A
Project Report
On
"E-commerce Web Application using MERN Stack including Payment Gateway"

Prepared by

Ruchik Pravasi (17DCE056)
Achal Rajyaguru (17DCE058)
Tirthkumar Shah (17DCE065)

Under the guidance of

Dr. Dweepna Garg

HOD Computer Engineering

A Report Submitted to

Charotar University of Science and Technology

for Partial Fulfillment of the Requirements for the

7th Semester Software Group Project-IV (CE446)

Submitted at

CE

DEPSTAR

At: Changa, Dist: Anand – 388421

Nov 2020

CERTIFICATE

This is to certify that the report entitled "**E-commerce Web Application using MERN Stack including Payment Gateway**" is a bonafied work carried out by **Mr. Tirthkumar Shah(17DCE065)** under the guidance and supervision of **Prof. Dweepna Garg** for the subject CE446 **Software Group Project-IV** (CE) of 7th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Name of Internal Guide
Dr. Dweepna Garg
Head of Department
Computer Engineering
DEPSTAR, Changa, Gujarat.

Dr. Amit Ganatra
Principal, DEPSTAR
Dean, FTE
CHARUSAT, Changa, Gujarat.

CERTIFICATE

This is to certify that the report entitled "**E-commerce Web Application using MERN Stack including Payment Gateway**" is a bonafied work carried out by **Mr. Ruchik Pravasi(17DCE056)** under the guidance and supervision of **Prof. Dweepna Garg** for the subject CE446 **Software Group Project-IV** (CE) of 7th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Name of Internal Guide
Dr. Dweepna Garg
Head of Department
Computer Engineering
DEPSTAR, Changa, Gujarat.

Dr. Amit Ganatra
Principal, DEPSTAR
Dean, FTE
CHARUSAT, Changa, Gujarat.

CERTIFICATE

This is to certify that the report entitled "**E-commerce Web Application using MERN Stack including Payment Gateway**" is a bonafied work carried out by **Mr. Achal Rajyaguru(17DCE058)** under the guidance and supervision of **Prof. Dweepna Garg** for the subject CE446 **Software Group Project-IV** (CE) of 7th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Name of Internal Guide
Dr. Dweepna Garg
Head of Department
Computer Engineering
DEPSTAR, Changa, Gujarat.

Dr. Amit Ganatra
Principal, DEPSTAR
Dean, FTE
CHARUSAT, Changa, Gujarat.

ACKNOWLEDGEMENT

- The Success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of my project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.
- We are extremely thankful to HOD, Department of Computer Engineering, Dr. Dweepna Garg for providing such a nice support and guidance, although she had busy schedule managing the college work. We owe our deep gratitude to our project guide, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good project. Also encouraged us till the completion of our project work.

ABSTRACT

- Today Developers around the world are making efforts to enhance user experience of using application as well as they are trying to enhance the developer's workflow of designing applications to deliver projects and rollout change requests under strict timeline. Stacks can be used to build web applications in the shortest span of time. The advantage of such JavaScript stacks helps to build an integrated solution by using only JavaScript. This project provides the introduction to concept and describes the MERN stack open source.
- The stacks used in web development are basically the response of software engineers to current demands. They have essentially adopted pre-existing frameworks (including JavaScript) to make their lives easier. While there are many, MEAN and MERN are just two of the popular stacks that have evolved out of JavaScript. Both of these stacks are made up of open source components and offer an end-to-end framework for building comprehensive web apps that enable browsers to connect with databases. The common theme between the two is JavaScript and this is also the key benefit of using either stack. You can basically avoid any syntax errors or any confusion by just coding in one programming language, JavaScript. Another advantage of building your next web project with MEAN or MERN is the fact that you benefit from its enhanced flexibility.
- The term MERN stack refers to a collection of JavaScript based technologies used to develop web applications. MERN is an acronym for MongoDB, ExpressJS, ReactJS and Node.js. From client to server to database, MERN is full stack JavaScript.
- MongoDB is a schema less NoSQL database system. Express is lightweight framework used to build web applications in Node. React is an open-source, front end, JavaScript library for building user interfaces or UI components. Node.js is a server-side JavaScript execution environment.
- The main advantage for developing this web application using the MERN stack is that every line of code is written in JavaScript. This is a programming language that's used everywhere, both for client-side code and server-side code. With one language across tiers, there's no need for context switching. For tech stack with multiple programming languages, developers have to figure out how to interface them together. With the JavaScript stack, developers only need to be proficient in JavaScript and JSON.
- MEAN Stack is one of the fastest growing open source stack development frameworks assists developer or teams with popular tools or plugins to reduces the time on system administration and it also allows the quicker deployment of web apps, websites and API's to concentrate on the complex development process of your project.
- Overall, using the MERN stack enables developers to build highly efficient web applications.

Table of Contents

Table of Figures

1. Project Definition

- An E-commerce web application with administrator and user module which has PayPal and Stripe payment gateway with backend admin panel and deployed on AWS Cloud.

2. Description

- An E-commerce web application which has two modules, admin and user. Whole web application is developed using MERN stack and deployed using AWS. MERN stack consists of MongoDB, ExpressJS, ReactJS and NodeJS.
- With the help of NodeJS, we created the server-side operations. With the help of ReactJS, we created the frontend of our web application. ExpressJS helped us to develop the actual web application. MongoDB was our database, the reason for mongo dB is it has high compatibility with NodeJS and huge community.
- User mode:
- User needs to signup if user visits for the first time. So next time when the user visits the web application, he will be authenticated as a user. After sign in, user will be redirected to home page where all the products will be displayed. User can scroll through the products and add to the cart. In the cart, user can place order for the specific product or all the products in the cart. After placing order, user will be redirected to payment gateway, where he/she will have to pay using PayPal.
- Admin mode:
- Admin needs to signup if admin visits for the first time. So next time when the admin visits the web application, he will be authenticated as a admin. After sign in, admin will be redirected to home page, where all the details of the products and operations, which only admin can do, will be displayed. Admin can create/manage the products, can create/manage the category, manage the order placed, and also manage the fulfilled orders.
- During authentication the password of user/admin is encrypted, using the NodeJS library and functions and there after stored in database. When the user/admin visits again, they are assigned a token in the backend which has a value, to identify through the web application who is admin and who is user.

3. Hardware Requirements (minimum)

Memory	Graphics Card	CPU	OS
4 GB	NVIDIA GeForce 6100	Intel Pentium 4 2.00GHz	Windows 7, 8.1, 10 or Mac OS X

4. Web Application Architecture

4.1 Backend

4.1.1 Software Requirements

Software	Description
NodeJS	Node.js is a free, open-sourced, cross-platform JavaScript run-time environment that lets developers write command line tools and server-side scripts outside of a browser.
ExpressJS	Express.js is a free and open-source web application framework for Node.js. It is used for designing and building web applications quickly and easily. Web applications are web apps that you can run on a web browser.

4.1.2 Screenshot

1. NodeJs Logo

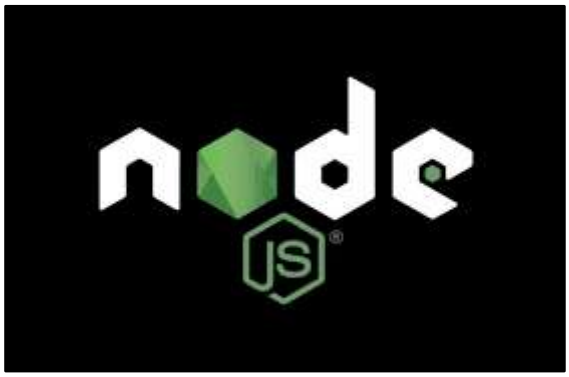


Figure 1 NodeJS

2. Backend Folder Structure

his PC › Desktop › T-Shirt Store using MERN › projbackend ›

Name	Date modified	Type	Size
controllers	20-10-2020 03:33 PM	File folder	
models	20-10-2020 03:33 PM	File folder	
node_modules	21-10-2020 12:45 PM	File folder	
routes	20-10-2020 03:33 PM	File folder	
.env	22-10-2020 01:53 PM	ENV File	1 KB
app.js	22-10-2020 03:54 PM	JS File	2 KB
dev.js	22-10-2020 01:53 PM	JS File	1 KB
package.json	24-02-2020 03:02 PM	JSON File	1 KB
package-lock.json	24-02-2020 03:02 PM	JSON File	185 KB

Figure 2 Backend Structure

3. Models Files

Name	Date modified	Type	Size
category.js	22-01-2020 04:25 PM	JS File	1 KB
order.js	23-02-2020 11:17 PM	JS File	1 KB
product.js	22-01-2020 09:42 PM	JS File	1 KB
user.js	25-01-2020 04:50 PM	JS File	2 KB

Figure 3 Models Files

4. Controllers Files

PC › Desktop › T-Shirt Store using MERN › projbackend › controllers

Name	Date modified	Type	Size
auth.js	28-01-2020 07:02 PM	JS File	3 KB
category.js	31-01-2020 01:54 PM	JS File	2 KB
order.js	06-02-2020 06:23 PM	JS File	2 KB
paymentb.js	20-10-2020 04:18 PM	JS File	1 KB
product.js	06-02-2020 04:54 PM	JS File	5 KB
user.js	30-01-2020 05:49 PM	JS File	2 KB

Figure 4 Controllers Files

5. Routes Files

is PC › Desktop › T-Shirt Store using MERN › projbackend › routes

Name	Date modified	Type	Size
auth.js	25-10-2020 02:02 PM	JS File	1 KB
category.js	25-10-2020 02:02 PM	JS File	1 KB
order.js	06-02-2020 06:19 PM	JS File	1 KB
paymentBRoutes.js	20-10-2020 08:03 PM	JS File	1 KB
product.js	25-10-2020 02:02 PM	JS File	2 KB
user.js	30-01-2020 04:13 PM	JS File	1 KB

Figure 5 Routes Files

4.2 Frontend

4.2.1 Software Requirements

Software	Description
ReactJS	React is a JavaScript library that specializes in helping developers build user interfaces, or UIs. In terms of websites and web applications, UIs are the collection of on-screen menus, search bars, buttons, and anything else someone interacts with to USE a website or app.
Bootstrap	Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.
CSS	CSS stands for Cascading Style Sheets. CSS describes how HTML elements are to be displayed on screen, paper, or in other media. It can control the layout of multiple web pages all at once.

