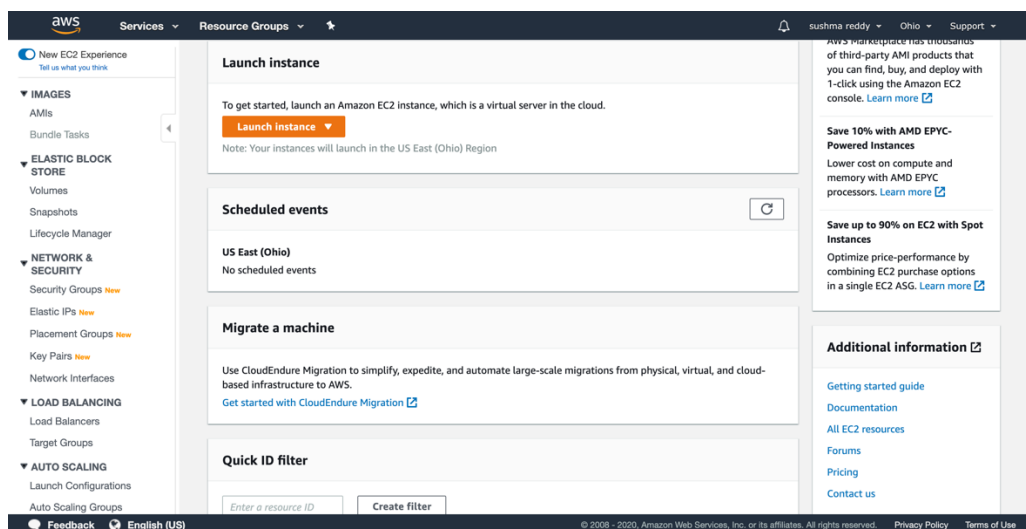


Figure 1: Instance Launch

Selecting the operating system

The next step is to select required operating system. Ubuntu machine is selected for this instance. Figure 2 depicts the selection of operating system.

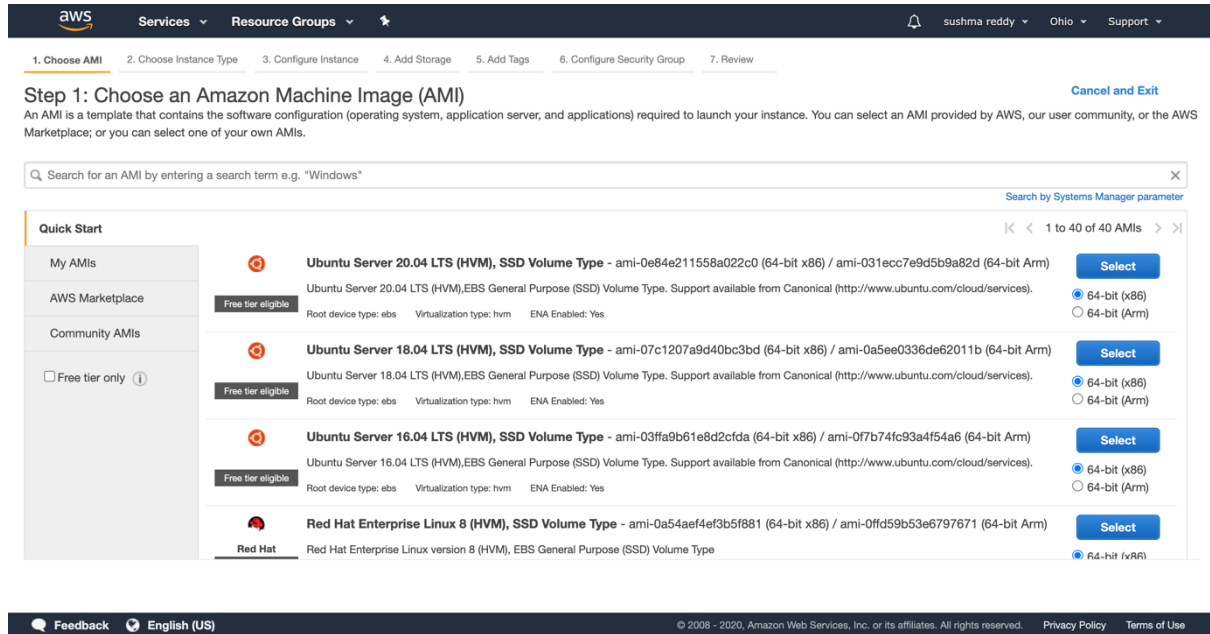


Figure 2: Selecting Operating System

Once the operating system is selected instance type should be selected. The type of instance varies based on the configuration like memory and the purpose of the machine.

