

# **Internship Report at Paxcom India Pvt Ltd**

Report submitted in partial fulfillment of the requirement for the degree  
of Bachelor of Technology

in

**Information Technology**

By

Sajal Lahoti (161461)

Under the supervision of

Mr Bobin Sondhi

To

Dr. Aman Sharma



Department of Computer Science & Engineering and Information  
Technology

## **ACKNOWLEDGEMENT**

The successful accomplishment of this internship is the outcome of contribution of a number of people to whom we are grateful and thanked them from the very deep of my heart.

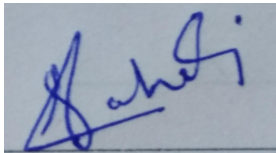
I would like to convey my sincere gratitude to Mr. Bobin Sondhi, my Mentor and Manager at Paxcom India. Without his guidance and assistance this internship would not be possible.

I am greatly indebted to my project supervisor Dr Aman Sharma, without his guidance this report would not have been possible.

I would also like to thank Mr Pankaj Kumar, training and placement coordinator, Jaypee University of Information Technology.

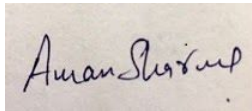
## STUDENT DECLARATION

I, Sajal Lahoti, hereby declare that the presented report of internship titled “Internship Report at Paxcom India Pvt Ltd” is uniquely prepared by me after the completion of 12 weeks work at Gurugram Office of Paxcom India pvt Ltd. I also confirm that the report is only prepared for my academic requirement, not for any other purpose. It might not be used with the interest of the opposite party of the corporation.

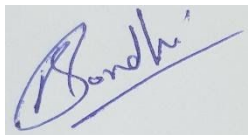


(Student Signature)  
Sajal Lahoti, 161461

This is to certify that the above statement made by the candidate is true to the best of my knowledge.



(Project Guide)  
Dr Aman Sharma



(Supervisor Signature)  
Mr Bobin Sondhi  
Paxcom India

# INDEX

<b>1. INTRODUCTION</b>	<b>3</b>
<b>2. DESCRIPTION OF THE INDUSTRY</b>	<b>4</b>
<b>3. LITERATURE SURVEY AND SYSTEM DEVELOPMENT</b>	<b>6</b>
3.1. Tools and Technologies used	6
3.2. Angular	6
3.3. Github	7
3.4. Typescript	9
3.5. Angular Material Design	10
<b>4. WORKFLOW AND PERFORMANCE ANALYSIS</b>	<b>11</b>
4.1. Database	11
4.2. Login/Register User Module	16
4.3. OAuth component	16
4.4. View Module	18
4.5. Cart/Order Module	21
<b>5. REFERENCES</b>	<b>25</b>

## LIST OF FIGURES

<b><u>Number</u></b>	<b><u>Figure Title</u></b>	<b><u>Page Number</u></b>
3.1	Github Workflow	8
4.1	Json-Server window	12
4.2	Rest API for user database	13
4.3	Rest API for product database	13
4.4	Rest API for order database	14
4.5	Rest API for cart database	14
4.6	Login page for application	16
4.7	OAuth component for login via google	17
4.8	Form for registering a user	17
4.9	View Product component	18
4.10	Popup component for product	19
4.11	Snackbar after adding product to cart	19
4.12	Cart component with save for later and pagination	20
4.13	Choosing address on delivery address page	21
4.14	Add address page	21
4.15	Billing address page	22
4.16	Review Order page	24
4.17	Your Order Page	24

# Chapter-1

## INTRODUCTION

Paxcom is an Indian based software company providing complete Ecommerce solutions from creating and managing stores (or markets) to advanced data analytics, exponential revenue growth management, warehouse management and supply chain management along with ensuring satisfied customers.

I worked at Paxcom India pvt ltd at Gurugram branch which is the head office of the company in India. I worked with the Omnichannel team and was successfully able to understand their working structure and pattern. I was trained in various technologies like Angular, C#, ASP.net, Typescript, Github, Javascript etc. I also learned soft skills like communicating within a corporate firm and working with a team. I was successfully able to understand various coding paradigms used by a company to build its application. I got a thorough understanding of some of the company's existing products and some of the upcoming products.

The working culture of the company is great. I thoroughly enjoyed myself working there. Paxcom has a typical blend of work and fun. Although I didn't get to spend much time in the company office and started working from home after the coronavirus pandemic lockdown, I really enjoyed the weekend football and various parties at the company. And even after the lockdown the communication between me and my team was good through various meeting platforms like skype and google meets.

## Chapter-2

### DESCRIPTION OF THE INDUSTRY

Paxcom is an analytics driven organization established by Punit Sindhwani. It is a gathering of innovation devotees and internet business specialists cooperating to make reasonable and exceptional online business items and administrations.

Paxcom makes it simpler for both the brand proprietors and dealers to ease every day activities with information driven examination and bits of knowledge concentrating on improving the business, item content/perceivability and limited time methodologies.

Paxcom's point is to be your confided in join forces with the intensity of many assets in it – your exploration examiner, your substance pro, your advancement administrator, your web based business investigator, your information researcher and your master specialist.

The activity of the organization is to give helpful experiences to our customers which will permit them to follow their opposition, screen their items, advance their substance across commercial centers, increment perceivability, support their deals and some more.

Few years ago Paxcom was acquired by Paymentus group and it's name was changed from "Paxcom" to "Paxcom India - A Paymentus Company".

In 2004, Paymentus was conceived from a longing to improve the manner in which bills get paid. Vision, development and commendable help have moved Paymentus to turn into the main paperless electronic charging and installment arrangement available, bringing about 1,500 customers remembering probably the biggest billers for North America.

We realize that so as to keep our answers present and pertinent, we need individuals with the expertise, drive and proclivity for encouraging a remarkably glad client experience. Our profoundly dedicated, imaginative representatives transformed a thought into a protected, SAAS-based Customer Engagement and Payment Platform; one that empowers direct-charge

associations to give a brought together client experience and lift selection of cost-sparing electronic charging and installment administrations.

Perceived by Deloitte to be among the quickest developing North American organizations in 2011, 2013, 2014, and 2016, Paymentus reliably endeavors to grow better, quicker, progressively secure, cost-effective charging and installment stages. We persistently look for higher incentive for our clients, in the two arrangements and administration.

It's what prompted our striking 40% year-over-year installment exchanges development in the most recent decade. We succeed when our customers succeed. They succeed when their client connections are upgraded and, thusly, their clients take an interest in these cost-sparing electronic administrations at high rates.



## Chapter-3

# LITERATURE SURVEY AND SYSTEM DEVELOPMENT

### Tools and Technologies used

My Internship at Paxcom involved learning skills in languages like Angular, C#, ASP.NET, HTML, CSS, Typescript, Github and Javascript. Apart from that this internship has also taught me soft skills like working culture in a corporate firm, communication within a team.

### **Angular**

One of the center contentions to pick angular over other single page application systems is its plainly characterized method of how things should be finished. It has an assessment, in a manner of speaking. That characterized way empowers the making of a uniform code base over an association, without agonizing over characterizing certain principles. Angular as of now accompanies these principles out of the crate.

Tailing them doesn't just build the consistency and hence the nature of the code. It makes your application more obvious, as well. That is, in the event that you are as of now acquainted with angular.

This exacting methodology likewise proves to be useful across collaboration fringes. It empowers new engineers to incorporate into another group rapidly, as a result of the high commonality with the code.

What I need to state is, you should follow the angular structure rules to capitalize on the angular system. It will make your life significantly simpler, when coming into new undertakings and will expand the nature of your code consequently.

A typical activity, when learning angular is to put everything into the application module. Obviously, that your application will advance into a total chaos. Modules are there which is as it should be!

Modules help to sort out your code into littler groups to make discovering things simpler. Yet, they are not just a corrective thing. With the assistance of lethargic stacked modules, you can likewise build the client experience by just downloading the pieces of the application, that are required at that point.

## **Github**

Git is utilized for dealing with the progressions to a task after some time. A task may be only a solitary record, a bunch of documents, or a large number of documents. Those records can be anything from plain content to pictures or recordings.

Since Git is centered around overseeing transforms, it is regularly utilized as a joint effort instrument permitting individuals to take a shot at a similar undertaking simultaneously. By following their individual changes, Git can unite everything to the last form.

Envision that you're composing a blog entry that has numerous records related with it. You may have one principle content record which is the real post, an extra document for references, just as some different records that are graphs and different pictures.

Without Git, you may have these different records put away in an organizer on your PC, however it is extremely unlikely to tell where the entirety of the documents are at a given point in time. Envision you send your post to two companions to duplicate alter. How would you consolidate their progressions back? Which record is the first, which is altered? With Git you can follow the entire repo at different focuses in time utilizing submits while likewise giving comments on why you chose to spare the undertaking by then. Afterward, you can per use this history of resolves to see an away from of your undertaking, and furthermore travel back in that history, if important.