4.2.2 Screenshot

1. SignIn Page

Figure 6 SignIn

2. SignUp Page

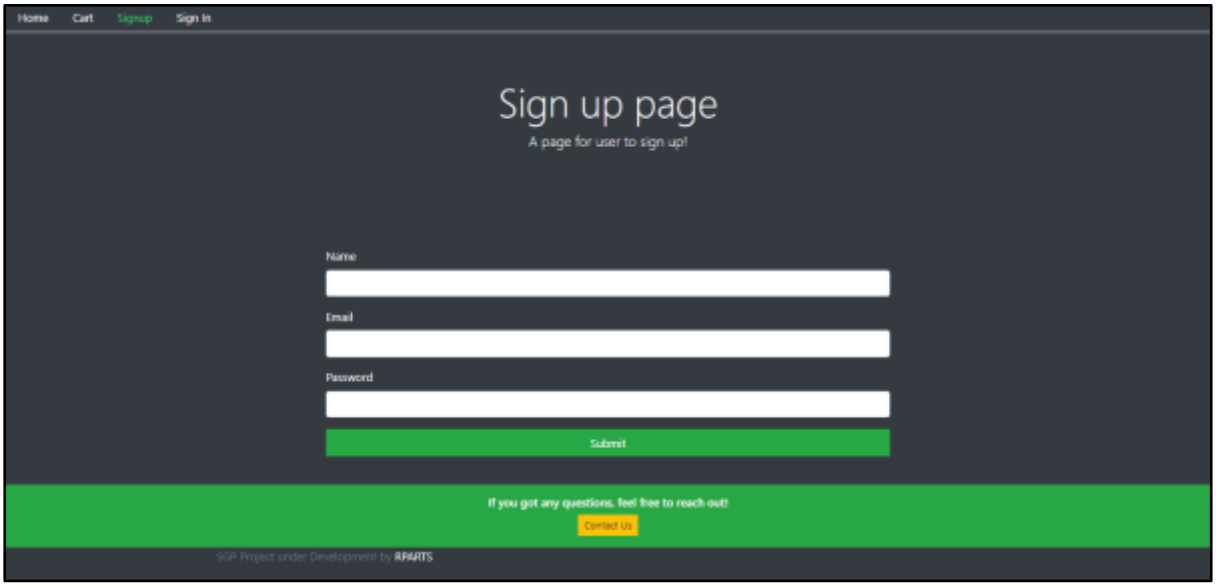


Figure 7 SignUp

3. Home Page

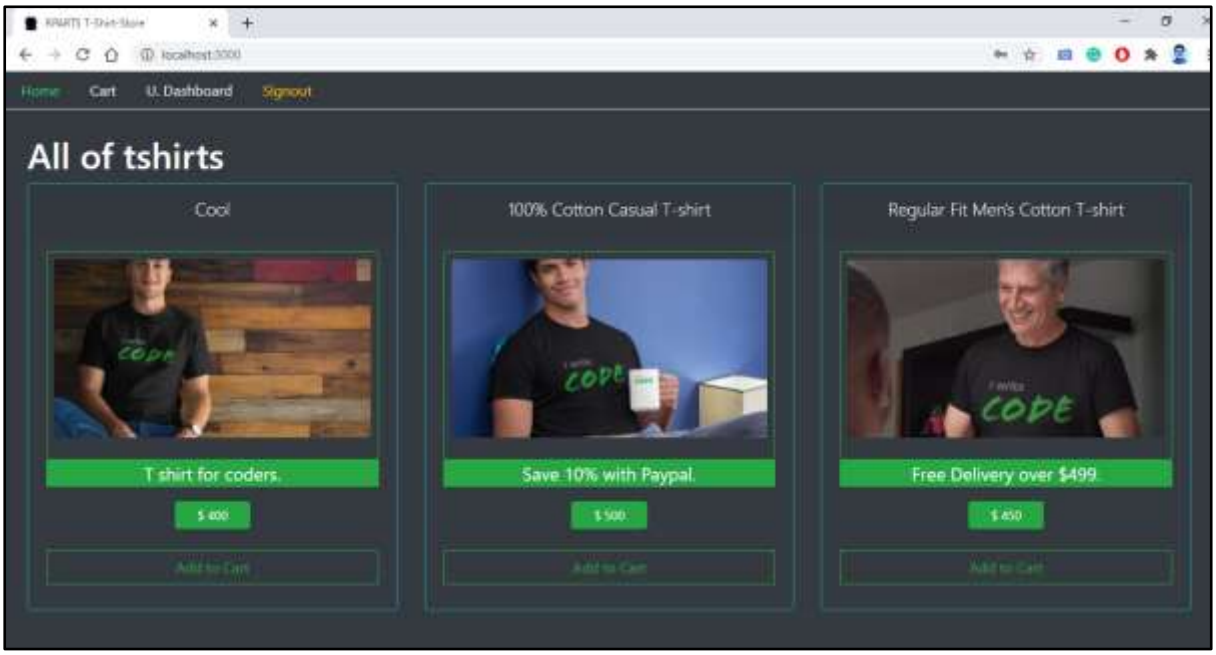


Figure 8 Home Page

4. Admin Dashboard

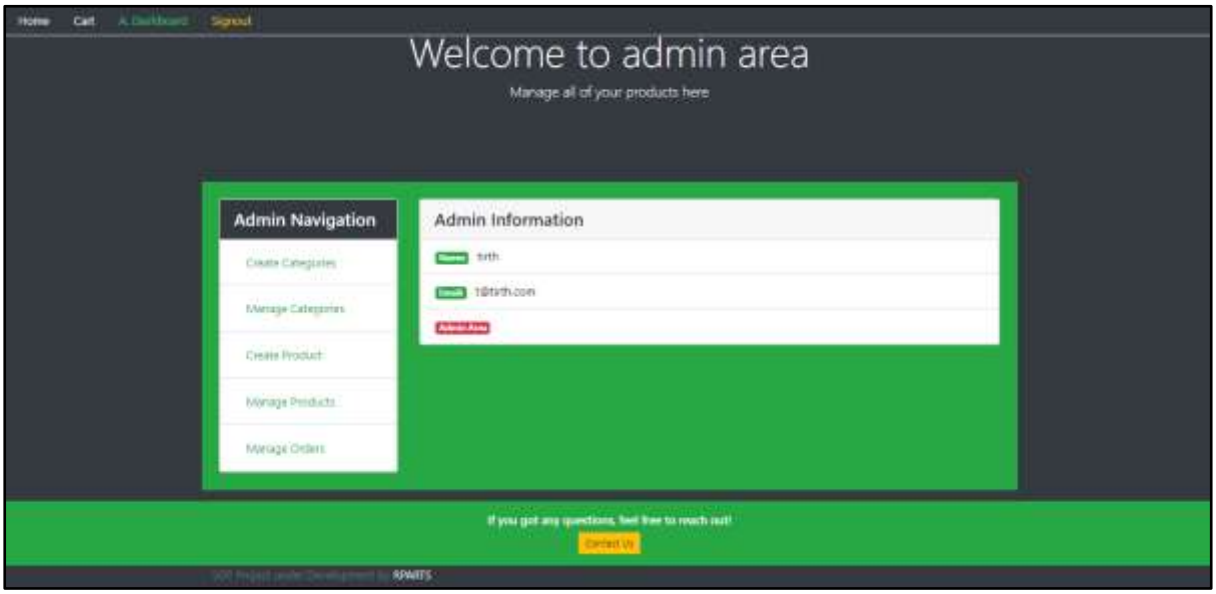


Figure 9 Admin Dashboard

5. Create Product

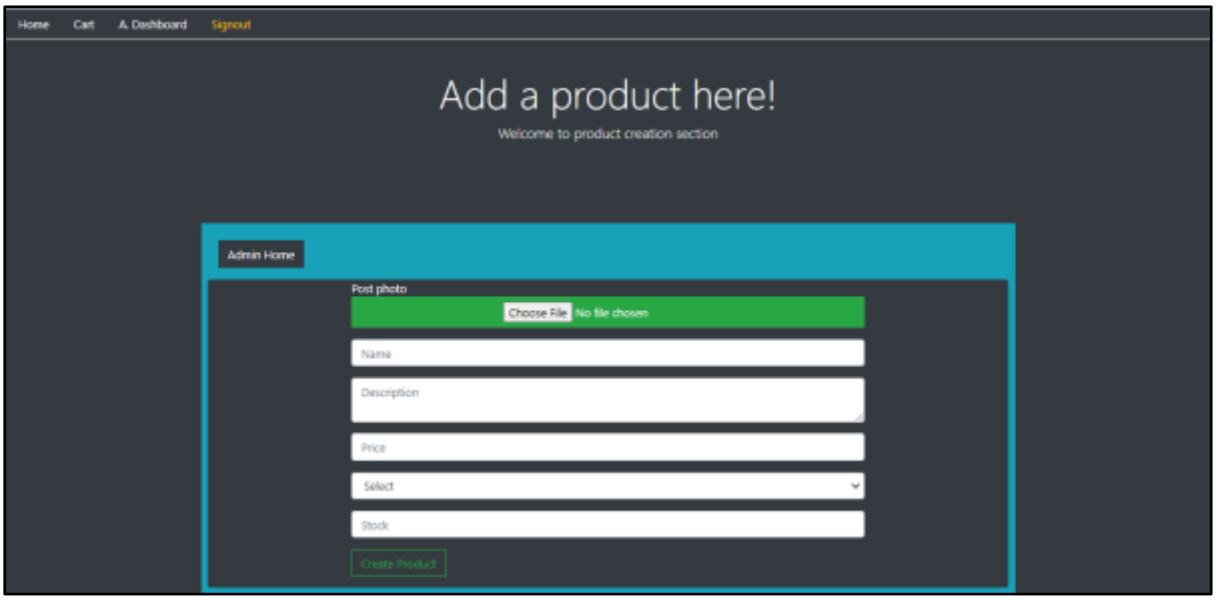


Figure 10 Create Product

6. Create Category

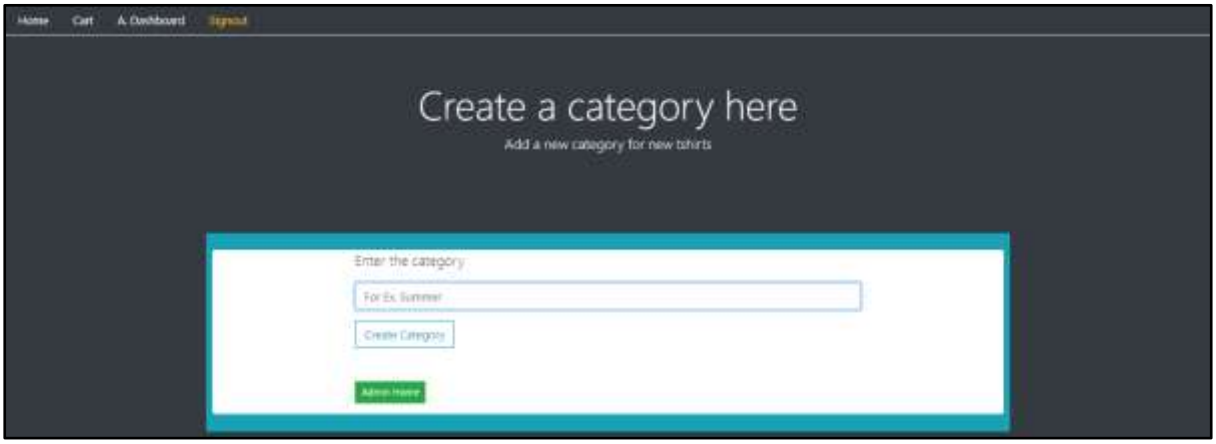


Figure 11 Create Category

7. Manage Product

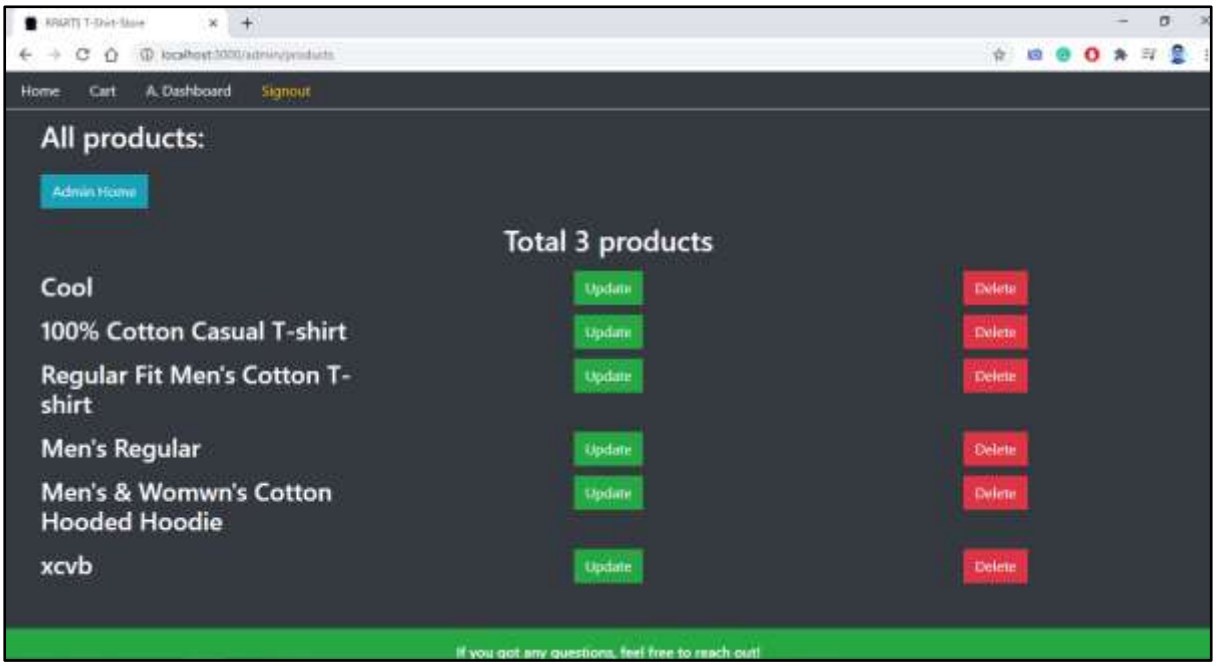


Figure 12 Manage Product

8. Update Product

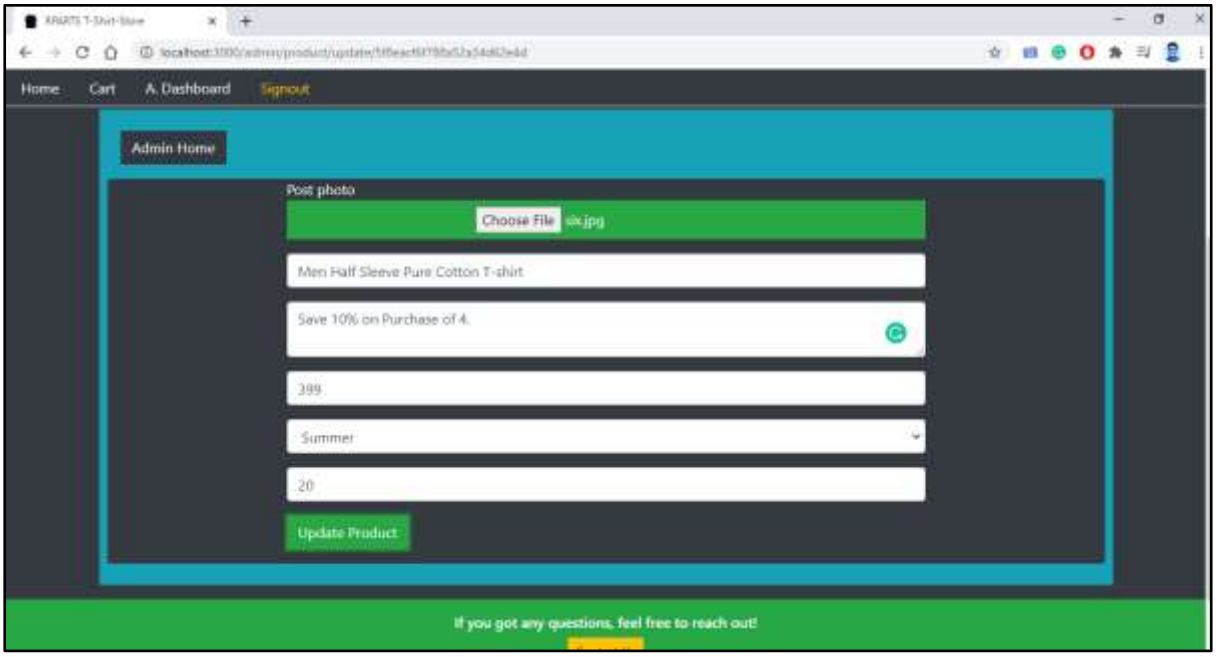


Figure 13 Update Product

4.3 Database

4.3.1 Software Requirements

Software	Description
MongoDB	MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