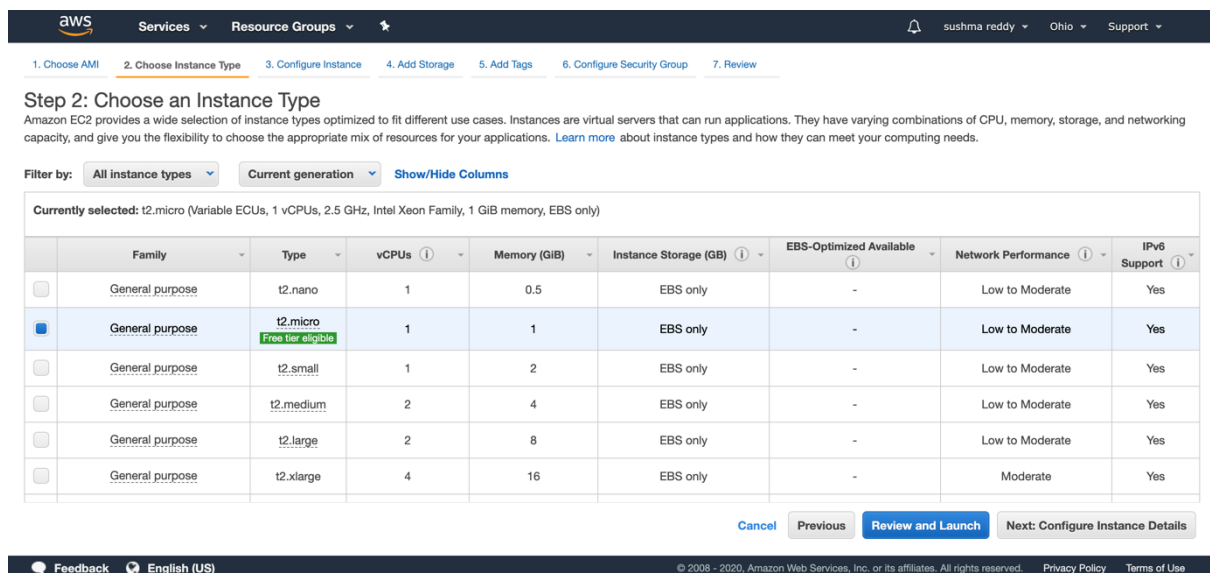


Figure 3: Select Type of Instance

Once the type of instance is selected, storage is added, and the security groups are configured. Figure 4 and 5 depicts the storage and the security rules.

Step 4: Add Storage
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-0d2a2e7855e80a405	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

Figure 4: Storage Configuration

Step 6: Configure Security Group
A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group
☐ Select an existing security group

Security group name:
 Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom 0.0.0.0/0, :::0	e.g. SSH for Admin Desktop

[Add Rule](#)

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

Figure 5: Security Group Configuration

Once the required configuration is selected, the instance is reviewed and launched. Figure 6 depicts the launch of the instance and figure 7 depicts the connection to the system using ssh.

aws

Services

Resource Groups

sushma reddy


Ohio

Support

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. Review

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.



Improve your instances' security. Your security group, launch-wizard-3, is open to the world.
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.
You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

AMI Details

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-07c1207a9d40bc3bd

Free tier eligible

Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
Root Device Type: ebs Virtualization type: hvm

Edit AMI

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Edit instance type

Security Groups

Security group name

launch-wizard-3

Description

launch-wizard-3 created 2020-06-16T17:35:02.197+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	:::0	

Edit security groups

Instance Details

Number of instances1

Purchasing optionOn demand

Networkvpc-85ef30ee

SubnetNo preference (default subnet in any Availability Zone)

EBS-optimizedNo

MonitoringNo

Termination protectionNo

Shutdown behaviorStop

Stop - Hibernate behaviorDisabled

Capacity Reservationopen

IAM roleNone

Tenancydefault

T2/T3 UnlimitedDisabled

Host ID

Host resource group name

AffinityOff

Kernel IDUse default

RAM disk IDUse default

Metadata accessibleEnabled

Metadata versionV1 and V2 (token optional)

Metadata token response hop limit1

User data

Assign Public IPUse subnet setting (Enable)

Assign IPv6 IPUse subnet setting (Enable)

Edit instance details

Storage

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0d2a2e7855e80a405	8	gp2	100 / 3000	N/A	Yes	Not Encrypted