A software codebase works simply like that blog entry. At its most essential level, it is an assortment of records that are connected to each other. At the point when an engineer is taking a

shot at a specific element, Git gives an approach to spare a depiction of the whole repo by means of a submit. This is generally done when gradual advancement has been made and an element is without bug. In making the submit, the designer can give comments clarifying what was changed and why it was changed. This message adds to the historical backdrop of the extend and can make it simple to decide when a specific element – or even a bug – was presented.

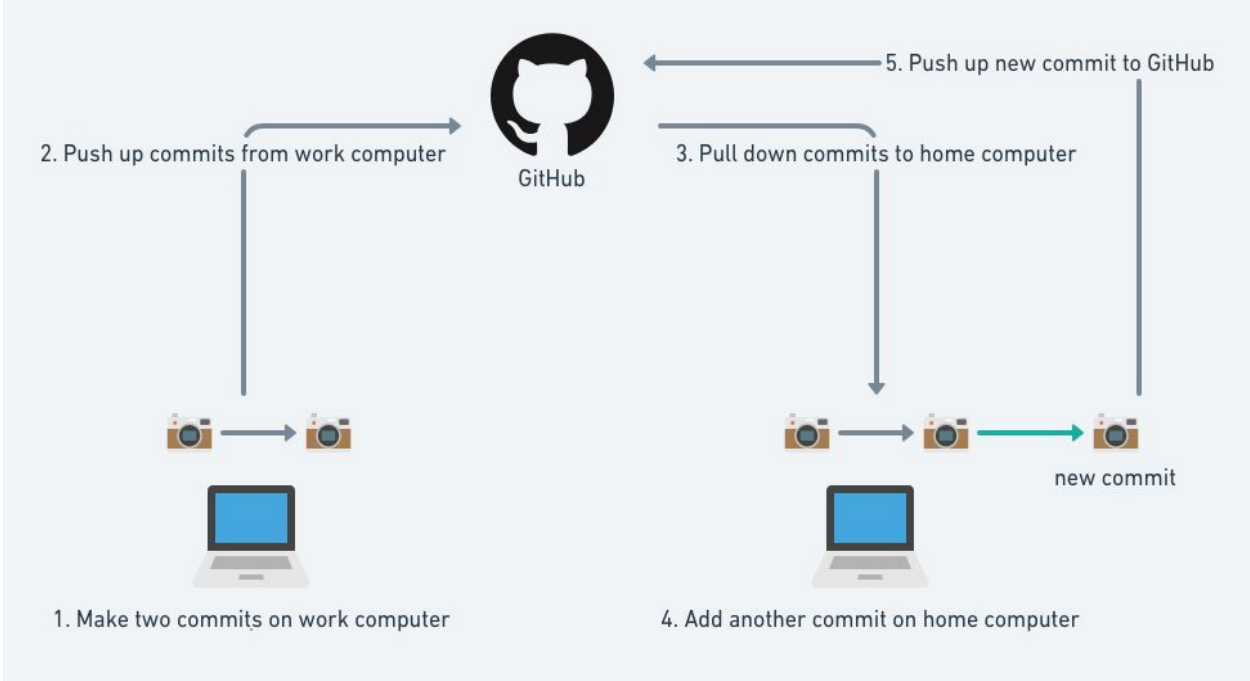


Fig 3.1 Github Workflow

## **Typescript**

The definition from the official site says: "a composed superset of JavaScript" yet it accept you comprehend what a "superset" is and what "composed" signifies. Rather to keep things straightforward you can consider TypeScript of "a layer on top" of JavaScript.

TypeScript is a layer since you can compose TypeScript code in your proofreader. After a gathering all that TypeScript stuff is gone and you're left with plain, straightforward JavaScript.

On the off chance that the possibility of an assemblage step confounds you remember that JavaScript is as of now accumulated and afterward deciphered. There is a JavaScript motor that peruses and executes your code.

Be that as it may, JavaScript motors can't peruse TypeScript code so any TypeScript record ought to go under a "pre-interpretation" process, called aggregation. Simply after the principal assemblage step you're left with unadulterated JavaScript code, prepared to run in a program. TypeScript is an exceptional sort of JavaScript yet it needs an "interpreter" before running in a program.

TypeScript begins from a similar punctuation and semantics that a large number of JavaScript designers know today. Utilize existing JavaScript code, join mainstream JavaScript libraries, and call TypeScript code from JavaScript. TypeScript arranges to spotless, basic JavaScript code which runs on any program, in Node.js, or in any JavaScript motor that underpins ECMAScript 3 (or more current).

## Angular Material Design

Material Design (codenamed Quantum Paper) is a visual language that can be utilized to make computerized encounters. It's a lot of standards and rules across stages and gadgets for intuitiveness, movement and segments that streamline the plan work process for groups structuring their item.

The Material parts permit you to make proficient UIs with ground-breaking measured quality, theming and customization highlights.

Angular Material is the execution of Material Design standards and rules for Angular. It contains different UI segments, for example,

- structure controls (input, select, checkbox, date picker and sliders and so on.),
- route designs (menus, sidenav and toolbar)
- format segments (matrices, cards, tabs and records )
- catches
- pointers (progress bars and spinners)
- popups and modals
- information tables with headers and pagination and so forth.

## Chapter-4

### WORKFLOW AND PERFORMANCE ANALYSIS

In my Internship, I worked on a team project and built an application involving basic E-Commerce operations like login/logout of user/admin, registration of user, browsing products and sorting them based on customer's choice of category of product, features like add to cart and save for later (wishlist). The application also generated a final invoice of customer on checkout on the delivery address selected by user.

#### Database

For database for users, products and carts, I used a json file to create objects storing data and with the use of json-server converted the data of the json file to a rest api. All the post, get, put, delete operations in the Angular application were performed on the REST API.

REST is an abbreviation for REpresentational State Transfer. It is a compositional style for disseminated hypermedia frameworks and was first introduced by Roy Fielding in 2000 in his renowned thesis. Like some other design style, REST additionally has it's own 6 controlling requirements which must be fulfilled if an interface should be alluded as RESTful. These standards are recorded beneath.

- Client-server – By isolating the UI concerns from the information stockpiling concerns, we improve the movability of the UI over various stages and improve adaptability by streamlining the server components.
- Stateless – Each solicitation from client to server must contain the entirety of the information important to comprehend the solicitation, and can't exploit any put away context on the server. Session state is accordingly kept altogether on the client.
- Cacheable – Cache constraints necessitate that the information inside a response to a

solicitation be verifiably or expressly marked as cacheable or non-cacheable. On the off chance that a response is cacheable, at that point a client reserve is given the option to reuse that response information for some other time, proportionate solicitations.

- Uniform interface – By applying the product building standard of all inclusive statement to the component interface, the general system design is rearranged and the perceivability of interactions is improved. So as to get a uniform interface, various building constraints are expected to direct the conduct of components. REST is characterized by four interface constraints: identification of assets; manipulation of assets through representations; self-engaging messages; and hypermedia as the motor of application state.
- Layered system – The layered system style permits a design to be made out of various leveled layers by constraining component conduct to such an extent that every component can't "see" beyond the quick layer with which they are interfacing.
- Code on demand (optional) – REST permits client functionality to be stretched out by downloading and executing code as applets or contents. This streamlines clients by diminishing the quantity of highlights required to be pre-executed.