4.3.2 Screenshot

1. User Database

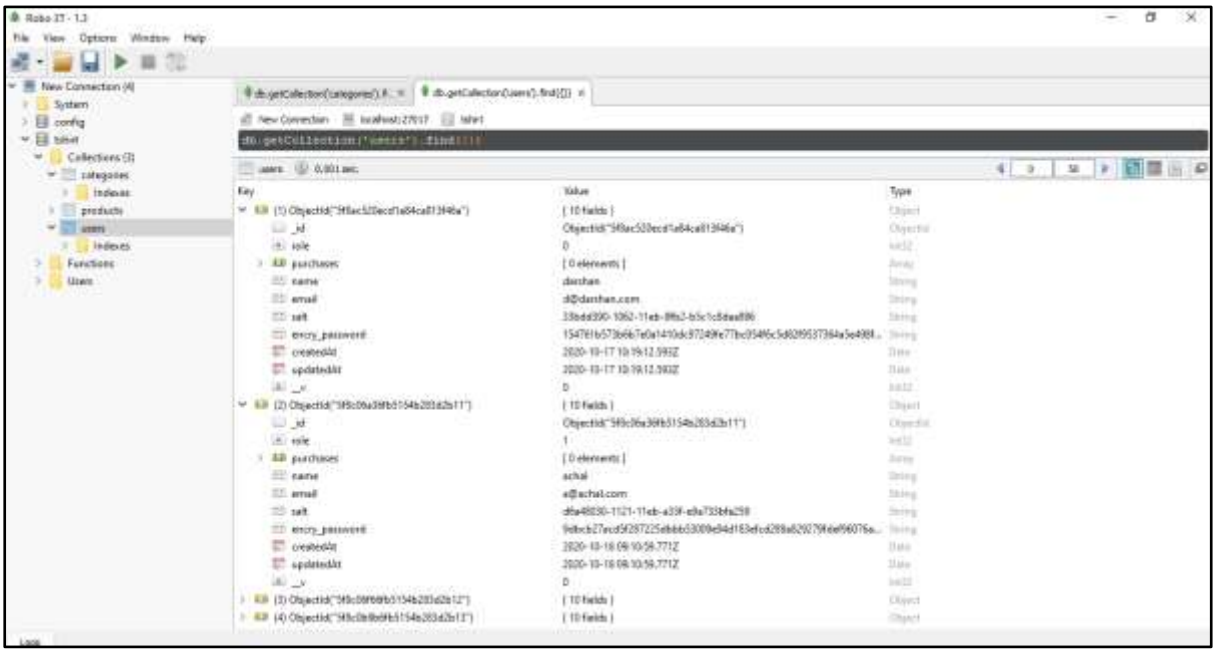


Figure 14 User Database

2. Category Database

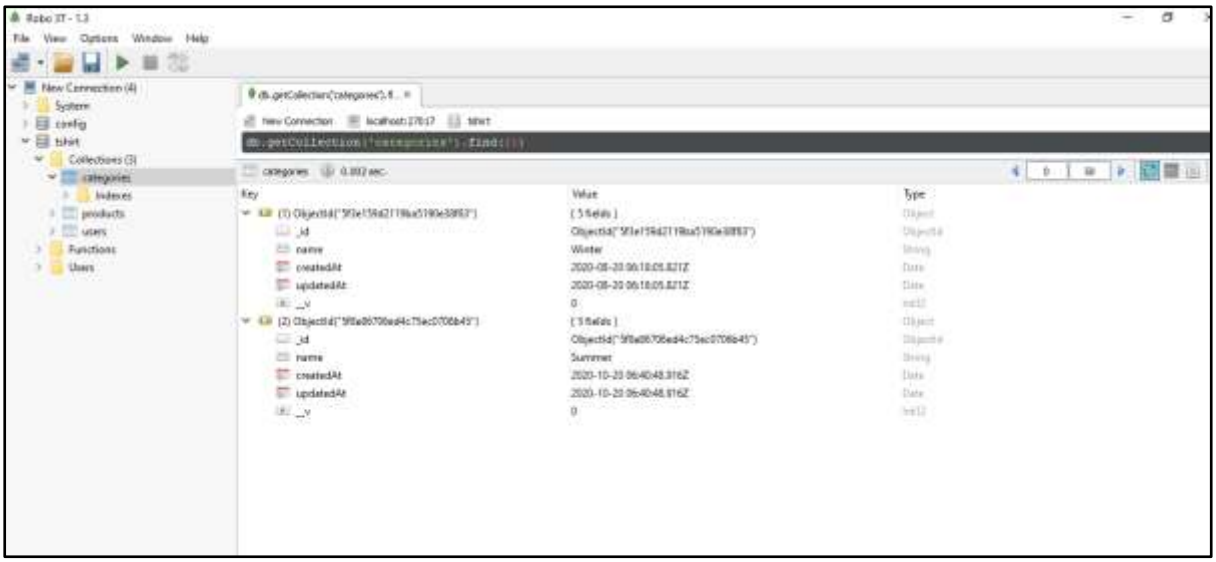


Figure 15 Category Database

3. Product Database

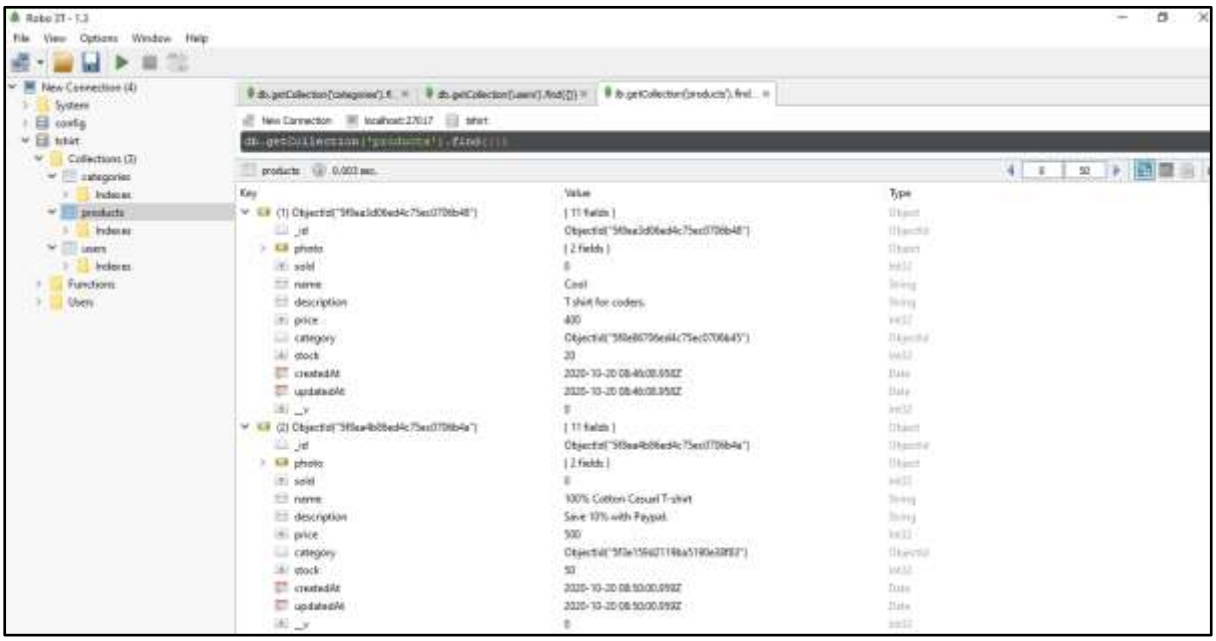


Figure 16 product Database

4.4 Cloud Deployment

4.4.1 Software Requirements

Software	Description
AWS	Amazon Web Services offers a broad set of global cloud-based products including compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security and enterprise applications. These services help organizations move faster, lower IT costs, and scale.