Edit storage

Tags

Key	Value	Instances	Volumes
-----	-------	-----------	---------

This resource currently has no tags

Edit tags

Cancel

Previous

Launch

Feedback

English (US)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

Figure 6: Review and Launch the Instance

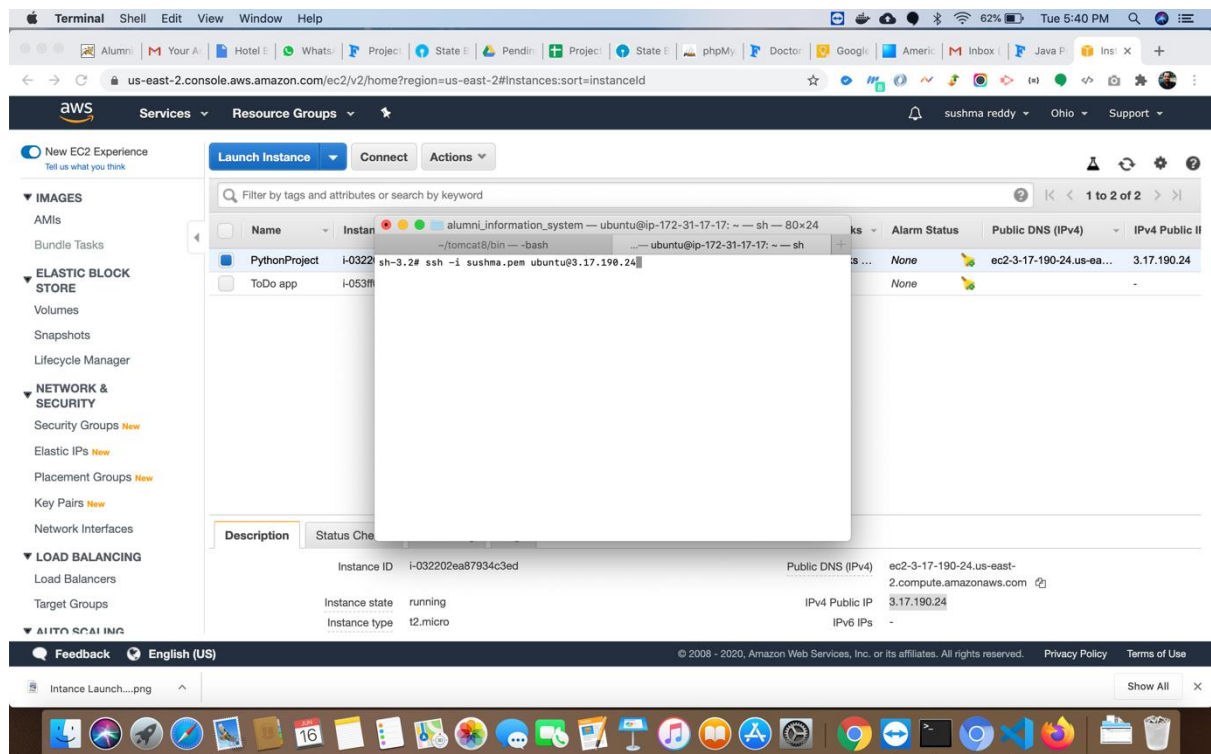


Figure 7(a): Connect to the Instance

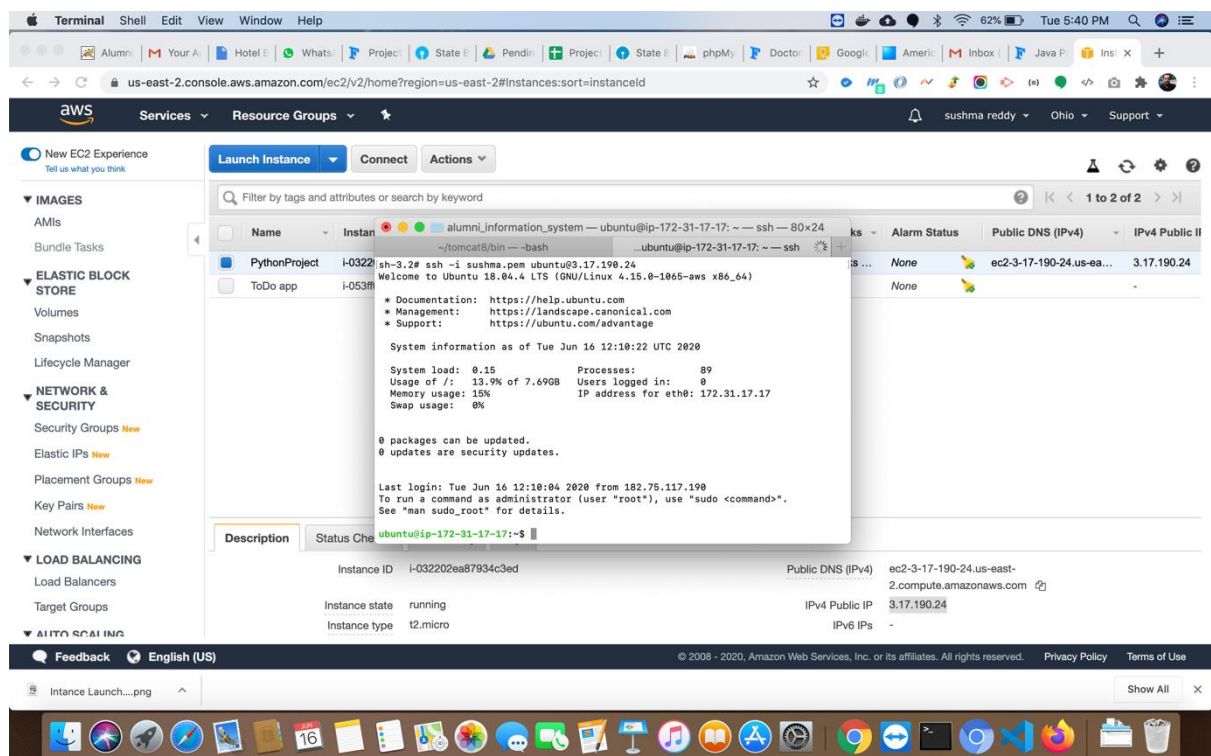


Figure 7(b): Connect to the Instance and login Successful