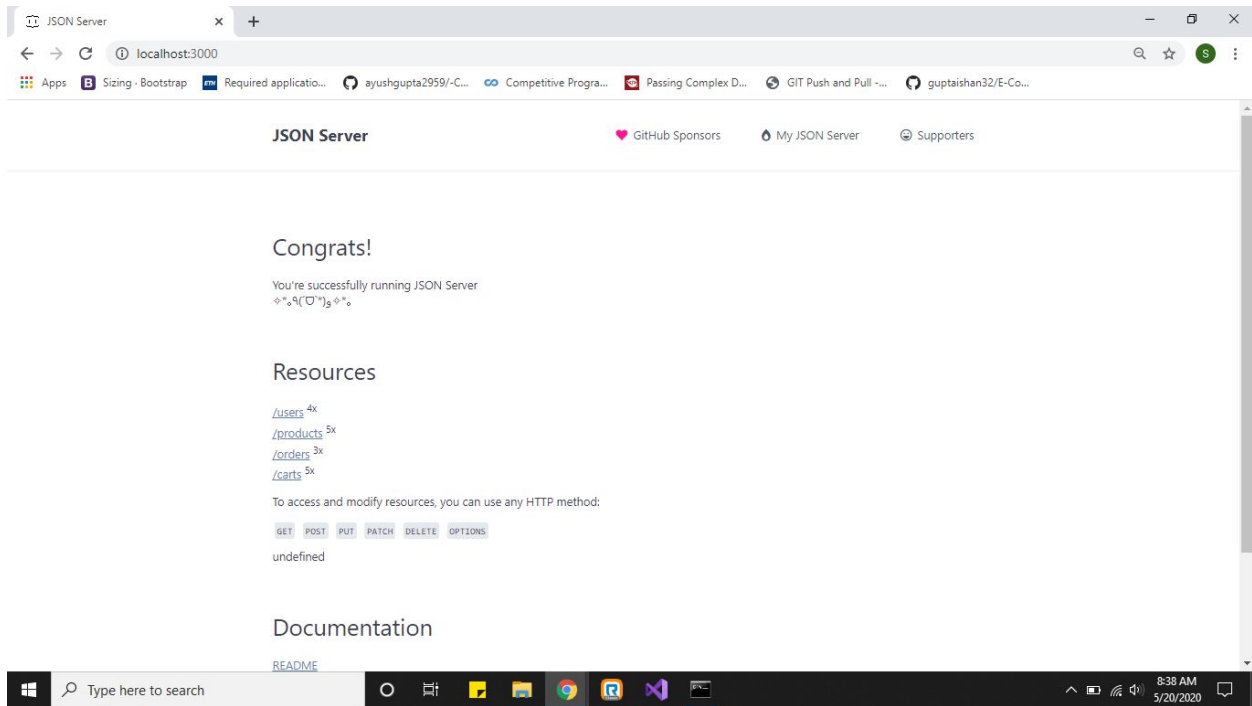


Fig 4.1 Json-Server Window

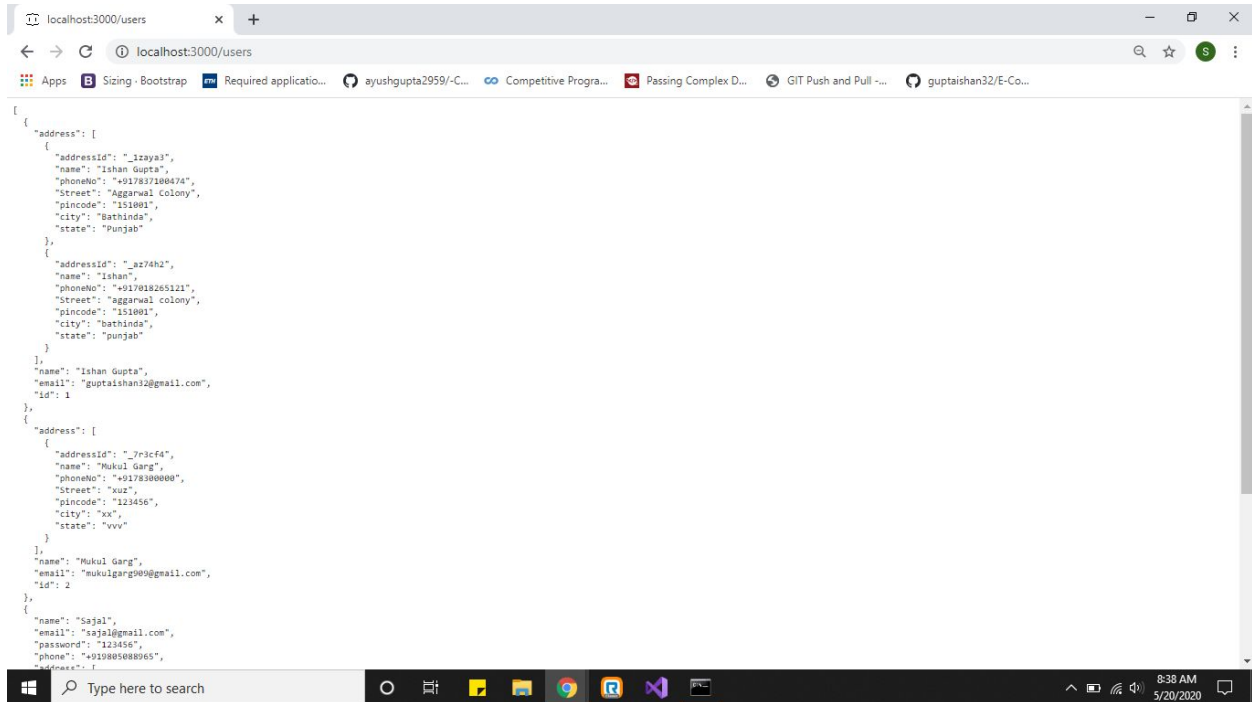


Fig 4.2 REST API for user database



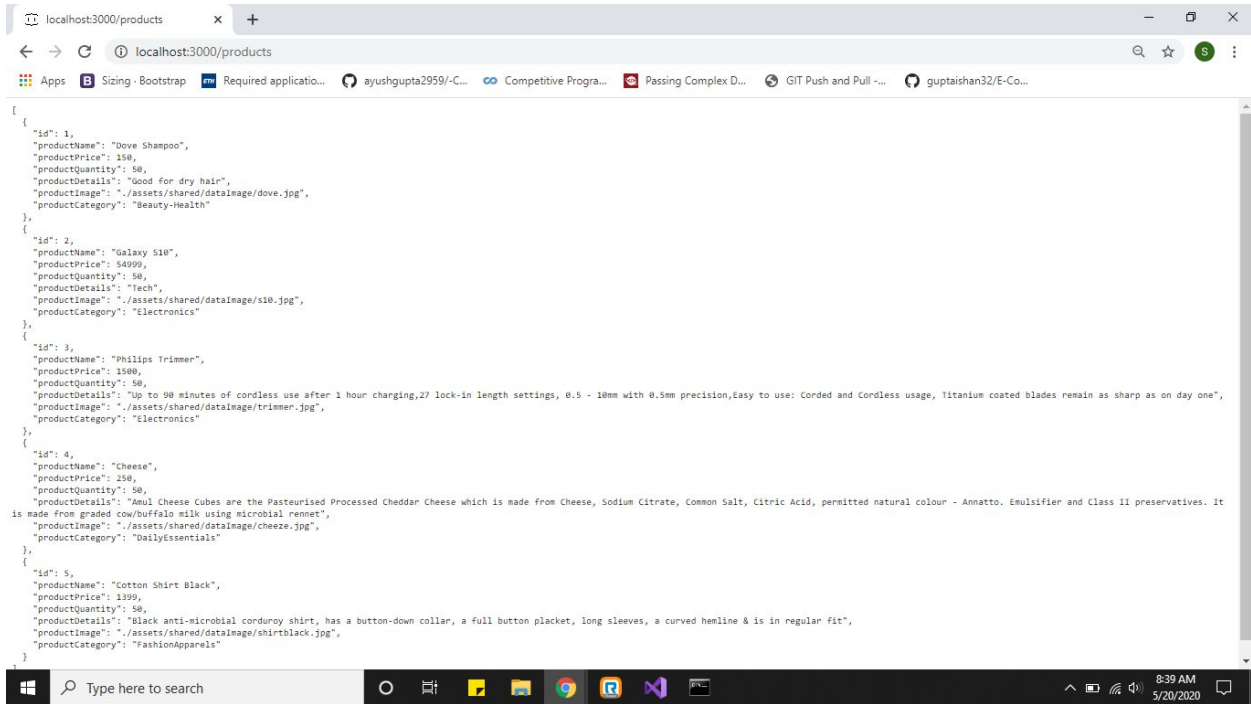


Fig 4.3 REST API for products database

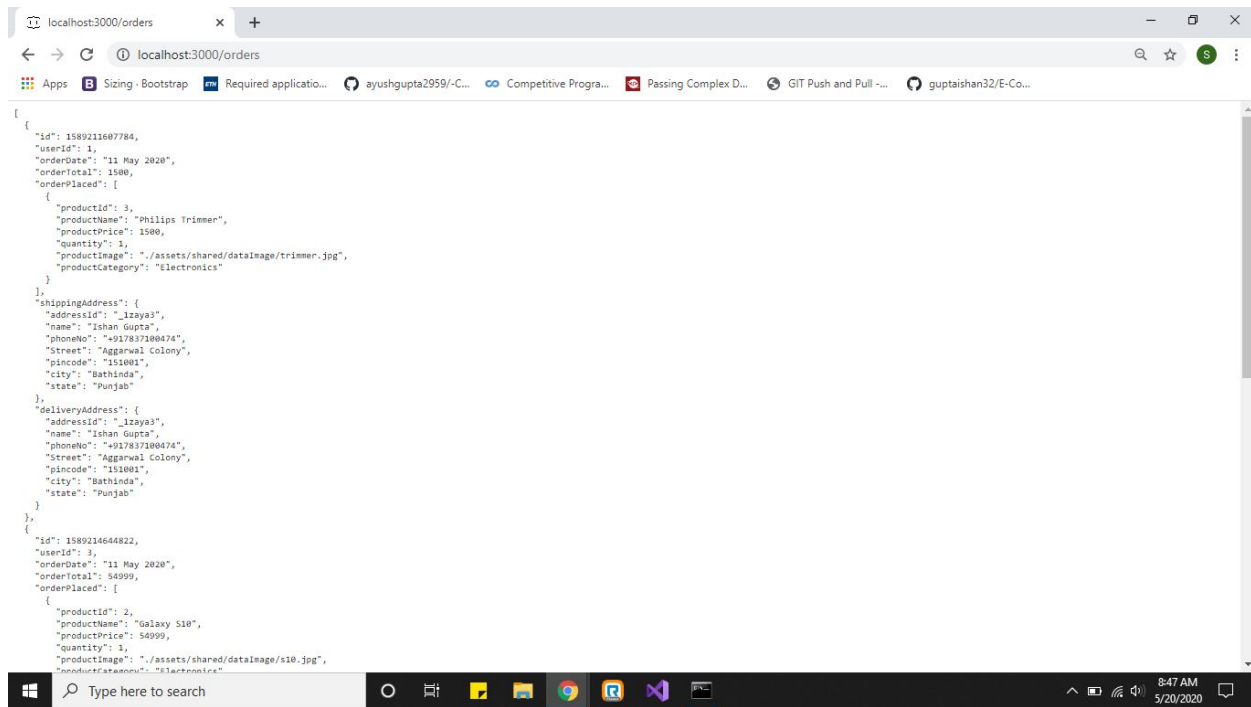


Fig 4.4 REST API for order database

```
[
  {
    "id": "1",
    "userId": "1",
    "saveForLaterItems": [
      {
        "productId": "3",
        "productName": "Phillips Trimmer",
        "productPrice": 1999,
        "quantity": 1,
        "productImage": "./assets/shared/dataImage/trimmer.jpg",
        "productCategory": "Electronics"
      }
    ],
    "cartOrder": []
  },
  {
    "id": "3",
    "userId": "3",
    "saveForLaterItems": [],
    "cartOrder": []
  },
  {
    "id": "2",
    "userId": "2",
    "saveForLaterItems": [
      {
        "productId": "1",
        "productName": "Dove Shampoo",
        "productPrice": 159,
        "quantity": 1,
        "productImage": "./assets/shared/dataImage/dove.jpg",
        "productCategory": "Beauty-Health"
      }
    ]
  },
  {
    "id": "2",
    "userId": "2",
    "saveForLaterItems": [],
    "cartOrder": []
  },
  {
    "id": "4",
    "userId": "4",
    "saveForLaterItems": []
  }
]
```

Fig 4.5 REST API for cart database

## **Login/Register User Module**

Login component involved two features that are login via google and login by validating email and password in the existing API database.