4.4.2 Screenshot

1. **Choose AMI in Amazon EC2 (Elastic Compute Cloud)**

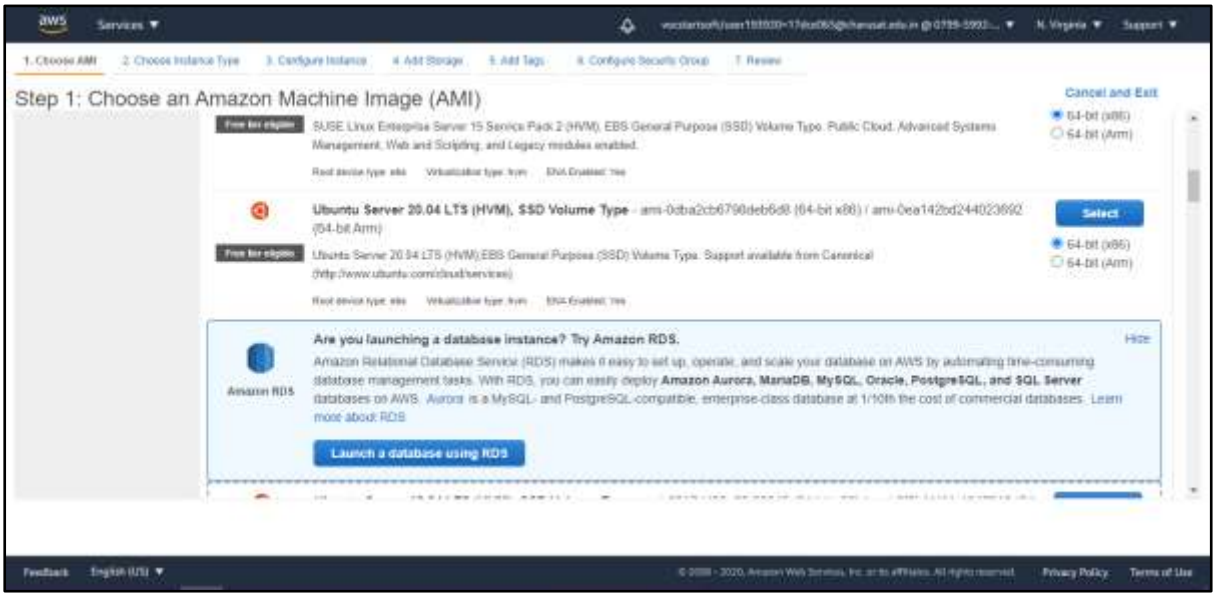


Figure 17 Choose Amazon Machine Image

2. **Instance Type:**

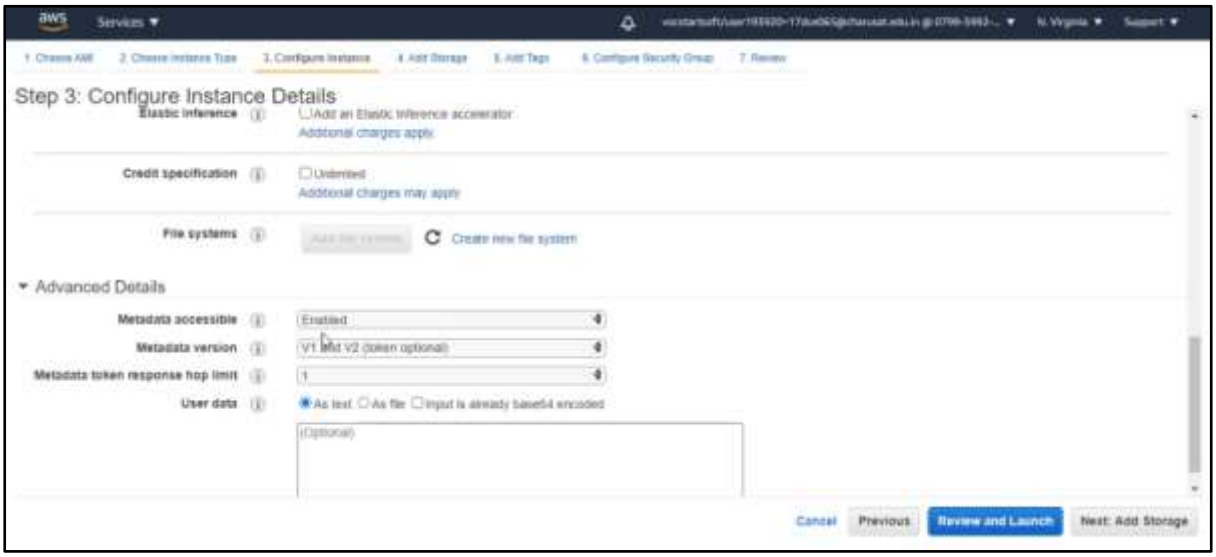


Figure 18 Instance Type of Amazon EC2

3. Security Group Configuration

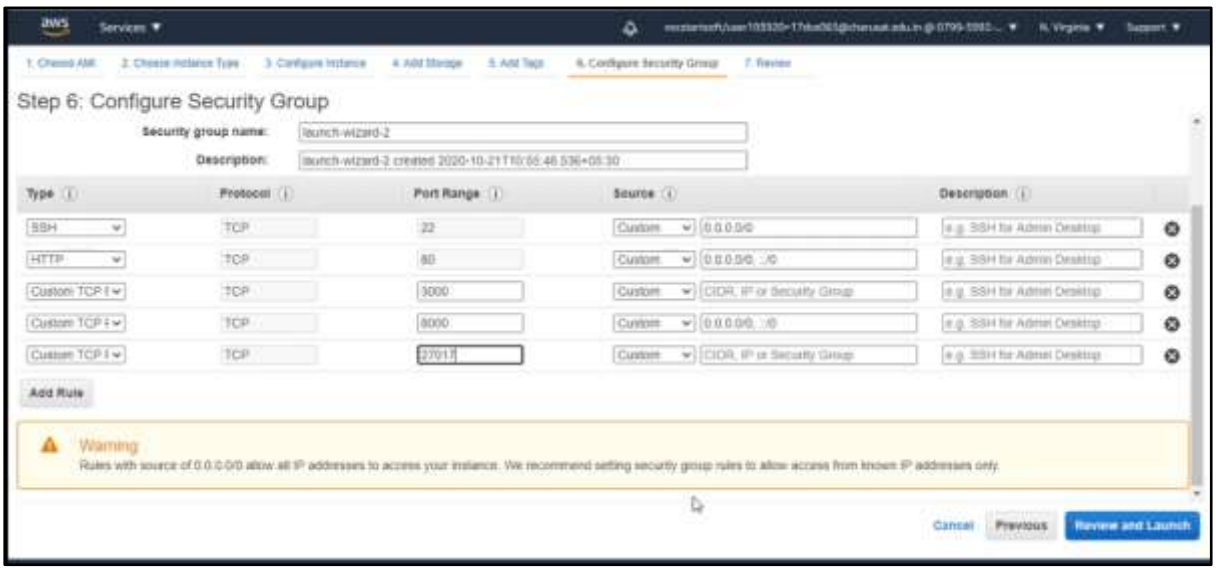


Figure 19 Configuration of Security Group

4. Download .pem File

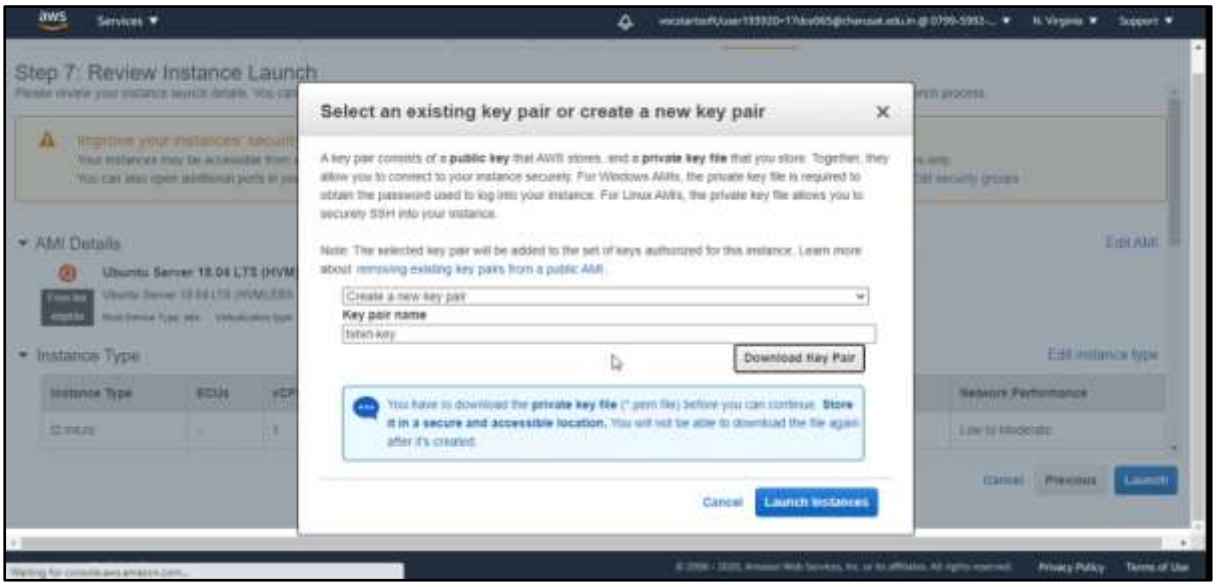


Figure 20 Pem File Download

5. Amazon EC2 instance creation successful

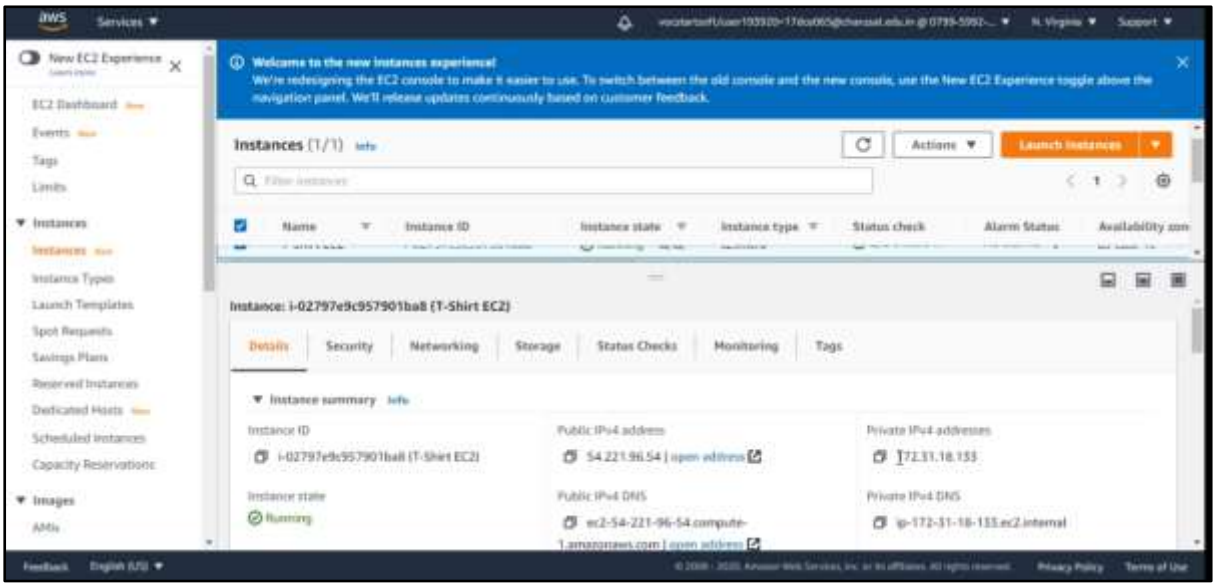


Figure 21 EC2 Creation

6. Connect to EC2 instance via SSH

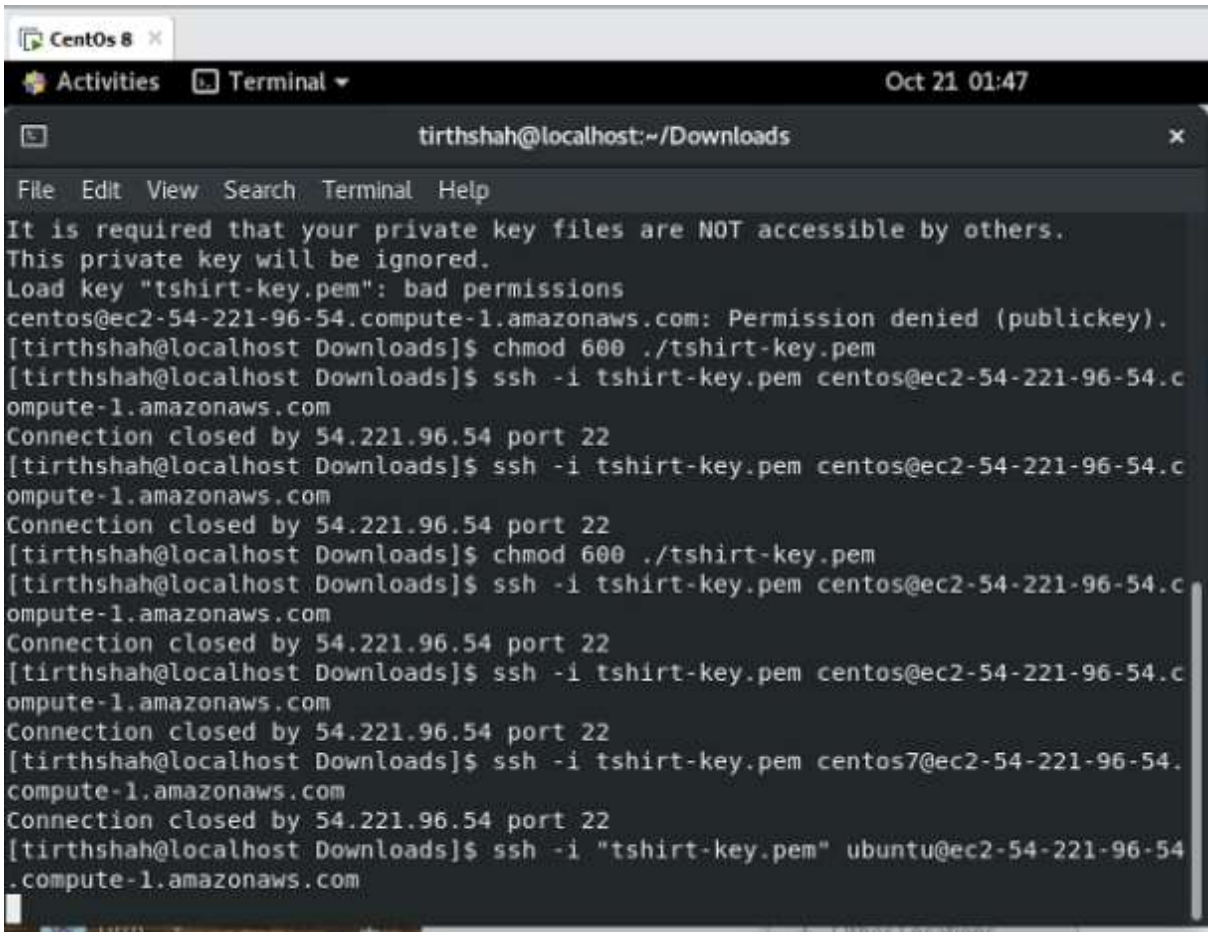


Figure 22 SSH Connection

7. Successfully connected through SSH

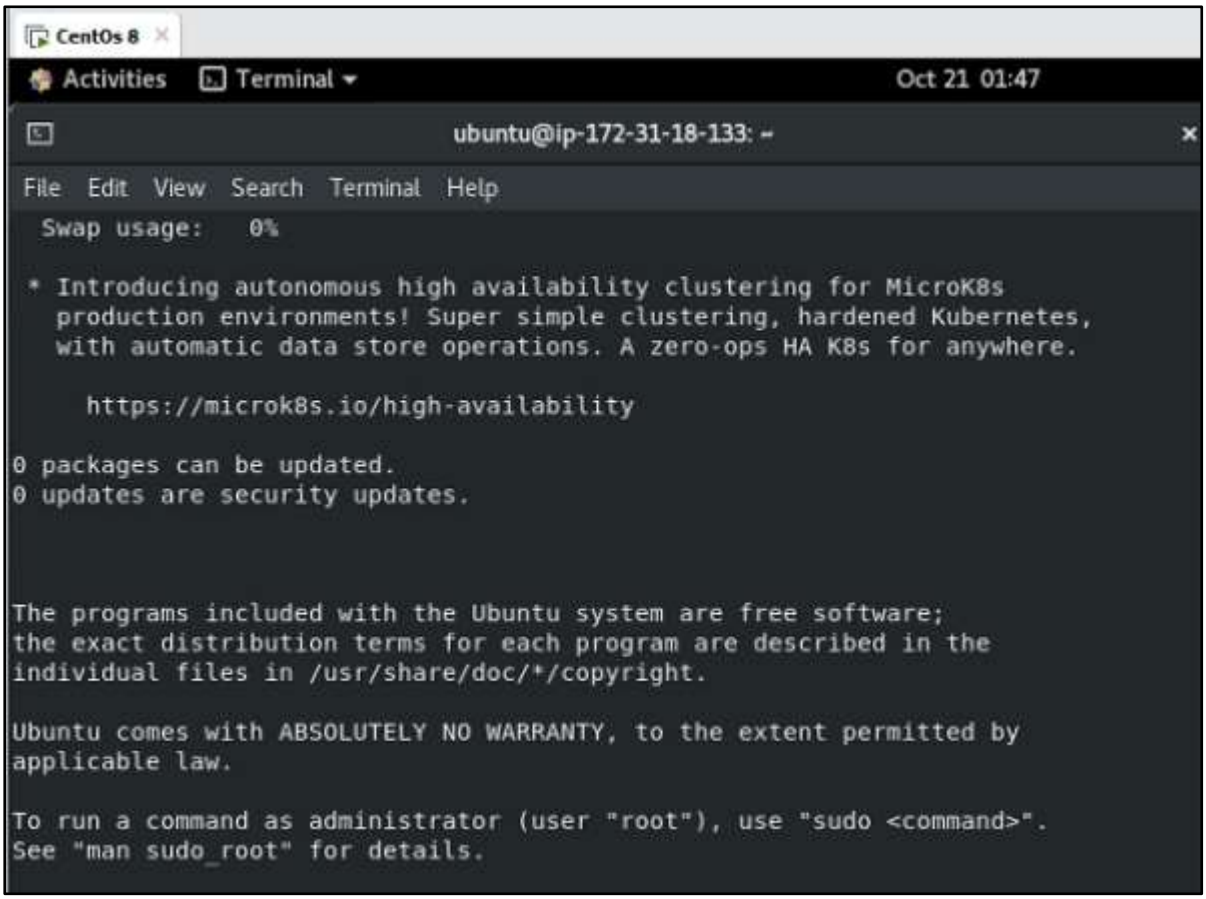


Figure 23 Connected Successfully through SSH

8. Updating the Ubuntu OS

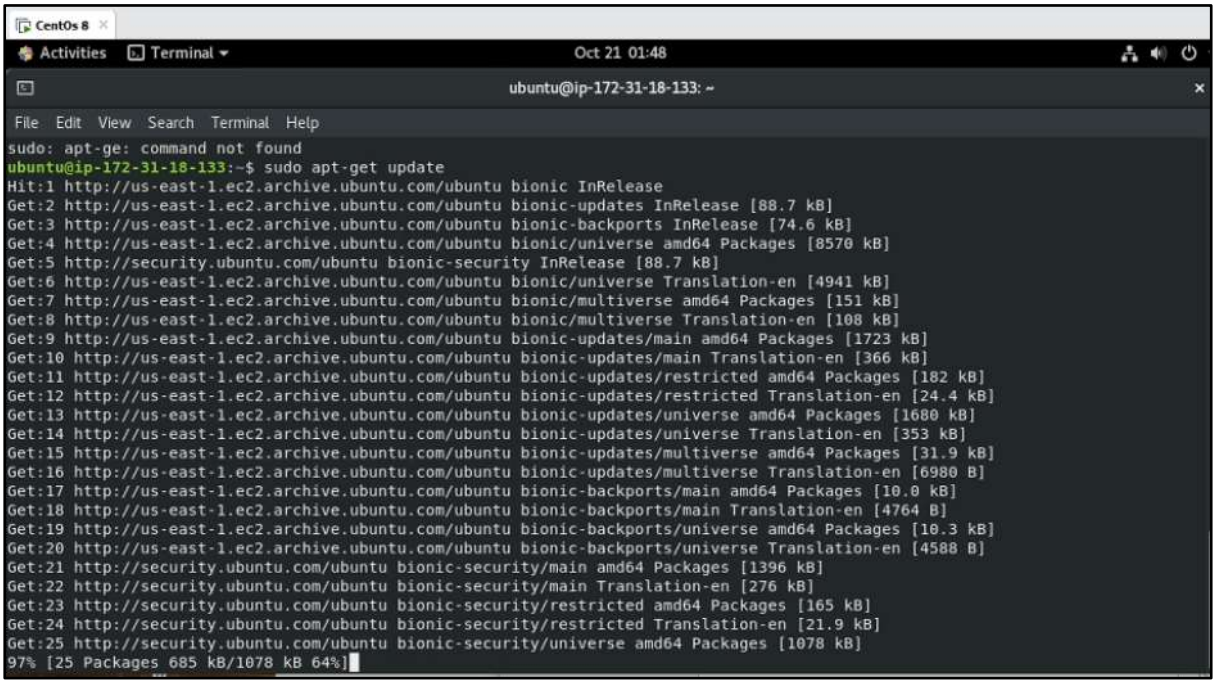


Figure 24 Updating Ubuntu OS

9. Download FileZilla for transferring the files

Figure 25 FileZilla Installation

10. Connect with EC2 instance

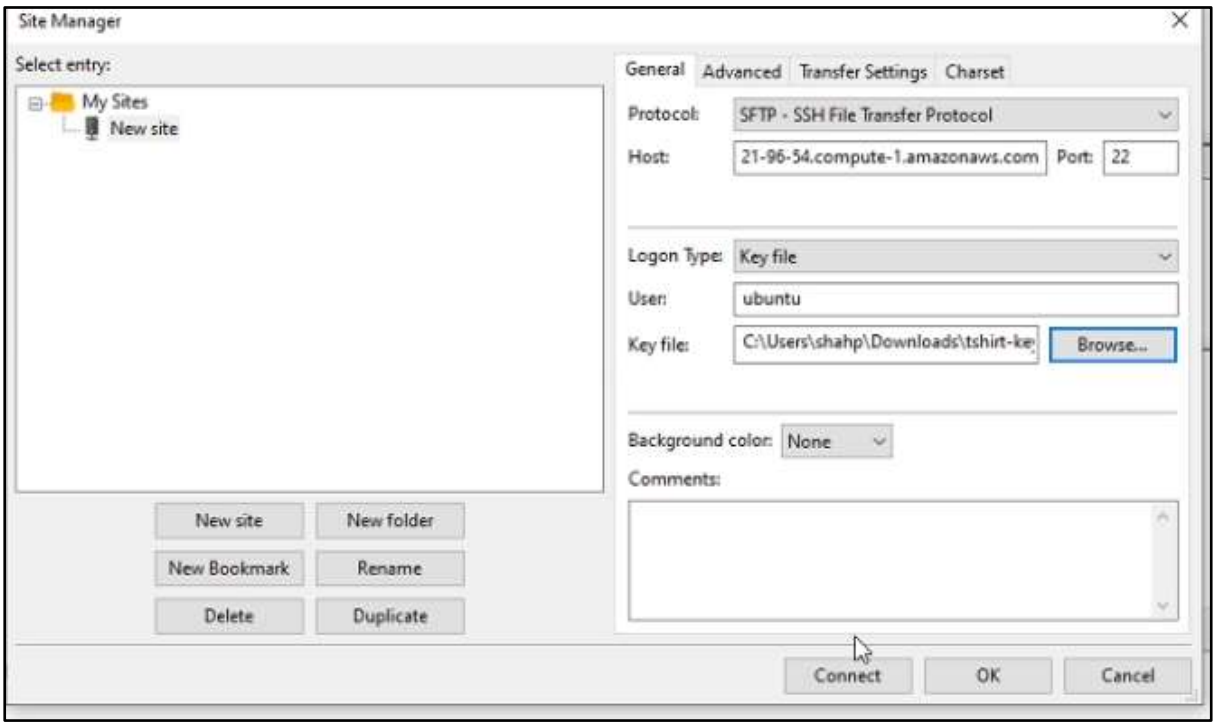


Figure 26 EC2 Connection in FileZilla

11. Connection successfully in EC2 through FileZilla

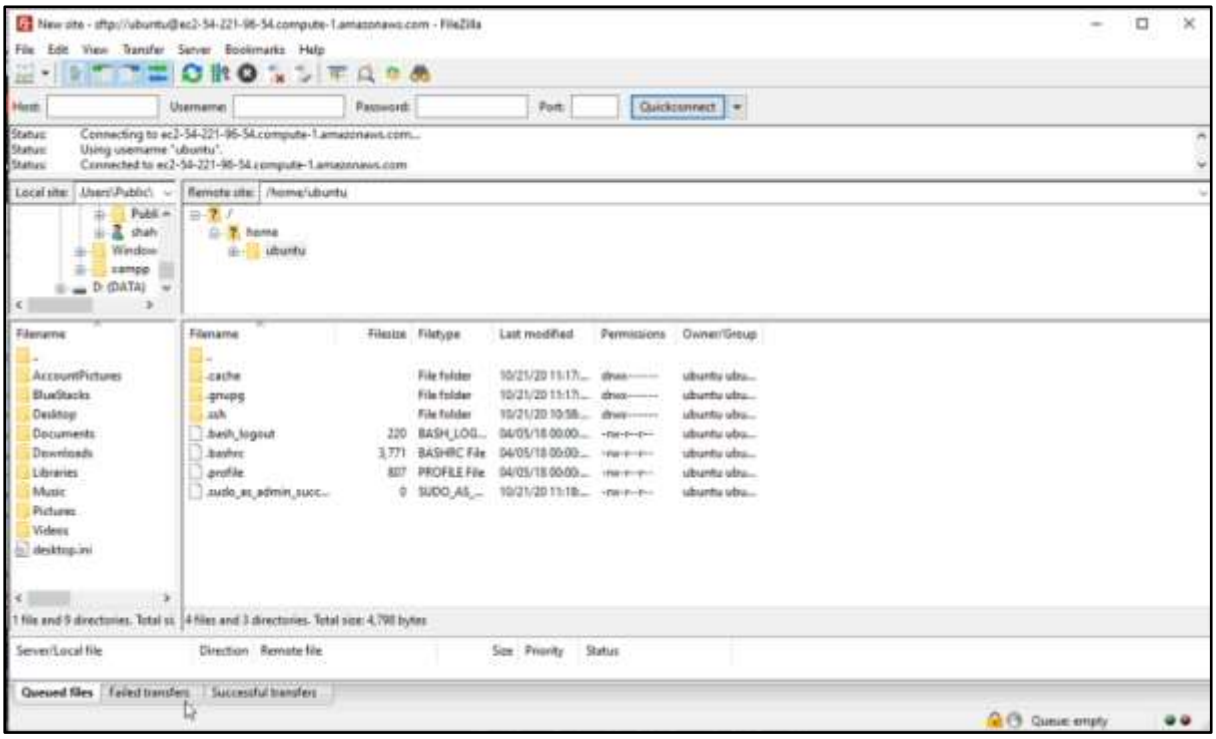


Figure 27 Connected with EC2

12. Create two folders in Ubuntu Directory

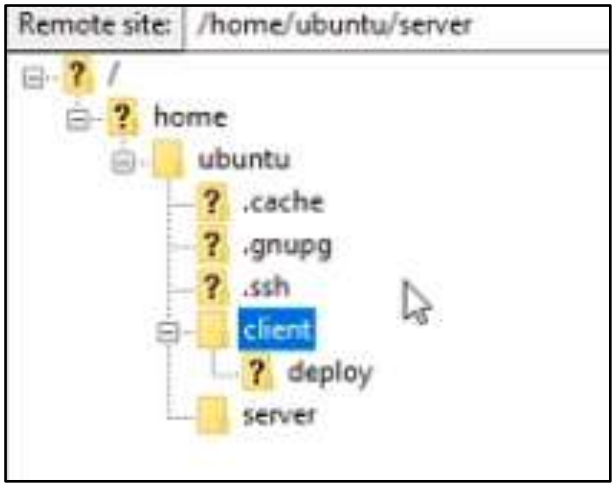


Figure 28 Ubuntu Server Structure

13. Installing NodeJS in Server

```
Activities  Terminal ▾                                  Oct 21 02:07
