CCD-Report-PoojaPantha-77356785.docx

by Pooja Pantha

Submission date: 25-Jan-2025 09:43PM (UTC+0545)
Submission ID: 2571155252
File name: CCD-Report-PoojaPantha-77356785.docx (20.9M)
Word count: 5734
Character count: 35048

Leeds Beckett University

A Comprehensive Analysis of Cloud Computing: From Technology Overview to Service Deployment

Module: Cloud Computing Development

Student Name: Pooja Pantha

Student ID: 77356785

Word Count: 4681

Year: 2024/2025

Contents

Table of Figures

1. Introduction

Cloud computing has emerged as one of the critical technologies in current IT environment where applications and services can be hosted in flexible, scalable and cost-effective manner by business and developers (Alam, 2020). The objective of this report is to review the case on the Cloud-based Service and to explore the provider, service model, and development technologies to design a useful and effective system.

Amazon Web Services (AWS) has been selected for this report. EC2 instance will be utilized for hosting the services. AWS EC2 has been chosen because of the reliability, flexibility, and the large set of tools for scalable, on-demand computing allowing to successfully host APIs (Amazon Web Services, 2024). IAAS model has been chosen for the purpose of utilizing the AWS EC2 instances for elastic computing capabilities. The website which is developed for this report uses HTML, CSS, and JavaScript to create a responsive frontend interface. Node.js is used for backend development, while MongoDB is used for the database.

The Website BodyMetrics offers several interactive features like BMI calculator, measures the distance when user provides two locations for cycling, walking or just for a ride, workout plan according to the need. These features are hosted through cloud APIs, so there is little latency and much availability for the user while offering basic calculation and real time data retrieval responses. The service will be running on the AWS EC2 instances, and it will contain backend API and all interconnections. The aim is to offer a clean and infinitely expandable user experience and give the users easy access to the information and tools they need through the Web interface of the project.

2. Overview of Cloud Computing Technologies

2.1 Definition of Cloud Computing and its characteristics

Cloud Computing describes a service delivery model which provides internet-hosted virtualized resources that can adjust their capacity automatically based on users' requirements (Gupta, 2013). Online IT infrastructure delivery enables users to access resources on demand through pricing model pay-as-you-go. Cloud providers like AWS enable businesses to access computing resources, including storage and databases, on demand, eliminating the need to invest in and maintain physical infrastructure (Amazon Web Services, 2021).

Characteristics of Cloud Computing

Cloud Computing stands out from local servers due to important elements that define its working principles. The main characteristics are:

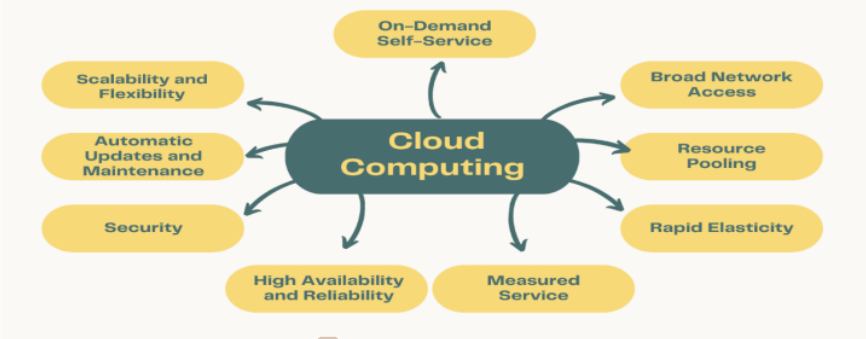


Figure 1: Characteristics of Cloud Computing

a. **On-demand self-service:**
   The user can utilize computing capabilities such as network storage automatically without requiring human interaction with service provider.
b. **Broad network access:**
   All capabilities are available over network, and it is accessible from anywhere by any client platforms (e.g., mobile phones, tablets, laptops, etc.)
c. **Resource pooling:**
   Cloud service provider pools his computer's resources to multiple consumers with different physical resources dynamically assigned according to consumer demand (Scribd, n.d.).
d. **Rapid elasticity:**
   Capabilities can be elastically set-up and scaled rapidly according to consumer demand.
e. **Measured service:**
   Resource utilization becomes approachable for management by both cloud service providers and end users through tracking and documentation procedures (Simplilearn, 2014)
f. **High availability and reliability:**
   Cloud providers offer robust infrastructure with failover mechanisms and multiple redundant systems across different geographic locations to ensure high availability and minimal downtime.
g. **Security:**
   Cloud service providers defend cloud-stored information and programs through encryption algorithms and security barriers with strict permission controls. Working with industry standards delivers protection and privacy for stored information.
h. **Automatic updates and maintenance:**
   Cloud service clients benefit from automatic system updates which is handled by providers as users dedicate their time to core work activities.
i. **Scalability and flexibility:**
   Every type of business benefits from cloud computing because users can instantly modify their resource capacity to suit their evolving requirements.