Once the instance is launched and logged in the application developed is moved into the machine. All the required tables and the data is inserted into the tables using SQL queries and the database is configured using RDS. Figure 8 depicts the configured database using the SQL queries configured.

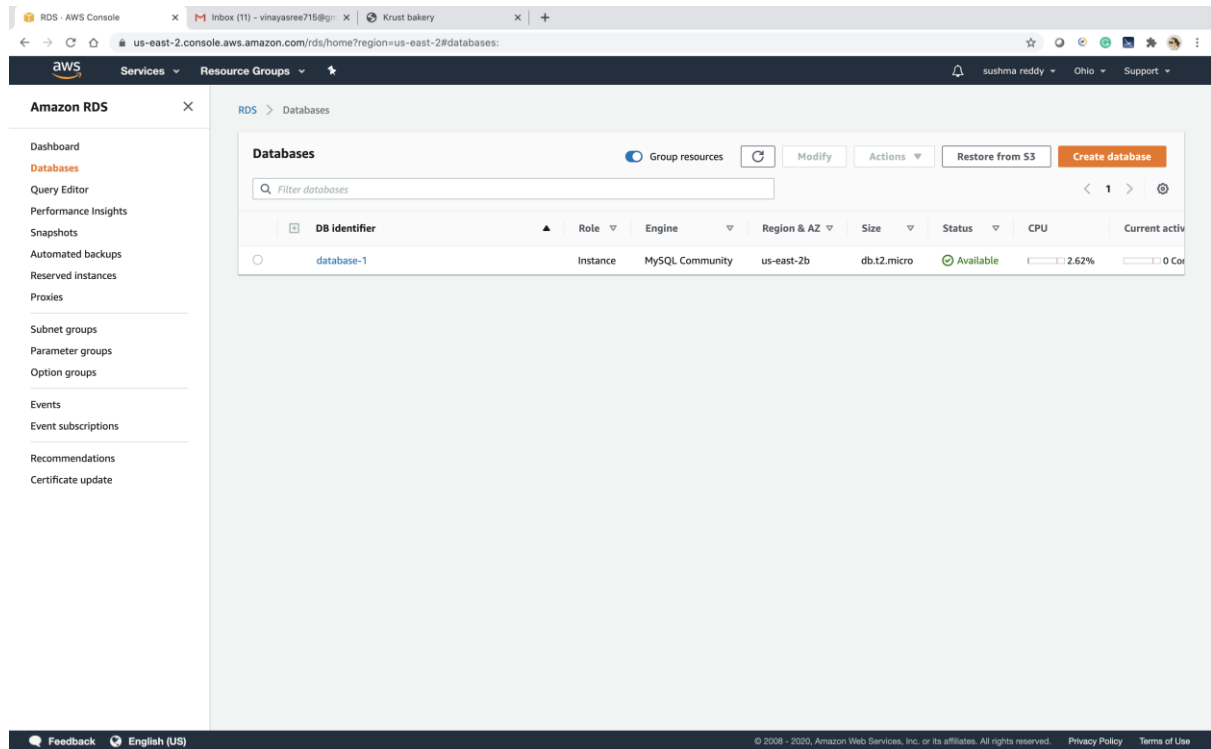


Figure 8: RDS Configuration

Features

The following are the features that are implemented so as to make the online bakery system to work as per the requirements of the customer.