                                                ubuntu@ip-172-31-18-133: ~
File  Edit  View  Search  Terminal  Help
ubuntu@ip-172-31-18-133:~$ sudo apt-get install nodejs
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libc-ares2 libhttp-parser2.7.1 nodejs-doc
The following NEW packages will be installed:
  libc-ares2 libhttp-parser2.7.1 nodejs nodejs-doc
0 upgraded, 4 newly installed, 0 to remove and 49 not upgraded.
Need to get 5606 kB of archives.
```

Figure 29 NodeJs Installation

14. Installing Nginx Web Server

```
ubuntu@ip-172-31-18-133:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libtiff5 libwebp6
  libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libtiff5 libwebp6
  libxpm4 nginx nginx-common nginx-core
0 upgraded, 18 newly installed, 0 to remove and 49 not upgraded.
Need to get 2462 kB of archives.
After this operation, 8210 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Figure 30 Nginx Installation

15. Start the Nginx Service

```
ubuntu@ip-172-31-18-133:~$ sudo service nginx start
ubuntu@ip-172-31-18-133:~$
ubuntu@ip-172-31-18-133:~$ sudo service nginx status
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2020-10-21 06:09:30 UTC; 2min 38s ago
     Docs: man:nginx(8)
 Main PID: 14051 (nginx)
    Tasks: 2 (limit: 1140)
   CGroup: /system.slice/nginx.service
           ├─14051 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
           └─14055 nginx: worker process