### **OAuth component**

OAuth is an open standard for get to assignment, for the most part used as a way for Internet customers to give destinations or applications access to their information on various locales anyway without giving them the passwords. This framework is used by associations, for instance, Amazon, Google, Facebook, Microsoft and Twitter to permit the customers to bestow information about their records to outcast applications or destinations.

Generally, OAuth provides for clients a "secured allotted access" to server resources in light of a legitimate concern for a benefit owner. It shows a strategy for resource owners to favor untouchable access to their server resources without sharing their accreditations. Arranged unequivocally to work with Hypertext Transfer Protocol (HTTP), OAuth essentially allows persuade to tokens to be given to pariah clients by an endorsement server, with the support of the benefit owner. The pariah by then uses the passageway token to get to the made sure about resources encouraged by the advantage server.

Google APIs use the OAuth 2.0 show for approval and endorsement. Google supports ordinary OAuth 2.0 circumstances, for instance, those for web server, client side, presented, and limited data contraption applications.

To begin, get OAuth 2.0 client accreditations from the Google API Console. By then your client application requests a passageway token from the Google Authorization Server, isolates a token from the response, and sends the token to the Google API that you have to get to. For an instinctive show of using OAuth 2.0 with Google (tallying the choice to use your own client capabilities), investigate various roads in regards to the OAuth 2.0 Playground.