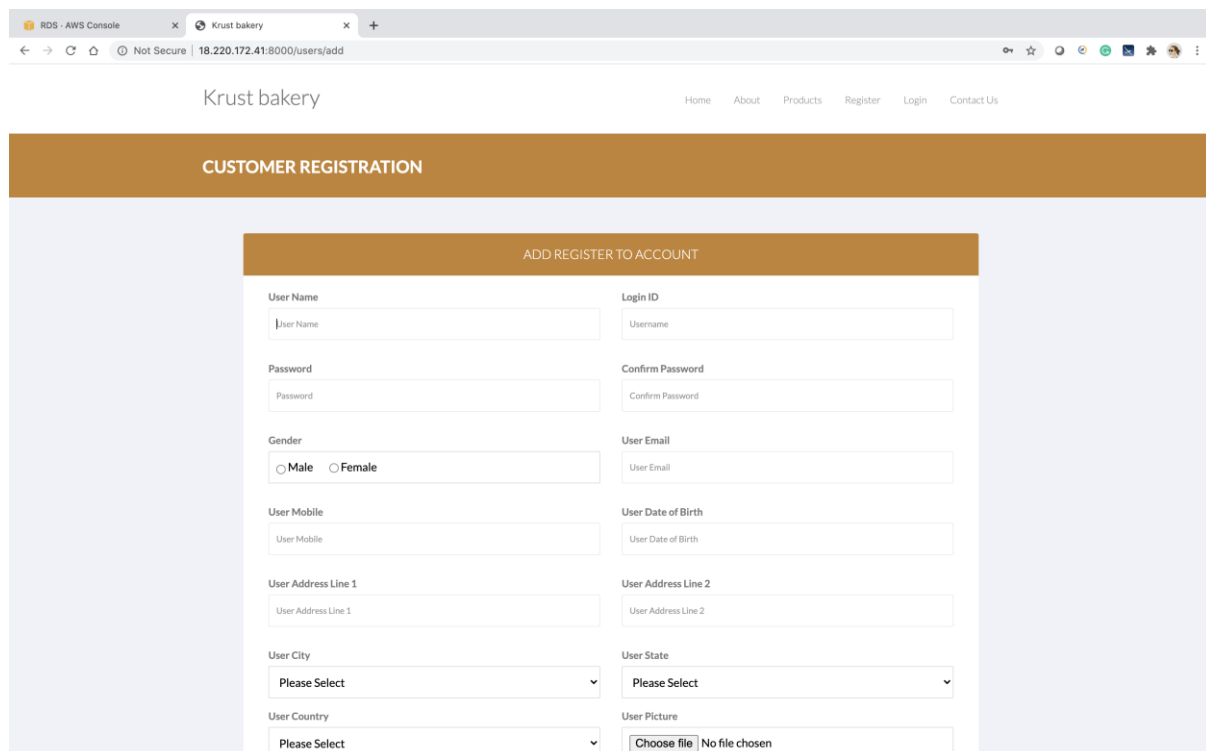
1. Registration of the customer
2. Login page
3. Products page
4. Cart
5. Checkout
6. Payment
7. Summary of order

8. Invoice of the order

Let's look into what these features include.

1. Registration of the customer

This feature offers the customer to sign up for a login so as to view and purchase the products. This includes a form with details as fields to be filled in. These details include full name of user, preferable username, password, confirm password, gender, e-mail id, mobile number, date of birth, address, and a profile picture to be uploaded. When user fills in all these details, they are stored in the table that we have created in the previous section. Validation is performed here in between password and confirm password should be same otherwise it shows an error that they are not same. And the username, if the name is already in the database it would prompt for a change in the username as it is the key that is sued in the data base table.



The screenshot shows a web browser window with the URL `18.220.172.41:8000/users/add`. The page title is "Krust bakery" and the navigation menu includes "Home", "About", "Products", "Register", "Login", and "Contact Us". The main heading is "CUSTOMER REGISTRATION". Below it is a form titled "ADD REGISTER TO ACCOUNT". The form contains the following fields:

ADD REGISTER TO ACCOUNT	
User Name	Login ID
<input type="text" value="User Name"/>	<input type="text" value="Username"/>
Password	Confirm Password
<input type="text" value="Password"/>	<input type="text" value="Confirm Password"/>
Gender	User Email
<input type="radio"/> Male <input type="radio"/> Female	<input type="text" value="User Email"/>
User Mobile	User Date of Birth
<input type="text" value="User Mobile"/>	<input type="text" value="User Date of Birth"/>
User Address Line 1	User Address Line 2
<input type="text" value="User Address Line 1"/>	<input type="text" value="User Address Line 2"/>
User City	User State
<input type="text" value="Please Select"/>	<input type="text" value="Please Select"/>
User Country	User Picture
<input type="text" value="Please Select"/>	<input type="text" value="Choose file No file chosen"/>

Figure 9: Registration Form

2. Login page

In this page the user is prompted for the username and password credentials and is validated from the database. If the username and password combo exist in the database and are valid only then the user is allowed to login otherwise it shows an error message. Once the user is logged in successfully the site redirects to the dashboard page in which they can view their information provided during registration, list of orders previously made and an option to change their password. Whenever the user requests for a change password they will be prompted for two fields password and confirm password once they are filled and submitted the user record is updated in the database and then in the front-end it displays a successful change message.

The screenshot shows a web browser window with the URL `ec2-3-17-190-24.us-east-2.compute.amazonaws.com:8000/users/`. The page is titled "Krust bakery" and features a navigation menu with links: Home, About, Products, Register, Login, and Contact Us. The main content area is a login page with a brown header labeled "LOGIN". Below this, a white box contains the login form. The form has a title "LOGIN TO YOUR ACCOUNT" and a message "Login to your account, to buy the product !!!". It includes two input fields: "User ID" and "Password", followed by a "SIGN IN" button. The footer is divided into four sections: "CONTACT INFO" (Address, Mars city, Country, Mobile, Phone, Email), "MODULES" (Shopping Module, User Module, Product Module, Login Module, Payment Module), "PROJECT LINKS" (Home, About Us, Contact, Login, Email Us), and "ABOUT PROJECT" (Online Artting is a form of electronic commerce which allows consumers to directly buy goods or services from a seller over the Internet using a web browser. Consumers find a product of interest by visiting the website of the retailer directly or by searching among alternative vendors using a shopping).