Oct 21 06:09:30 ip-172-31-18-133 systemd[1]: Starting A high performance web server and a reverse proxy server...
Oct 21 06:09:30 ip-172-31-18-133 systemd[1]: nginx.service: Failed to parse PID from file /run/nginx.pid: Invalid argument
Oct 21 06:09:30 ip-172-31-18-133 systemd[1]: Started A high performance web server and a reverse proxy server.
ubuntu@ip-172-31-18-133:~$
```

Figure 31 Started Nginx Service

16. Transferred Source file into Server

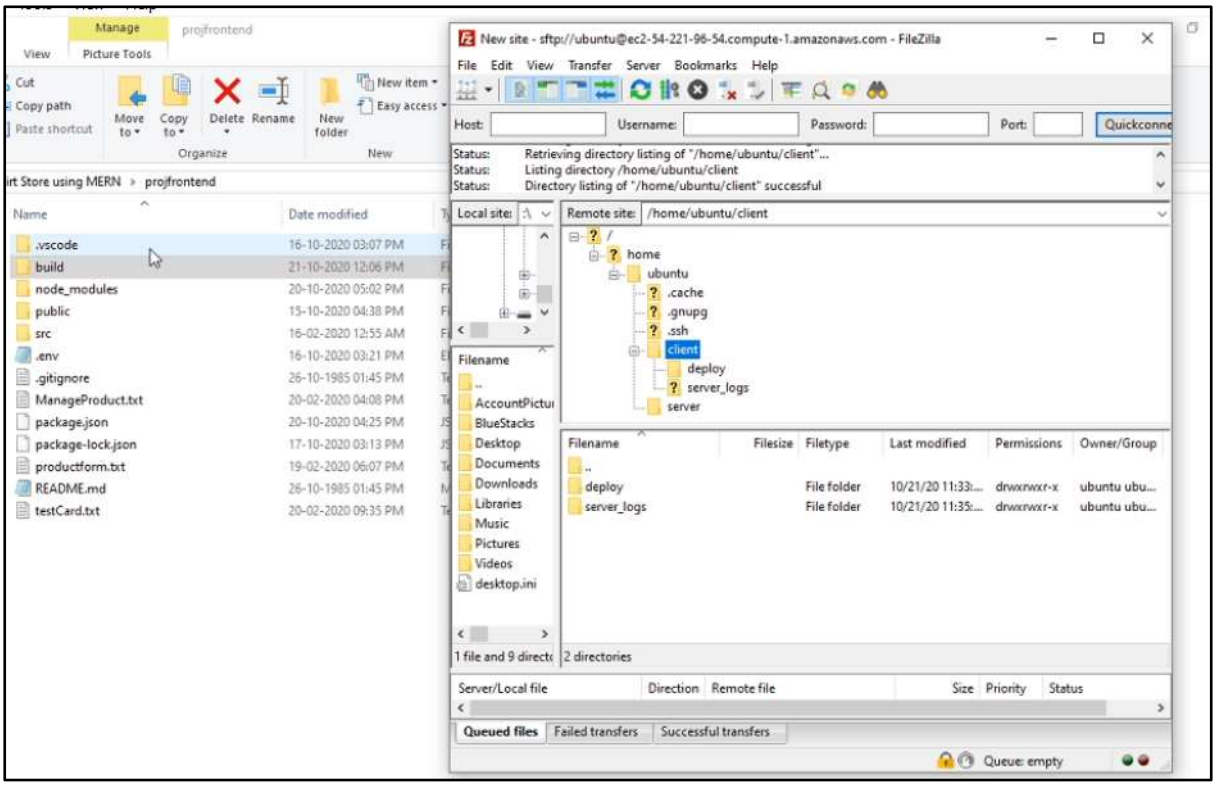


Figure 32 Copy source file into server

17. Installing MongoDb

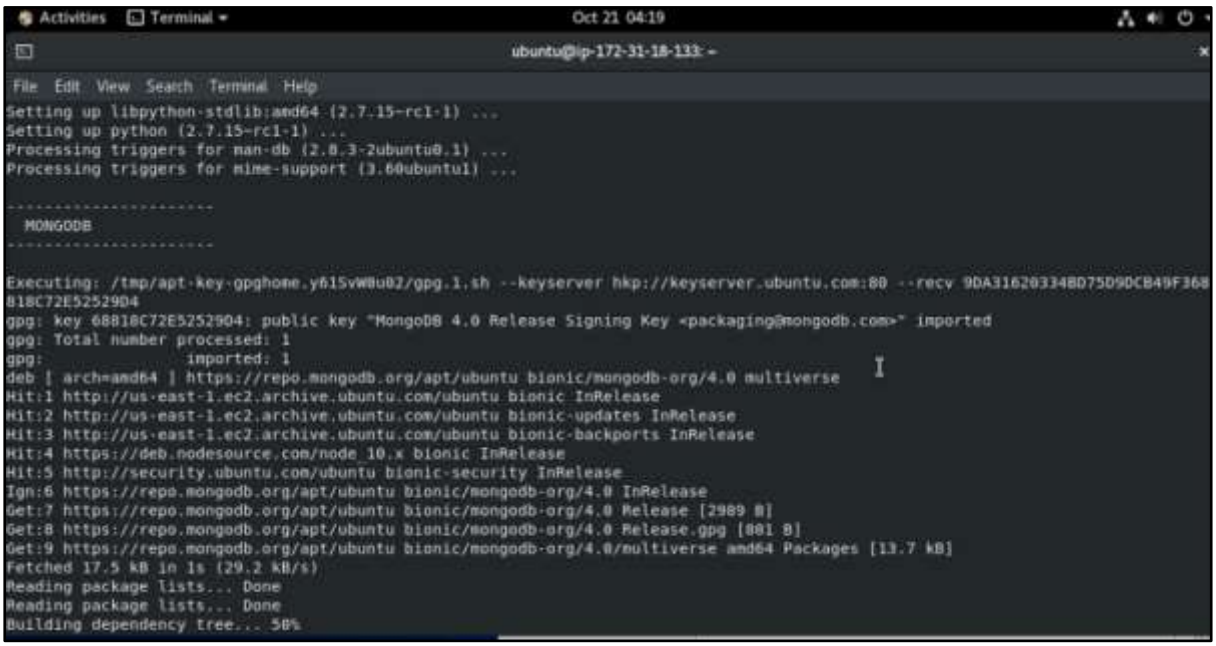


Figure 33 MongoDb Installation

18. Path directed to Backend Server

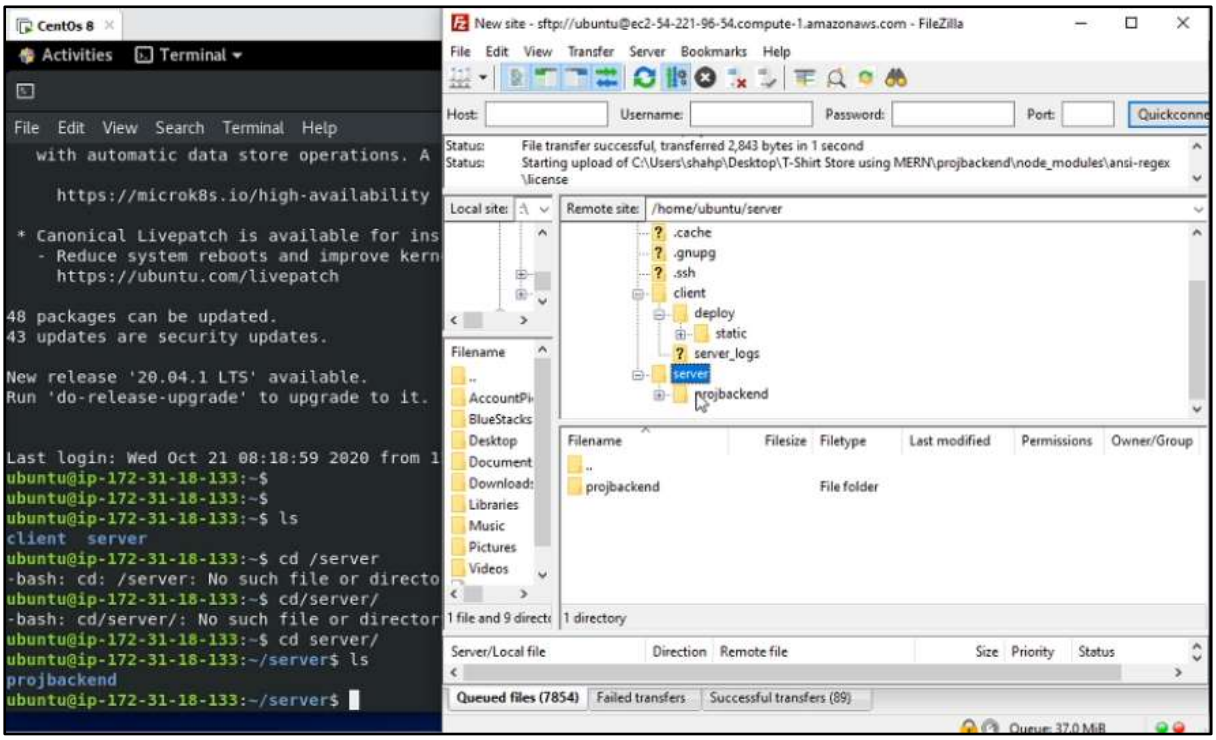
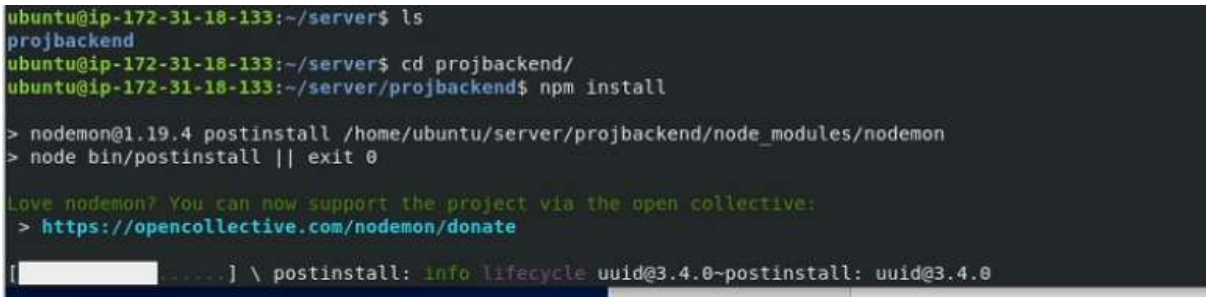


Figure 34 Path Navigation

19. Installing Node Packet Manager



20. Database Connected

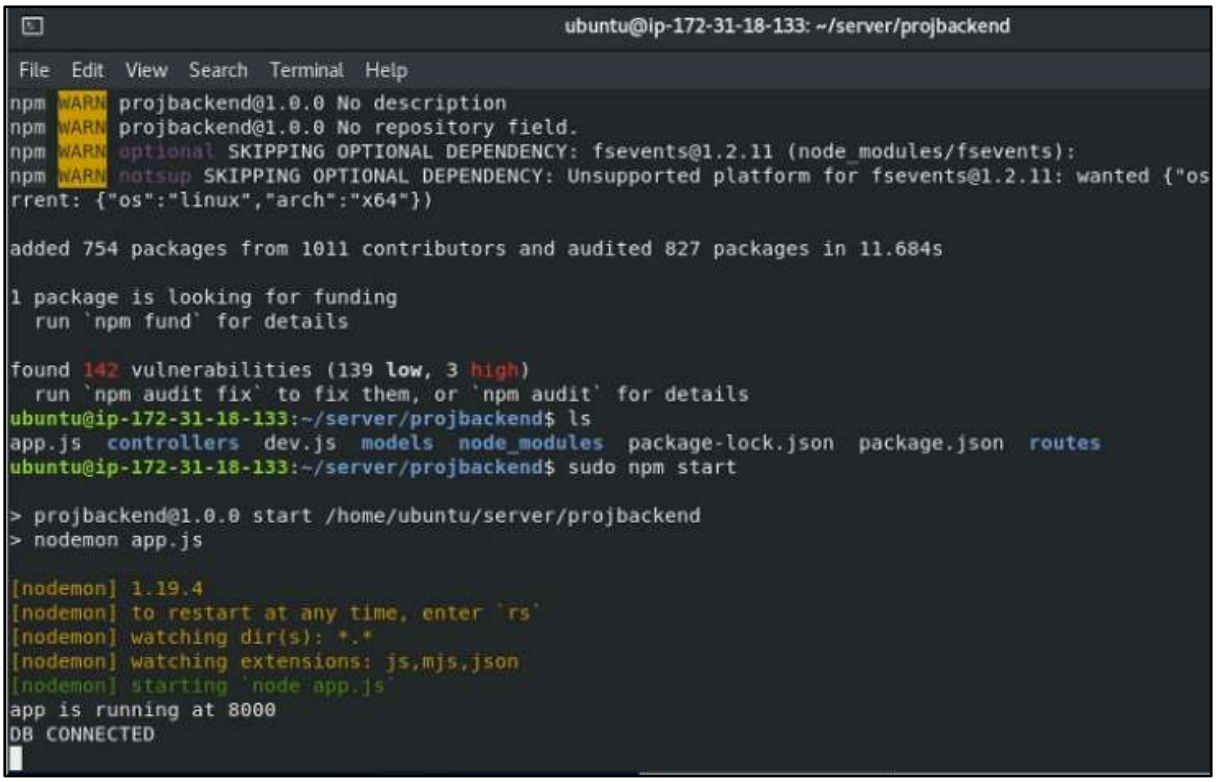


Figure 35 Database Connection

21. Web Application is Live through AWS DNS Hostname

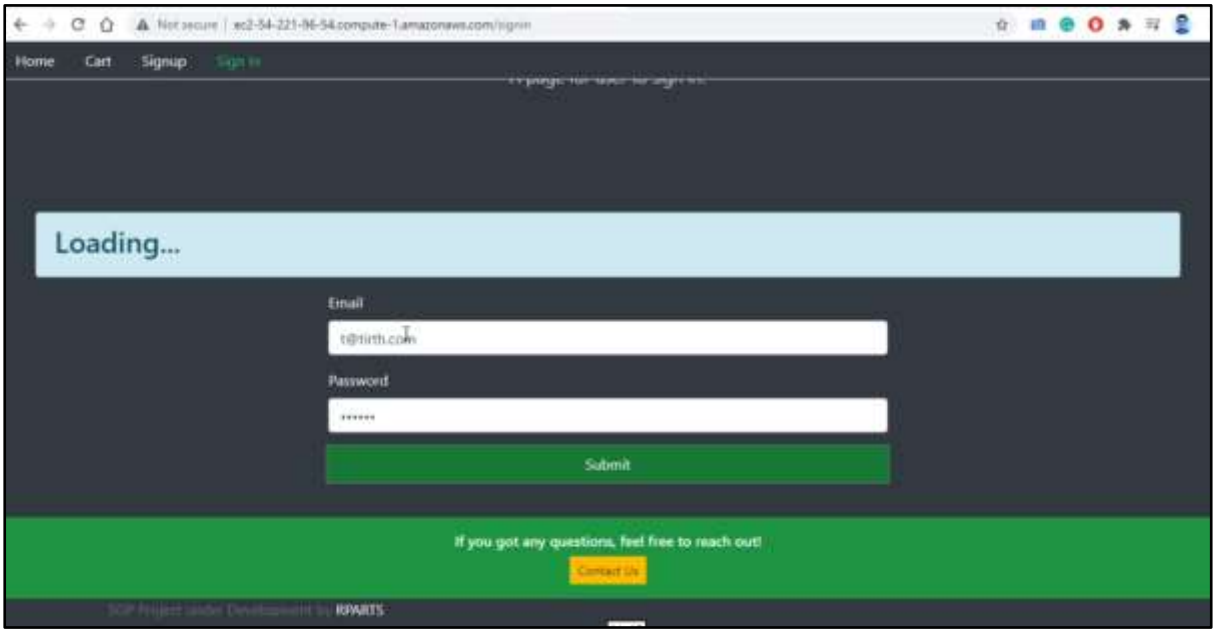


Figure 36 Deployment Successful

22. Home Page

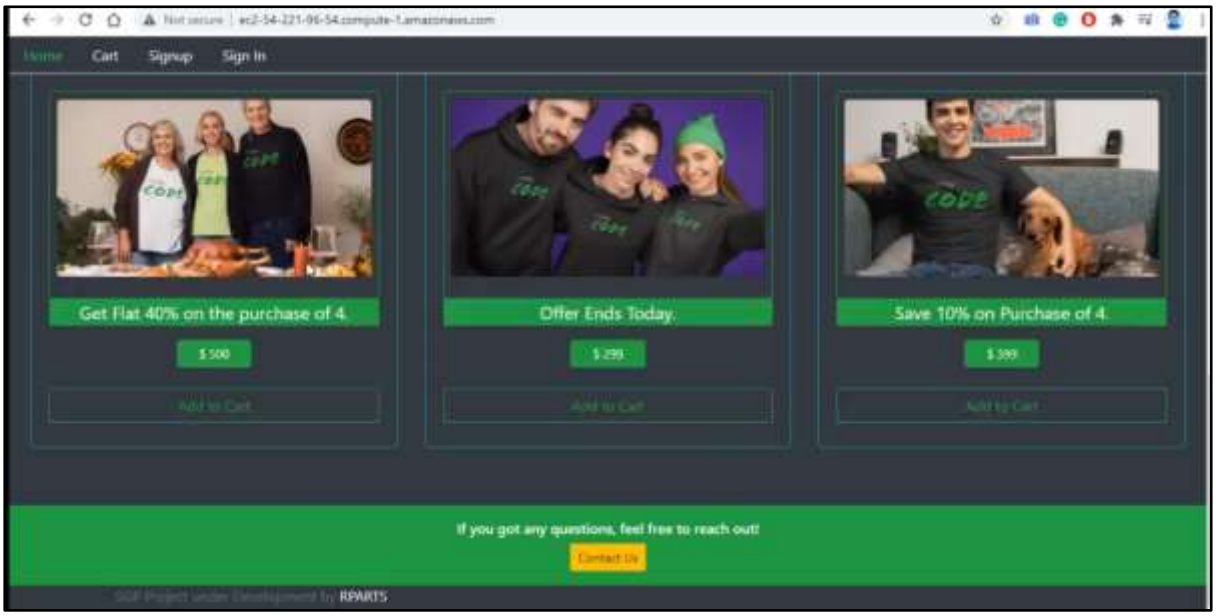


Figure 37 Live Web Application running in AWS

4.5 Editor and Tools

4.5.1 Software Requirements

Software	Description
VSCode	Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.
Postman	Postman is a collaboration platform for API development. Postman's features simplify each step of building an API and streamline collaboration so you can create better APIs—faster.
Robo3T	Robo 3T (formerly Robomongo) is a popular desktop graphical user interface (GUI) for your MongoDB hosting deployments that allows you to interact with your data through visual indicators instead of a text-based interface.