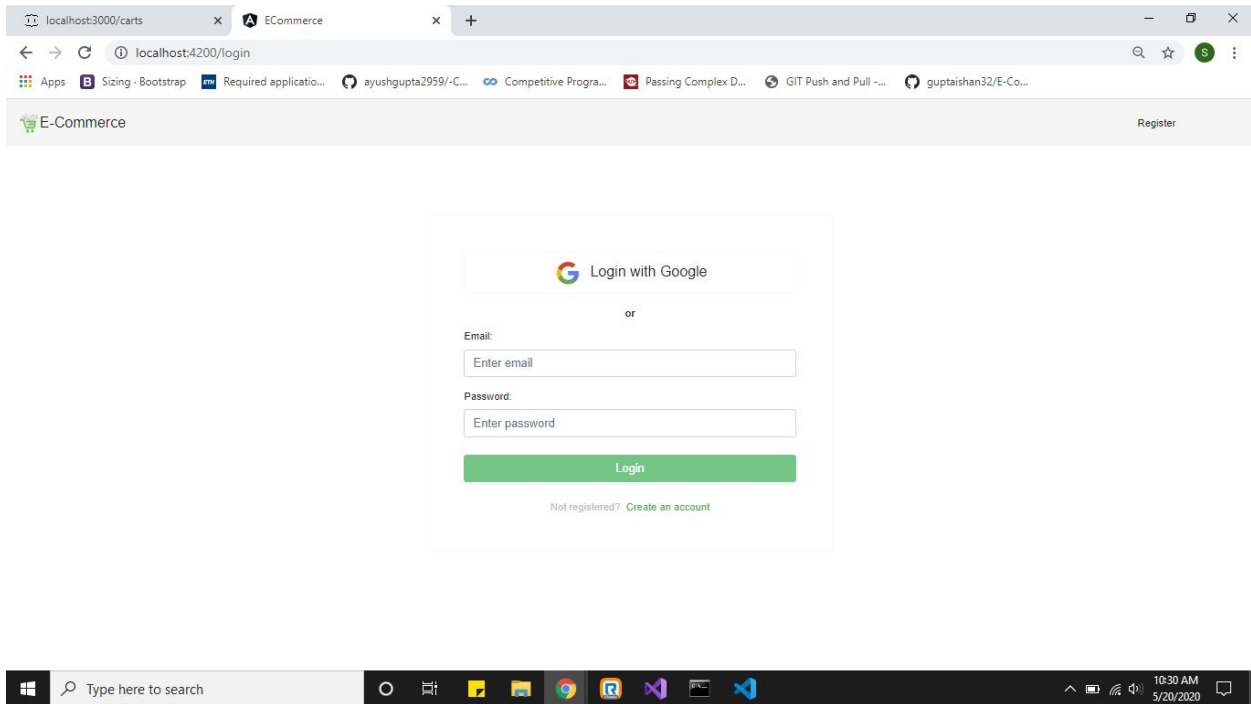


Fig 4.6 Login Page for Application

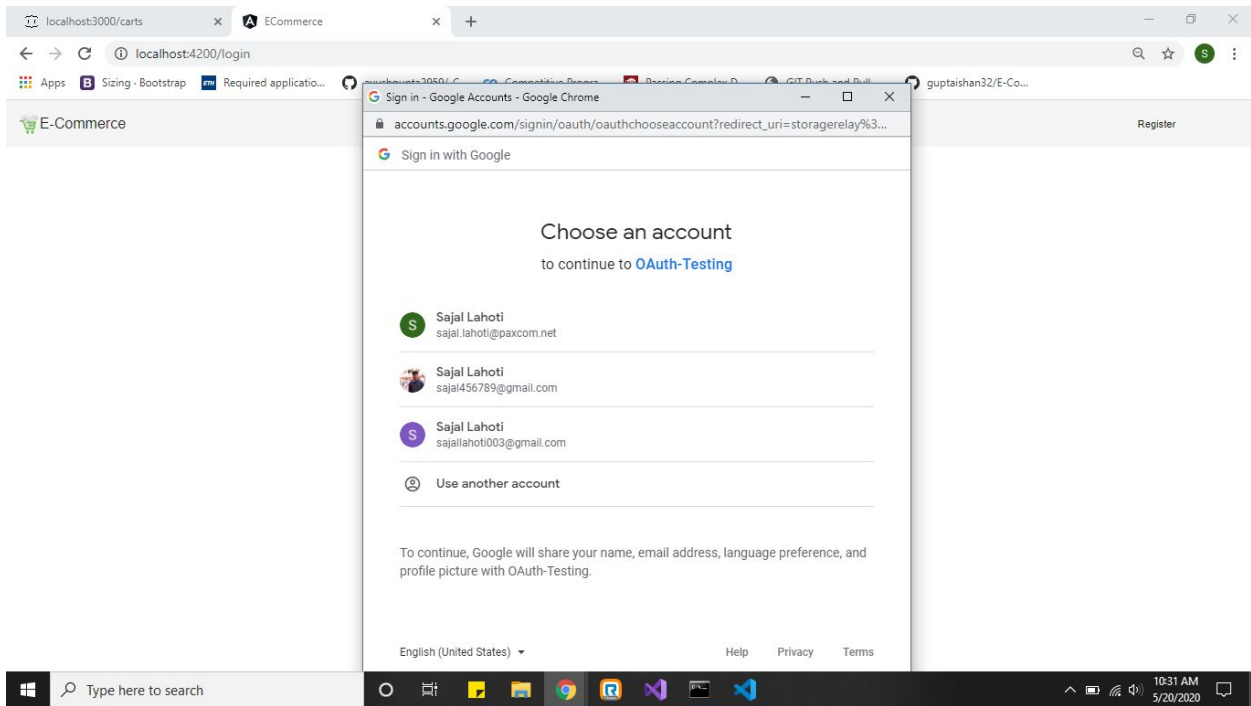


Fig 4.7 OAuth component for login via Google

The register component consists of a template driven form for performing post operation in users database through REST API.

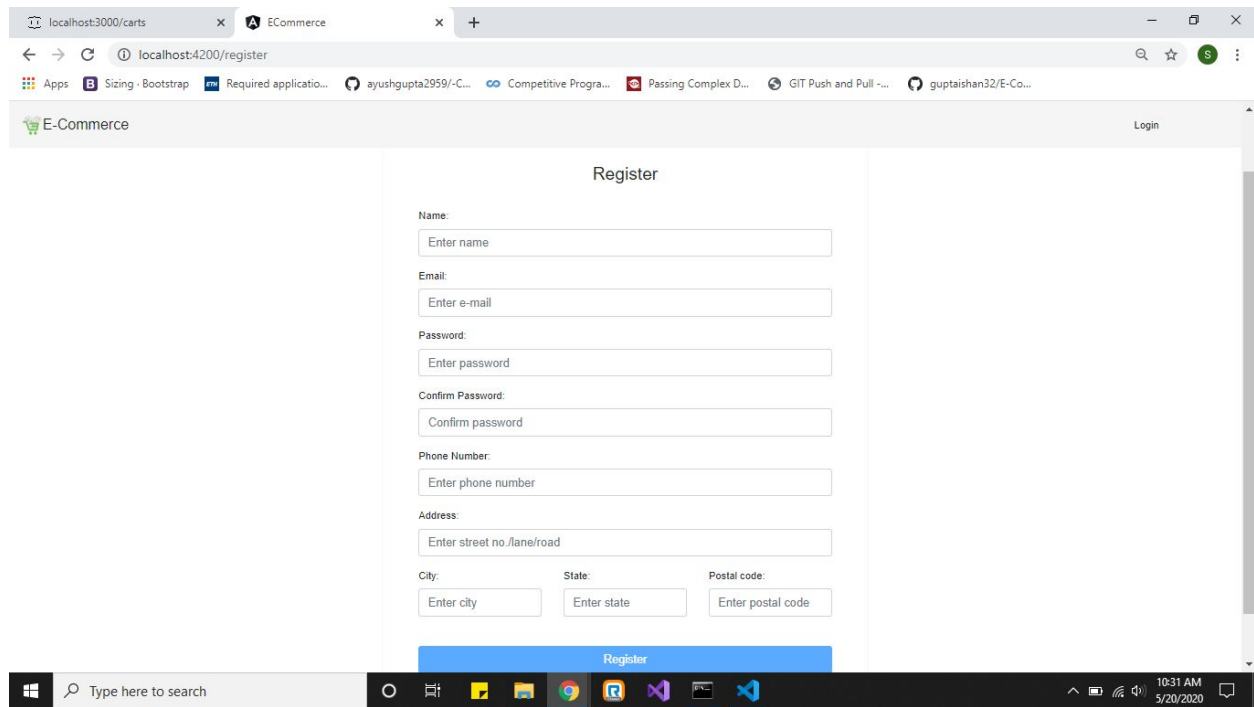


Fig 4.8 Form for registering a user

## View Module

The view product component consisted of cards with product image, product price, product name and product image. Each card is itself a popup component that further displays the description of the product. Both the individual card and the popup component consist of an add to cart feature with the card component limited to one item and popup component limited to six items to add to cart.

The component also contains a side nav bar with different categories which sorts the product based on category.