Figure 10: Login Page

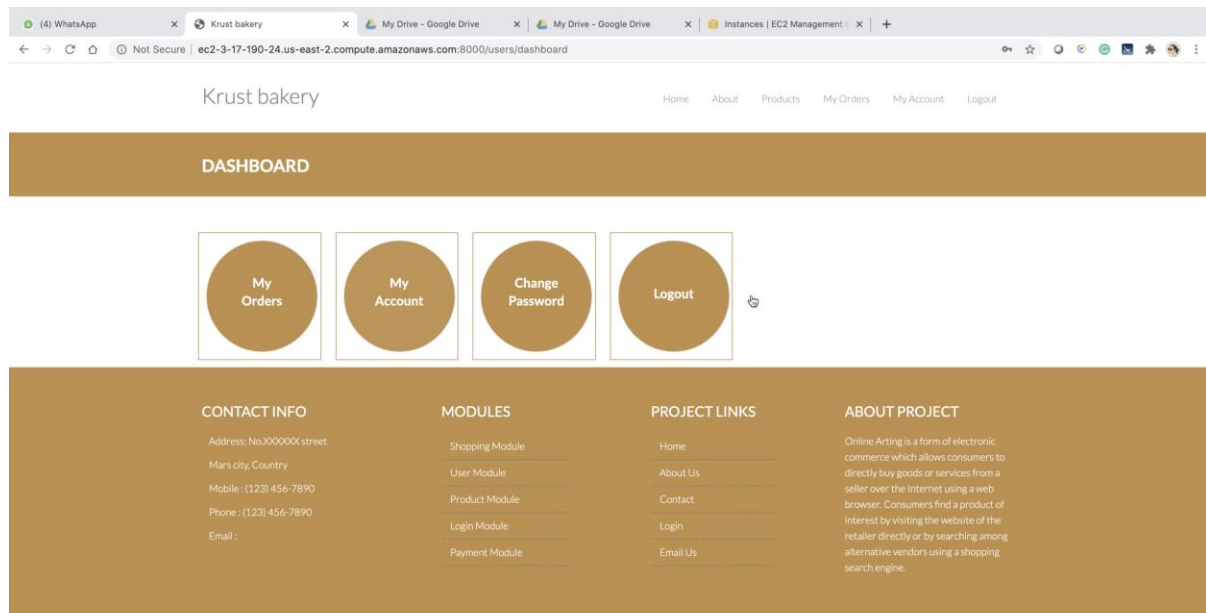


Figure 11: Dashboard

3. Products page

This page retrieves the predefined data from the products table that is created during the implementation of database tables. Any changes that the store wants to make regarding the products that are displayed on the page can be made by updating this table. All the products are displayed along with a thumbnail and brief description of the product including the price and a button to add the item to cart.

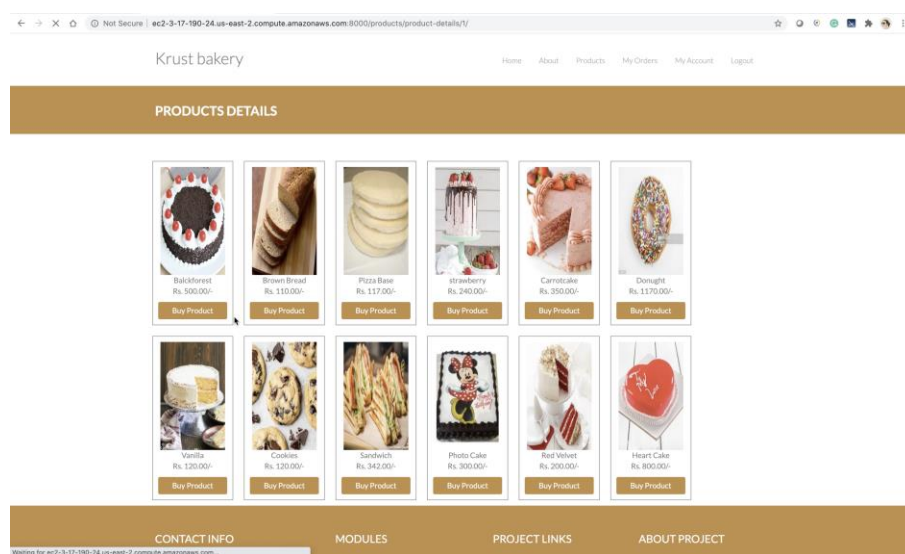


Figure 11: Products Page

4. Cart

Once the user adds the item to the cart the table created for holding the products data which are in the cart is updated. When the user clicks on the cart all this data is retrieved and displayed on the page. This include the name and thumbnail of the product along with price and number of items. It also includes the total amount of the products. Here the user will be provided an option to delete and existing item in the cart, update the quantity of the products that are already in the cart or can go back to the products page to add some more items. Finally, when the user is ready to purchase, they can click on the checkout button which redirects to a checkout page.