4.5.2 Screenshot

1. **VS Code**

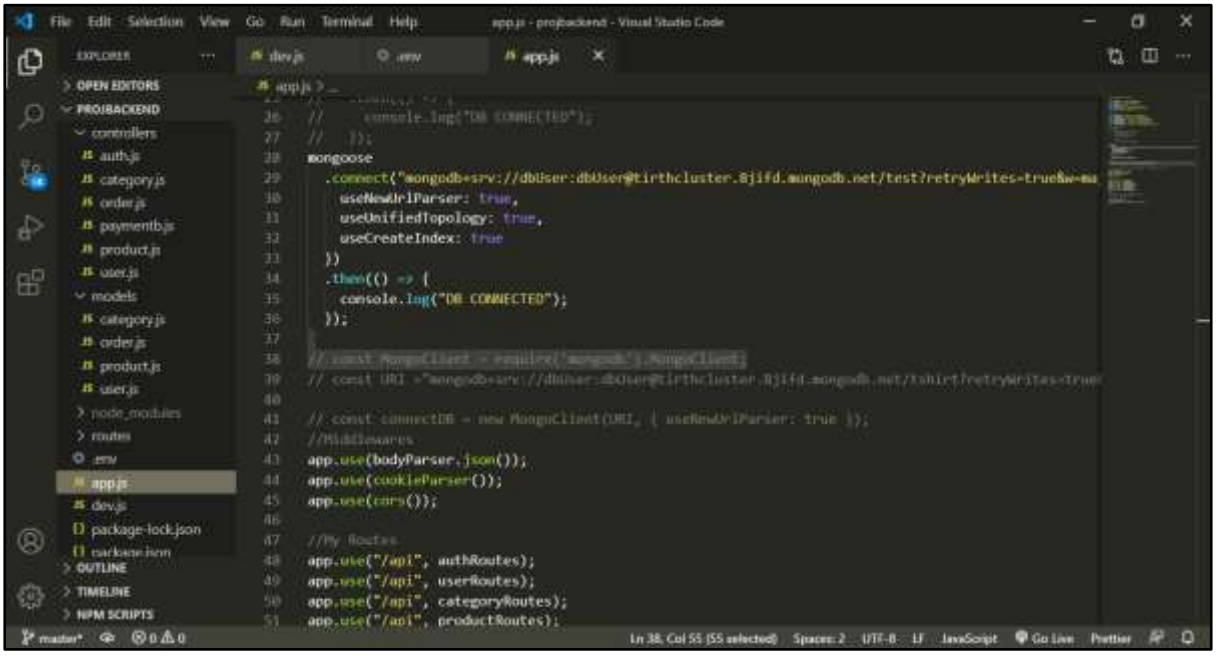


Figure 38 Visual Studio Code

2. Postman

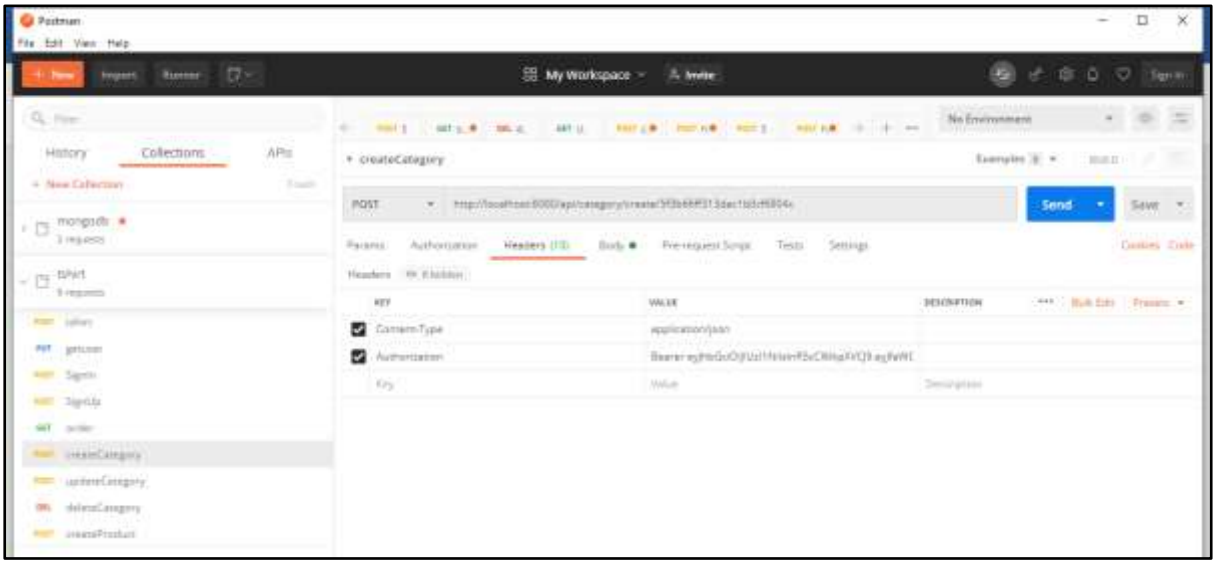


Figure 39 Postman-API Testing

3. Robo3T

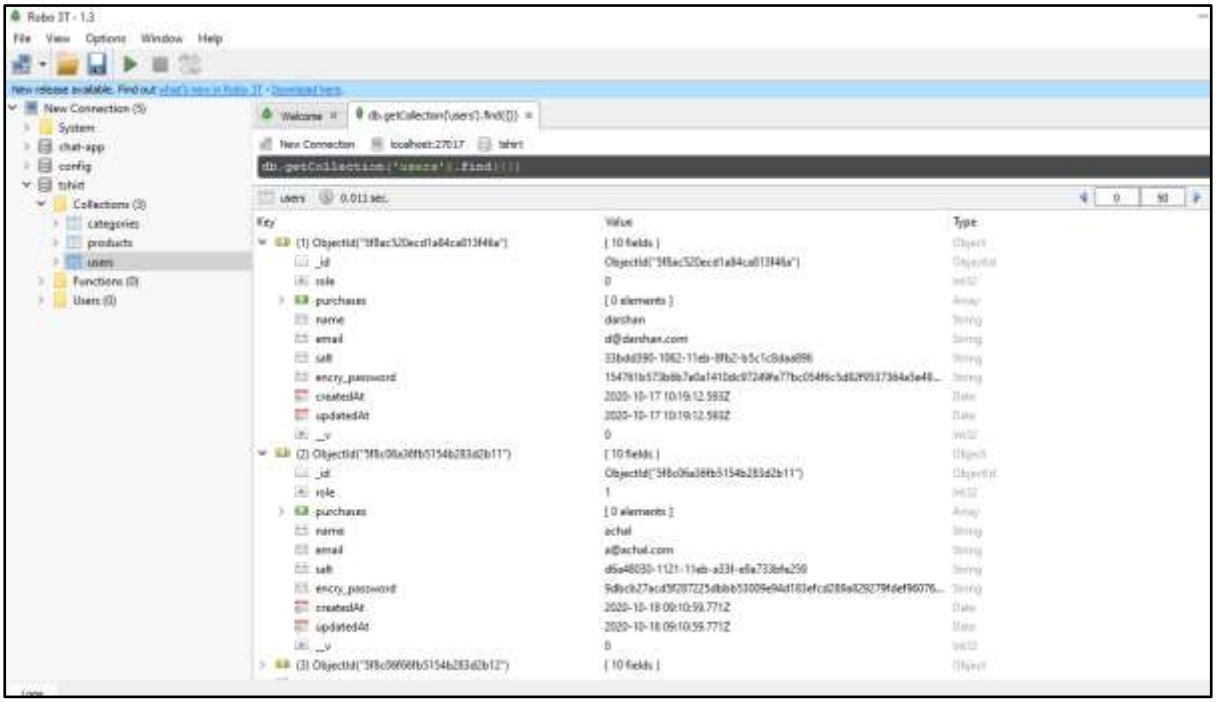


Figure 40 Robo3T-MongoDb

4.6 Payment Gateway

4.6.1 Software Requirements

Software	Description
Paypal	PayPal Holdings, Inc. is a company operating a worldwide online payments system that supports online money transfers and serves as an electronic alternative to traditional paper methods like checks and money orders.

4.6.2 Screenshot

1. **BrainTree Admin Panel for Payment Information**

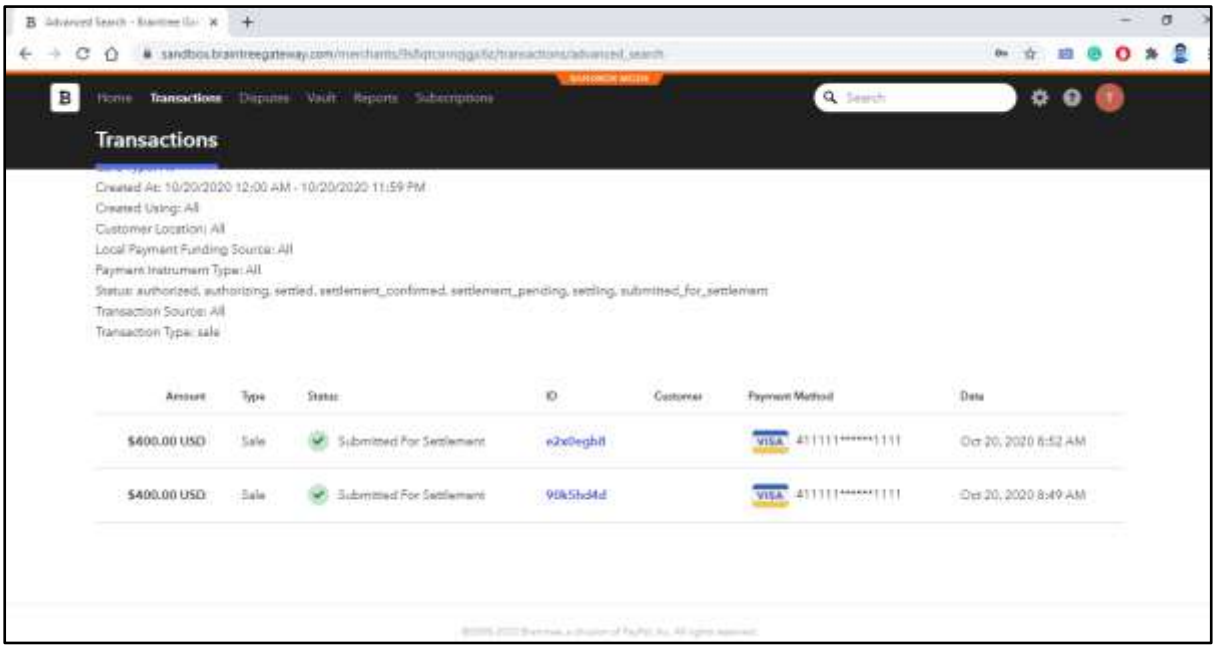


Figure 41 BrainTree Dashboard for Administarator

2. Backend Process

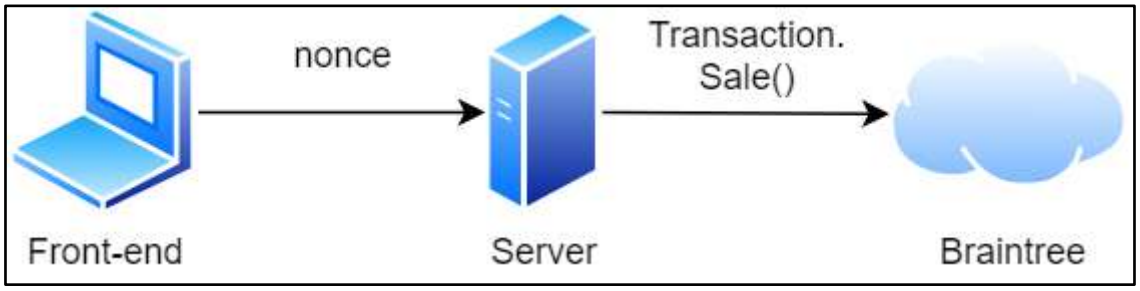


Figure 42 Integration of Payment Gateway

3. Checkout Page

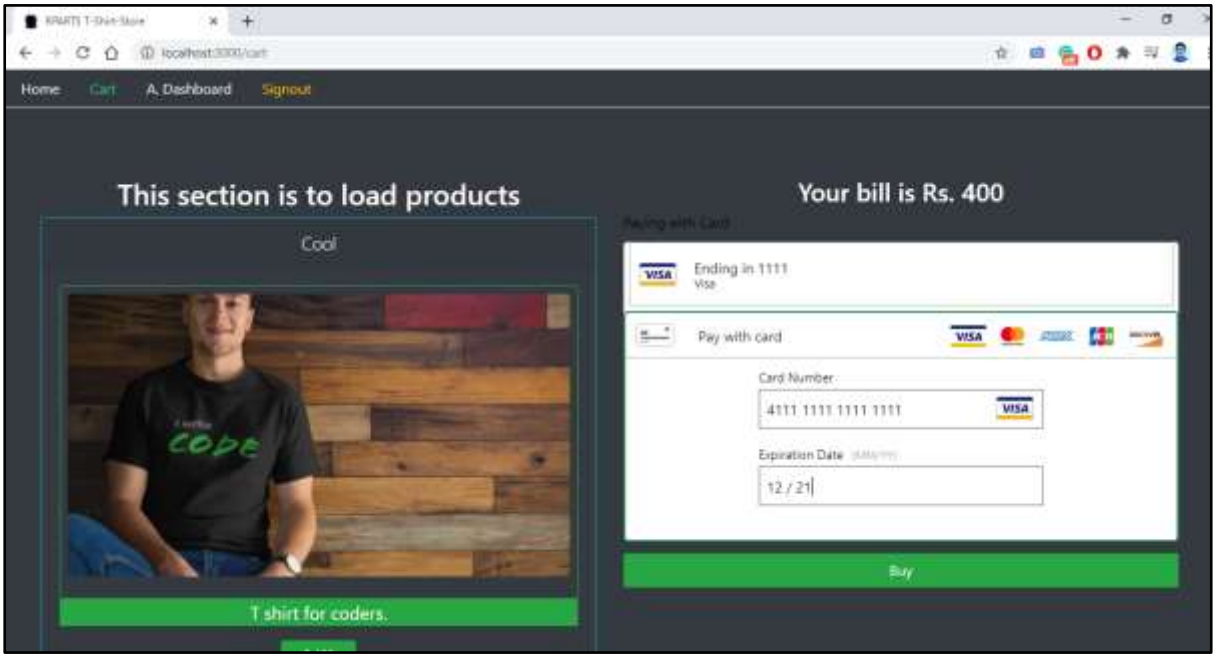


Figure 43 Checkout Page

5. Major Functionality

Following are the major functionality of our online fashion store with two module:

1. User
    a. Signup/SignIn
    b. Password Encrytion
    c. As user sign in, token is generated and stored in local memory to identify that he/she is user. The value of token is 1 for user.
    d. User can go through the products
    e. Add products to cart
    f. Pay for the products through Paypal
2. Admin
    a. Signup/SignIn
    b. Password Encrytion
    c. As admin sign in, token is generated and stored in local memory to identify that he/she is admin. The value of token is 0 for admin.
    d. Admin can create category/manage category