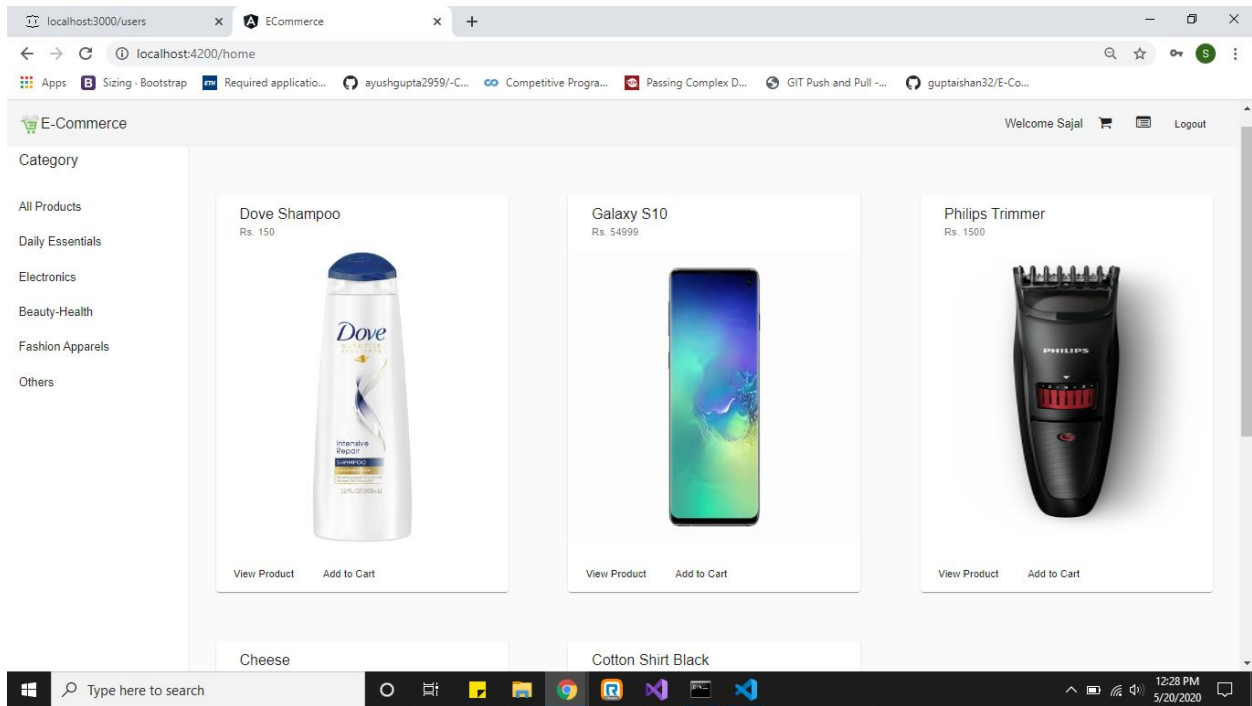


Fig 4.9 View Product Component

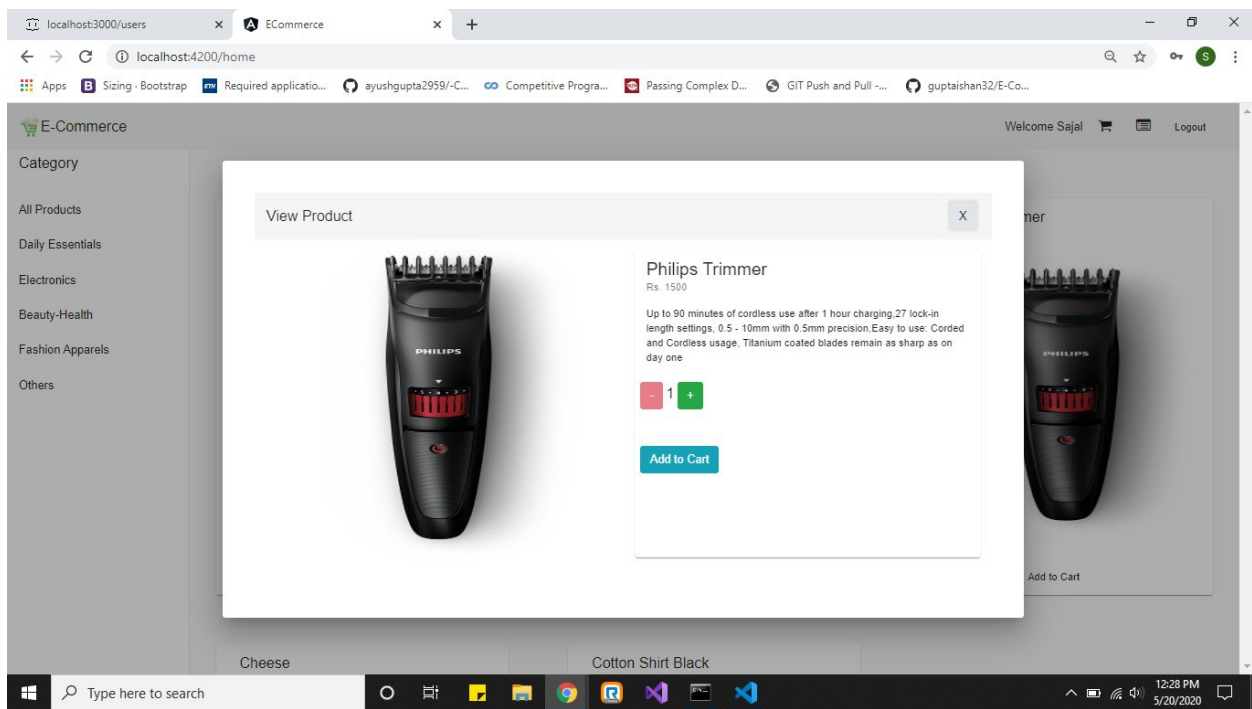


Fig 4.10 Popup component for product.

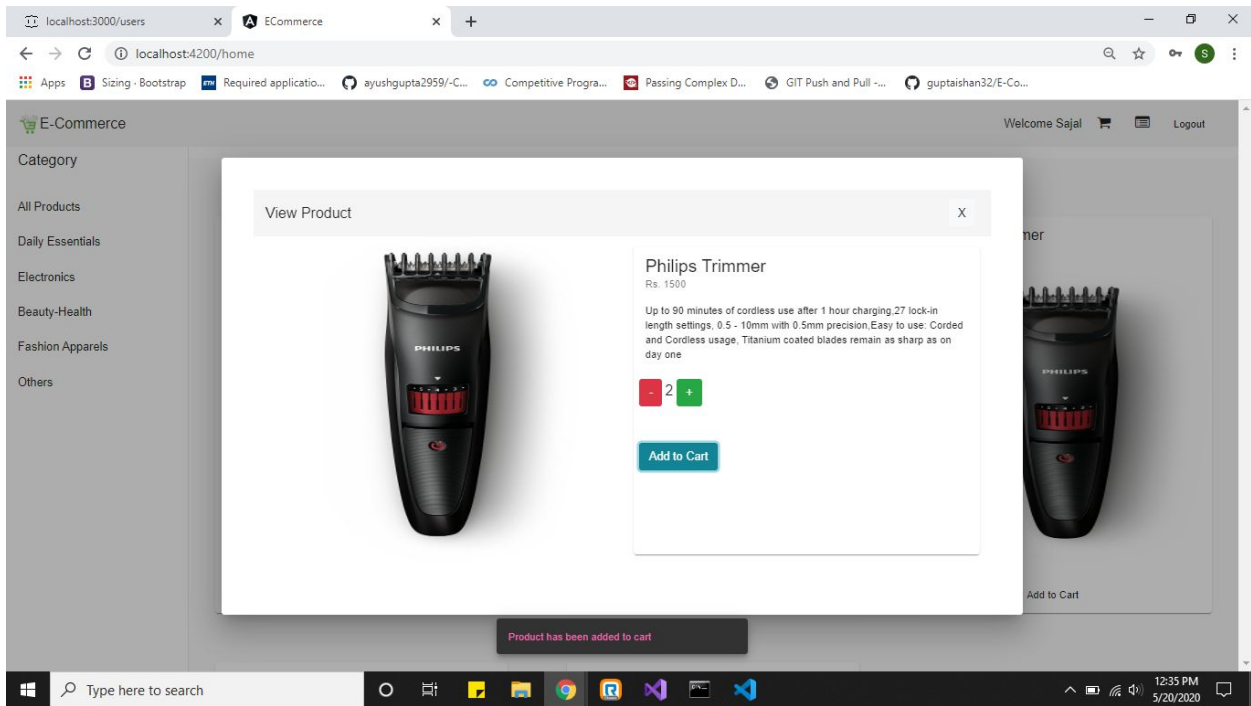


Fig 4.11 Snackbar after adding product to cart

## Cart/Order Module

The third module of the project consisted of the cart component, add address component and invoice component which presented the final bill to the user.

The cart component also consists of save for later feature which acts as a wishlist for user and a ngx pagination component which displays the products available in stock at the bottom left of the screen in case user wants to add a product from cart page itself.

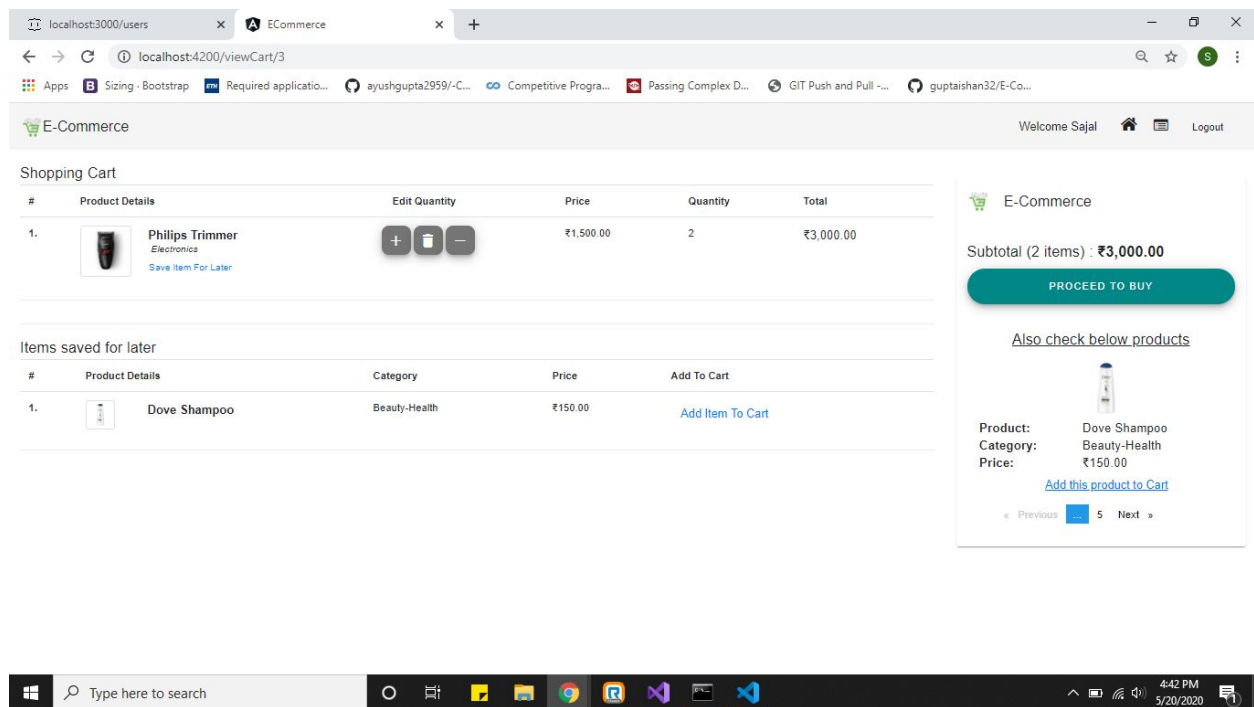


Fig 4.12 Cart component with save for later and pagination component.

The proceed to buy button in the cart components takes the flow of user to selecting address for delivery page where user can choose from existing address or add a new address.