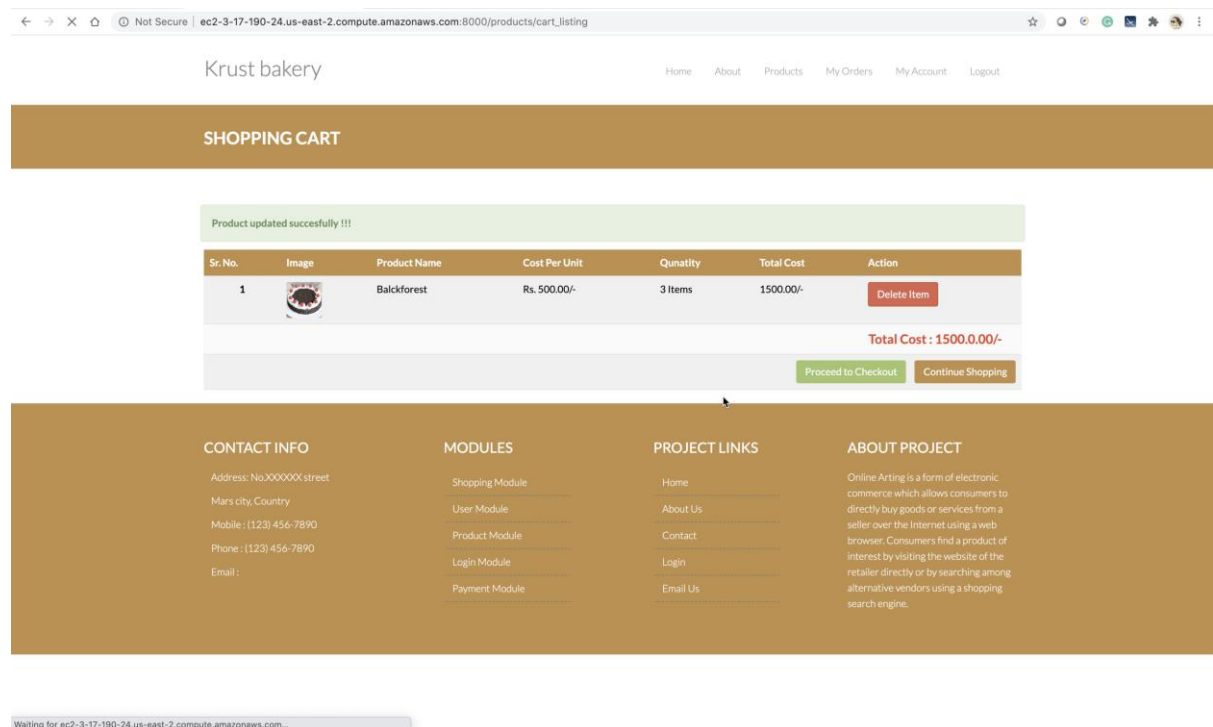


Figure 12: Shopping Cart

5. Checkout and Payment

In the checkout page the user is prompted to pay the amount through an online payment and a form containing the fields to enter the card details is included to get the information on the card and proceed for a transaction. Usually from here a secure gateway is provided to complete the transaction as the system is for education

purpose, the payment process is bypassed by only validating the card information and moving to a page that shows up the payment is successful.

Krust bakery

Home About Products My Orders My Account Logout

MAKE ONLINE PAYMENT

Name of Card:

Card No:

Card Type: Please Select

Card Expiry: Month Year

Amount: 2670.00/-

[Make Payment](#)

CONTACT INFO

Address: No.XXXXXXX street
 Mars city, Country
 Mobile : (123) 456-7890
 Phone : (123) 456-7890
 Email :

MODULES

Shopping Module
 User Module
 Product Module
 Login Module
 Payment Module

PROJECT LINKS

Home
 About Us
 Contact
 Login
 Email Us

ABOUT PROJECT

Online Artting is a form of electronic commerce which allows consumers to directly buy goods or services from a seller over the Internet using a web browser. Consumers find a product of interest by visiting the website of the retailer directly or by searching among alternative vendors using a shopping search engine.

Figure 13: Checkout and Payment

6. Summary of order

Once the payment is successful the user is redirected to page that summarizes the order details which includes the order details and amount paid.