    e. Admin can create products/manage products
    f. Admin can manage stocks of products
    g. Admin can manage the order generated and order fulfilled

6. Web Application Flow Diagram

6.6.1 Flow Chart

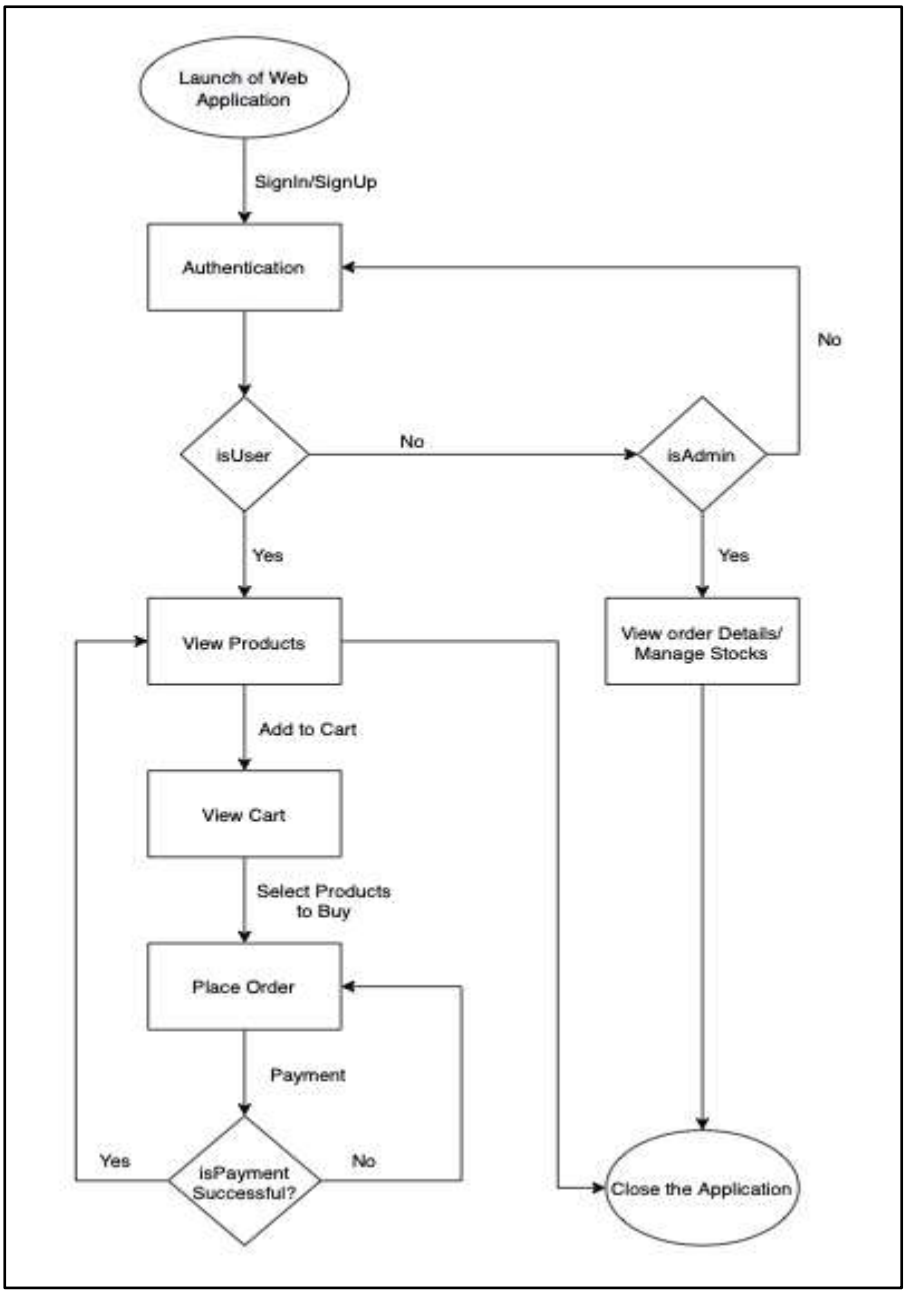


Figure 44 Work Flow of Web Application

6.6.2 Fishbone Diagram

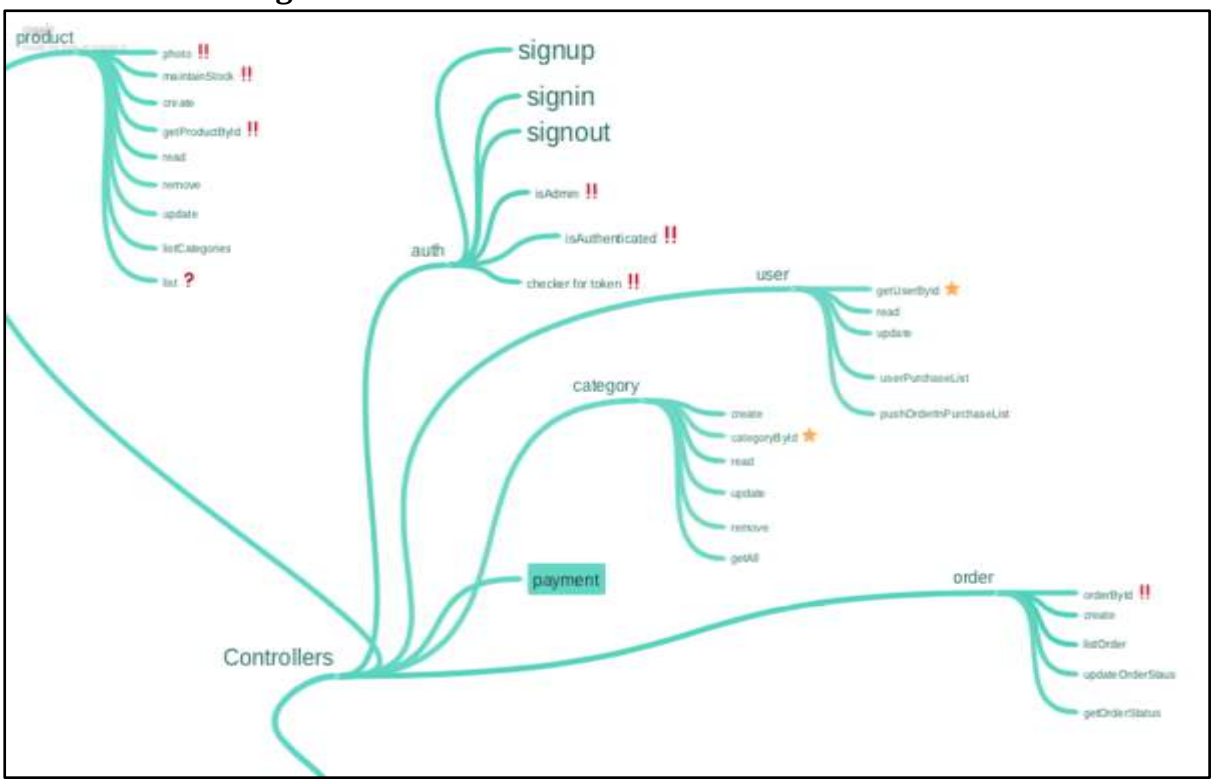


Figure 45 Controllers Fishbone Diagram

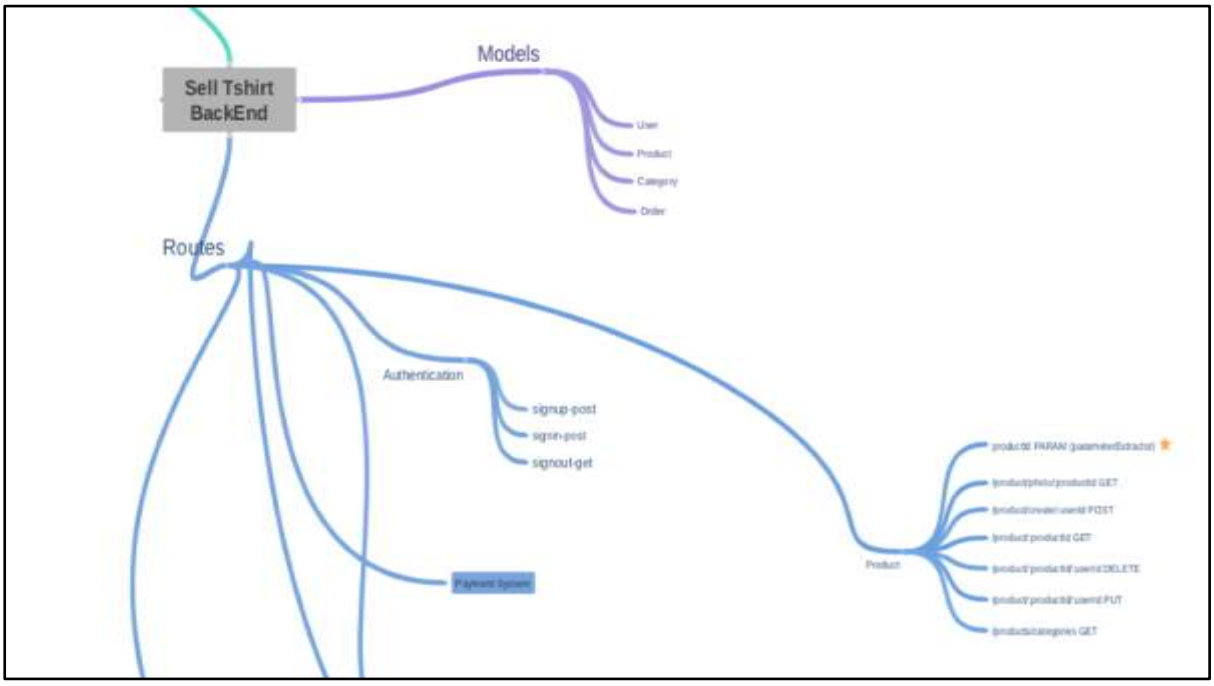


Figure 46 Models and Routes Fishbone Diagram

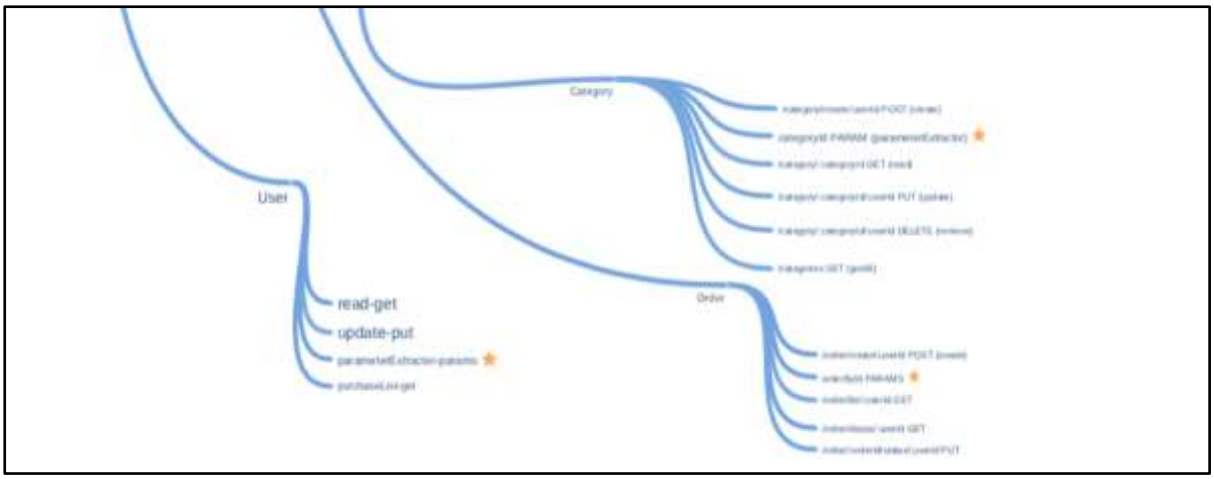


Figure 47 Detailed Model Fishbone Diagram

7. Project Outcomes

- Build a full stack e-commerce app with React, Redux, Node, Express & MongoDB
- Learned to build Admin and User Dashboard
- Integrate React with an Express backend in an elegant way
- Learned to Implement Payment Gateway using Paypal.
- Learned to Build Scalable React App with Proper Layouts and Routes.
- Learned to Store Sold Products Record into the Database for Further Processing and manage the stock.
- Learned to implement Order Management System by Admin.
- Learned to deploy the web application on AWS.

8. Future Enhancements

- Implement Advance Searching/Filtering based on Price Range
- use Cloudflare's free SSL to secure your app
- Deploy your app to Digital Ocean's Cloud Servers
- Cloudflare's CDN to serve your app

9. References

- https://nodejs.org/docs/latest-v13.x/api/
- https://www.npmjs.com/package/package
- https://reactjs.org/docs/getting-started.html
- https://docs.mongodb.com/manual/
- https://www.tutorialspoint.com/mongodb/index.htm
- https://expressjs.com/en/guide/routing.html
- https://developer.mozilla.org/en-US/docs/Web/JavaScript