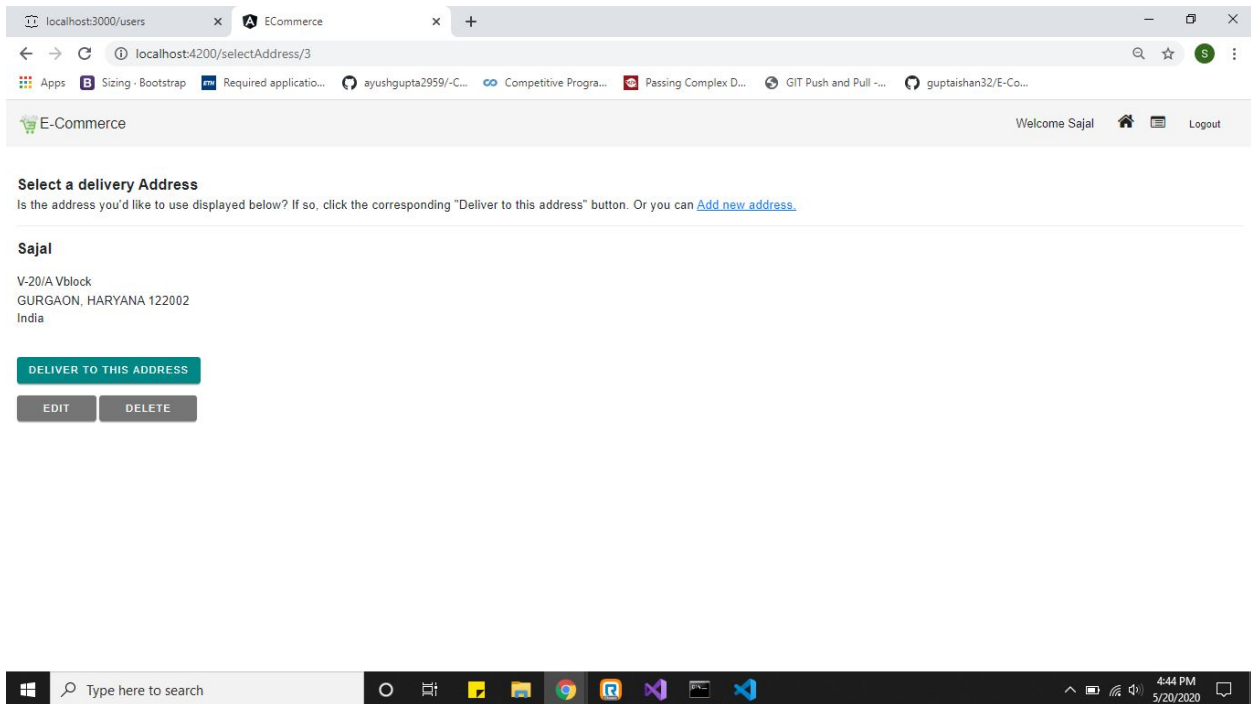


Fig 4.13 Choosing address on delivery address page

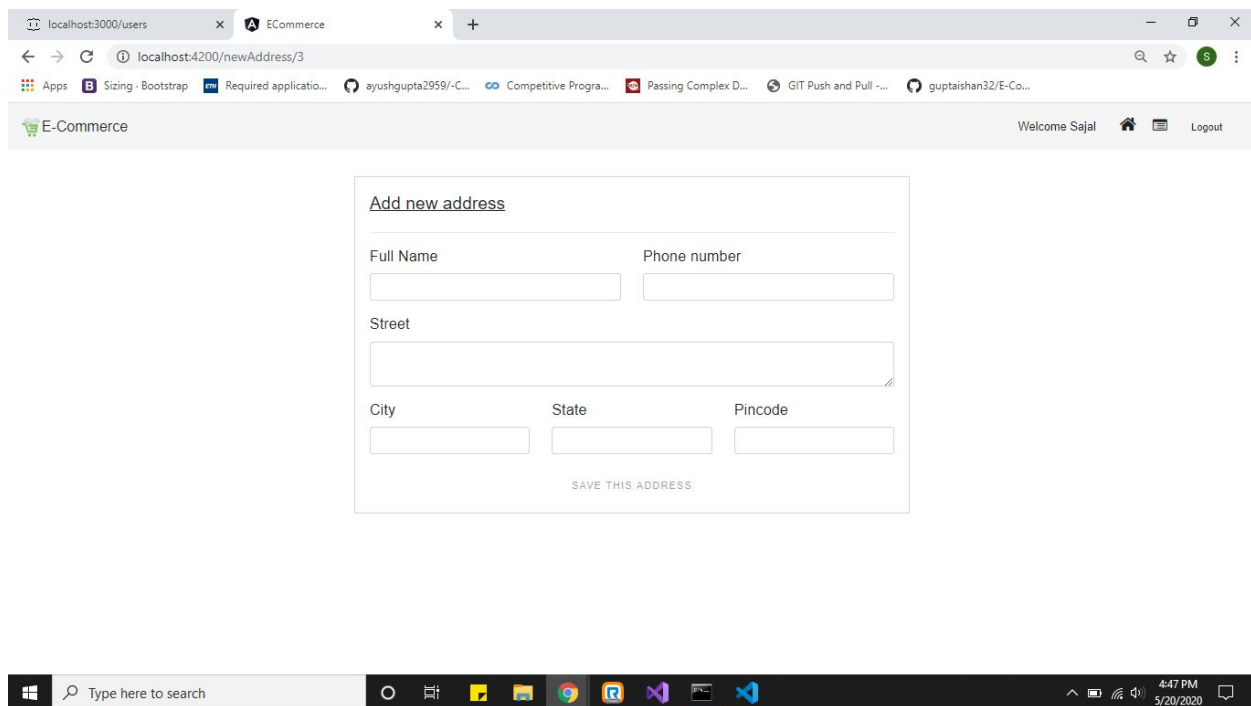


Fig 4.14 Add address page

The flow then goes to the shipping address page where the user can select the shipping address for the order.

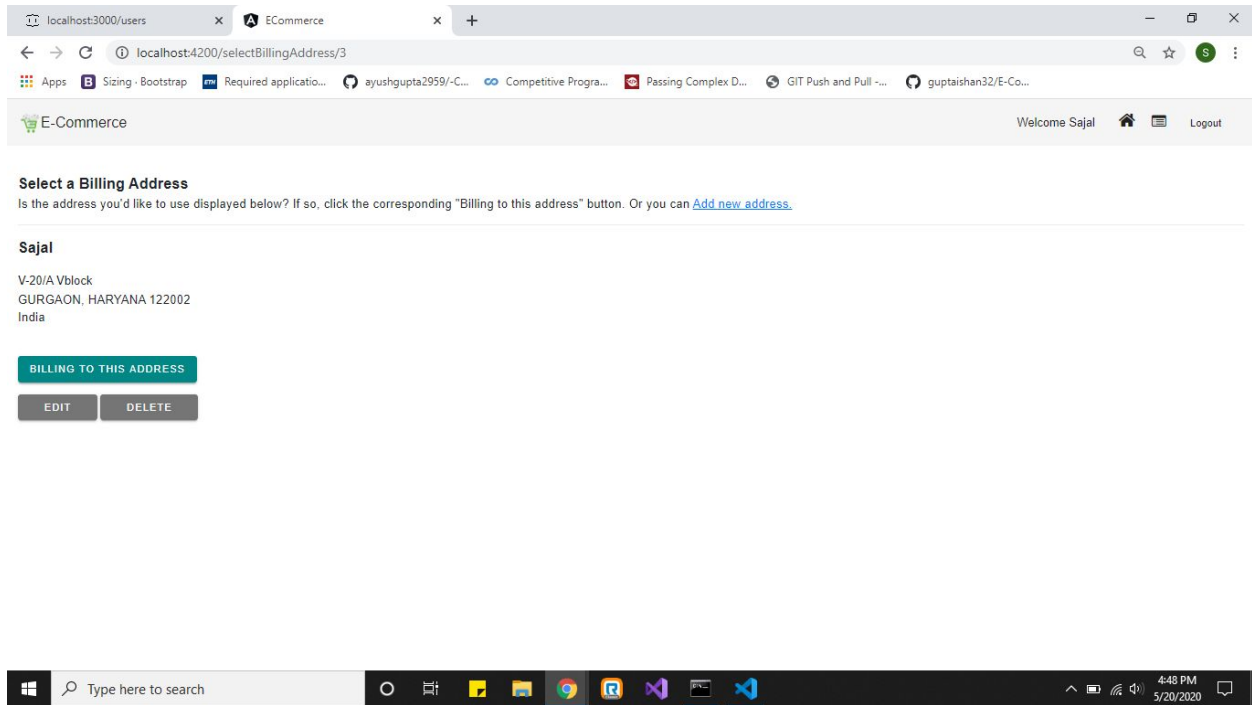


Fig 4.15 Billing address page

Then the flow goes to final review before order. This page consists of place you order button after which the order gets placed.