Krust bakery

Home About Products My Orders My Account Logout

ORDER RECEIPT AND DETAILS

Customer and Order Details

Order ID	16
Date	11:01AM on June 18, 2020
Order Status	Confirmed
Customer Name	vinaya srini
Email	vinayasrini715@gmail.com
Contact	09985875149

Sr. No.	Image	Product Name	Cost Per Unit	Quantity	Total Cost
1		Bakchforest	Rs. 500.00/-	3 Items	1500.00/-
2		Donught	Rs. 1170.00/-	1 Items	1170.00/-
					Total Cost : 2670.00/-

[Print Receipt](#)

CONTACT INFO

Address: No.XXXXXXX street
 Mars city, Country
 Mobile : (123) 456-7890

MODULES

Shopping Module
 User Module
 Product Module
 Login Module
 Payment Module

PROJECT LINKS

Home
 About Us
 Contact
 Login
 Email Us

ABOUT PROJECT

Online Artting is a form of electronic commerce which allows consumers to directly buy goods or services from a seller over the Internet using a web browser. Consumers find a product of interest by visiting the website of the retailer directly or by searching among alternative vendors using a shopping search engine.

Figure 14: Order Summary

7. Invoice of the order

Invoice copy or a receipt of the order that is made is available when the user clicks on the print receipt button. This exports as a pdf file with all the information related to the order and this can be downloaded or printed.

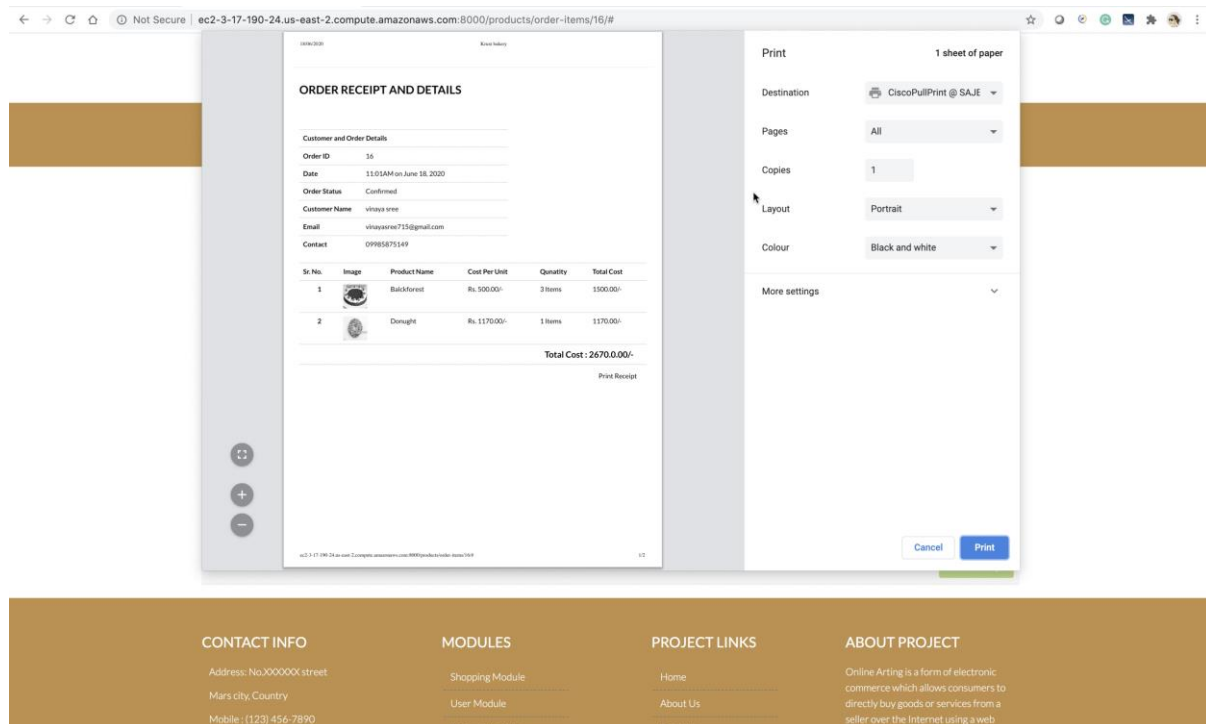


Figure 15: Invoice of Order

Conclusion

An online bakery store name krust bakery is successfully implemented using python and Django with the features including customer registration, login, place order, payment and summary of it. It also includes a home and contact us page in which the address of the store and related information are added along with a Google map API to locate the store easily. Finally, a cloud deployment is done using AWS and RDS database service using MYSQL is up and running. When we want to deploy into the local machine tomcat server is used to run the application. Future enhancement of the project can be addition of a secured gateway so that the bypassed payment system can be implemented and can be used in production for a real time project.

Bibliography

Python Software Foundation. Python Language Reference, version 3
Available at <http://www.python.org>

Working with MySQL. MySQL documentation.
Available at <https://dev.mysql.com/doc/refman/8.0/en/packages.html>

Deploy a Python WebApp. AWS documentation.
Available at <https://aws.amazon.com/getting-started/projects/deploy-python-application/>