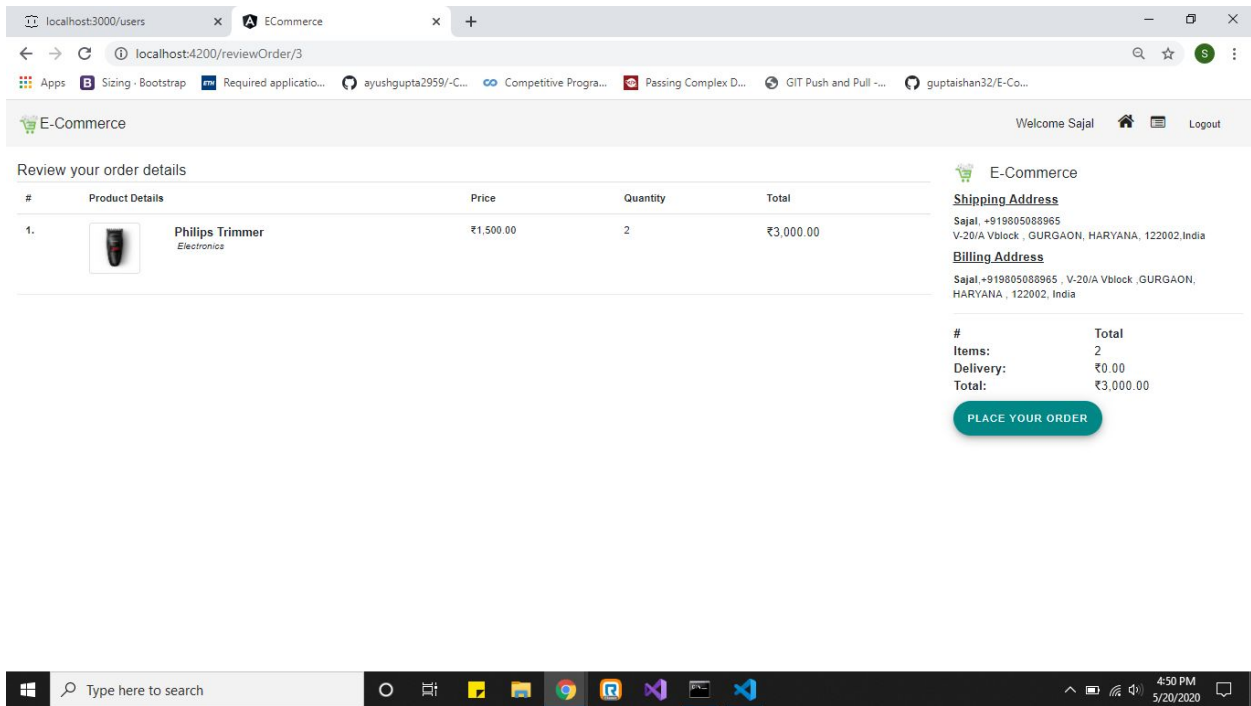


Fig 4.16 Review Order Page

The flow then goes to the page where past orders of the logged in user are displayed sorted on the basis of date from most recent to least recent.

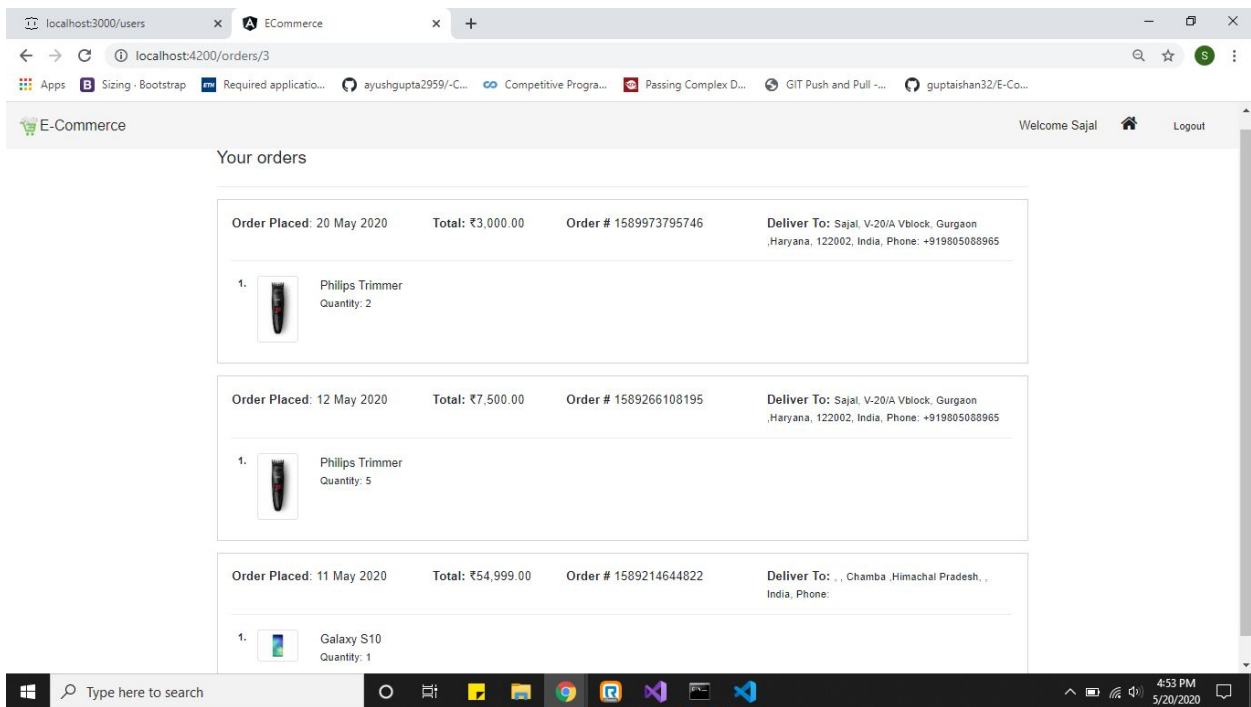


Fig 4.17 Your Order Page

## REFERENCES

- [Angular.io](#)
- [Github.com](#)
- [Paxcom.net](#)
- [Paymentus.com](#)
- [Wikipedia.com](#)
- [Material.angular.io](#)
- [Npmjs.com](#)
- [restfulapi.net](#)

ORIGINALITY REPORT

11%	4%	2%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Institute of Technology Blanchardstown Student Paper	3%
2	en.wikipedia.org Internet Source	3%
3	Submitted to University of Northumbria at Newcastle Student Paper	2%
4	Shawon S. M. Rahman, Nazmul Hossain, Md Alam Hossain, Md Zobayer Hossain, Md Hassan Imam Sohag. "chapter 5 OAuth 2.0", IGI Global, 2020 Publication	1%
5	Submitted to Sri Lanka Institute of Information Technology Student Paper	1%
6	Submitted to Melbourne Institute of Technology Student Paper	1%
7	mafiadoc.com Internet Source	<1%

---

8

## Submitted to University of Reading

Student Paper

<1%

---

9

## Submitted to Manipal University

Student Paper

<1%

---

10

## Submitted to Graphic Era University

Student Paper

---

<1%

---

Exclude quotes      On

Exclude matches      Off

Exclude bibliography      On

# JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

## PLAGIARISM VERIFICATION REPORT

Date: .....14/07/2020.....

Type of Document (Tick): PhD Thesis M.Tech Dissertation/ Report **B.Tech Project Report** ✓ Paper

Name: \_\_\_\_\_ Sajal Lahoti \_\_\_\_\_ Department: \_\_\_\_\_ CSE&IT \_\_\_\_\_ Enrolment No \_\_161461\_\_

Contact No. \_\_\_\_\_ 9805088965 \_\_\_\_\_ E-mail. \_\_\_\_\_ sajal456789@gmail.com \_\_\_\_\_

Name of the Supervisor: \_\_\_\_\_ Dr Aman Sharma \_\_\_\_\_

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters):

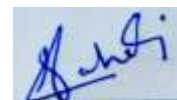
\_\_\_\_\_ INTERNSHIP REPORT AT PAXCOM INDIA PVT LTD \_\_\_\_\_

### UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/ revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

- Total No. of Pages =30
- Total No. of Preliminary pages = 4
- Total No. of pages accommodate bibliography/references = 1

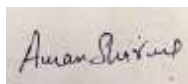
(Signature of Student)



### FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at **11**.(%).

Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.



(Signature of Guide/Supervisor)

Signature of HOD

### FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Excluded	Similarity Index (%)	Abstract & Chapters Details	
	<ul style="list-style-type: none"><li>• All Preliminary Pages</li><li>• Bibliography/ Images/Quotes</li><li>• 14 Words String</li></ul>	<b>11%</b>	Word Counts	
<b>Report Generated on</b>			Character Counts	
		<b>Submission ID</b>	Page counts	
			File Size	

Checked by

Name & Signature

.....

Librarian

Please send your complete Thesis/Report in (PDF) & DOC (Word File) through your Supervisor/Guide at [plagcheck.juit@gmail.com](mailto:plagcheck.juit@gmail.com)