2.2 Cloud computing service models

Cloud Computing enables users to store information on many interconnected Internet-based servers instead of using personal devices for data storage and management. Companies that offer cloud service models are cloud providers and set prices based on customer usage (GeeksforGeeks, 2023). The most common cloud computing service models are:

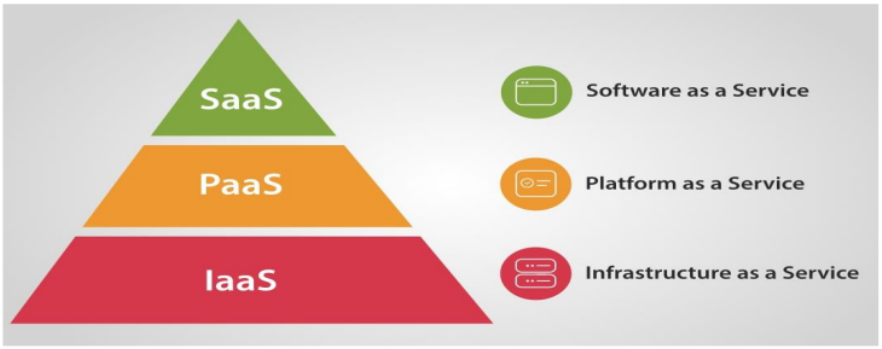


Figure 2: Cloud Computing Service Models (Litslink, 2024)

a. **Infrastructure as a Service (IaaS)**

IaaS also known as Hardware as a Service (HaaS), is a computing infrastructure managed over the internet (Javatpoint, 2025). IaaS lets users access virtual computing resources through the internet including virtual machines, storage and networking. Cloud users get OS and application freedom with hardware support provided by service providers.

Suitability:

- Businesses use this type for hosting platforms because it lets them maintain full control of their web environment for development testing purposes.
- This service lets businesses grow their resources without buying extra physical machines.

Pros:

- Services are highly scalable
- GUI and API-based access

Cons:

- Users are fully responsible for keeping their system secure and updating their programs.
- Technical experts are required in order to manage servers and system.

b. **Platform as a Service (PaaS)**

PaaS serves as a platform solution that gives developers everything they need to create and test applications while handling the important system maintenance tasks. A PaaS provider operates its own system for hardware and software. PaaS free users from installing proprietary hardware and software requirements to build and launch new applications (GeeksforGeeks, 2023).

Suitability:

- The platform lets developers write their code freely while freeing them from infrastructure responsibilities.
- Suitable platform for web and mobile application developers who need quick releases.

Pros:

- The same development application lets multiple individuals join and operate it together.
- Support multiple languages and frameworks.
- Automatic updates and maintenance by the provider.

Cons:

- User can have minimal influence over the essential parts that make application run.
- Vendor lock-in can be a concern when migrating to another provider.

c. **Software as a Service**

Saas also known as “on-demand software”, is a service provider that manages and runs applications through their servers (JavatPoint, 2025). Users receive software applications online when they pay for subscription-based plans. The software operates through internet browsers so users can use it without installing client versions.

Suitability:

- Best for businesses looking for ready-to-use software solutions such as email, CRM, or collaboration tools.
- Suitable for users with minimal technical knowledge who require cloud-based applications for everyday tasks.

Pros:

- Any internet-connected device lets you access this service.
- Users need not maintain its operation in any way.
- Users only pay for what you use at any given time.
- Updates are applied automatically.

Cons:

- Data may be less secure since it lives on remote servers instead of user's own servers.
- Need of an Internet connection to access the site throughout.
- Limited customization compared to self-hosted solutions.

d. Function as a Service (FaaS)

FaaS is a serverless computing service that lets developers run code automatically when events happen without managing the infrastructure. Cloud providers activate and expand resources automatically as the usage increases. It is also same as PaaS. Both PaaS and FaaS are providing the same function but there is still some differentiation in terms of Scalability and Cost.

Suitability:

- Perfect for systems where real-time events trigger responses from separate services and deal with changing workloads.
- Businesses can concentrate more on code development by using this method to cut operational costs.

Pros:

- Automatic server scaling takes care of server maintenance needs.
- Users only pay when function executes.
- Functions can be written in any programming language.

Cons:

- Functions have time limit or execution.
- Debugging systems often gets complicated and complex.
- Limited customization compared to self-hosted solutions.
- Users have to secure their data by themselves due to security concerns.

Comparison of cloud service models

Feature	IaaS	PaaS	SaaS	FaaS
Control Level	High	Medium	Low	Very low
Maintenance	User-managed	Partially managed	Fully managed	Fully managed
Cost Structure	Pay for infrastructure	Pay for platform usage	Subscription based	Pay per function execution

Flexibility	Highly flexible	Limited by platform	Predefined functionalities	Limited function scope
Scalability	Manual or automated	Automatic	Automatic	Automatic

2.3 Cloud computing deployment models

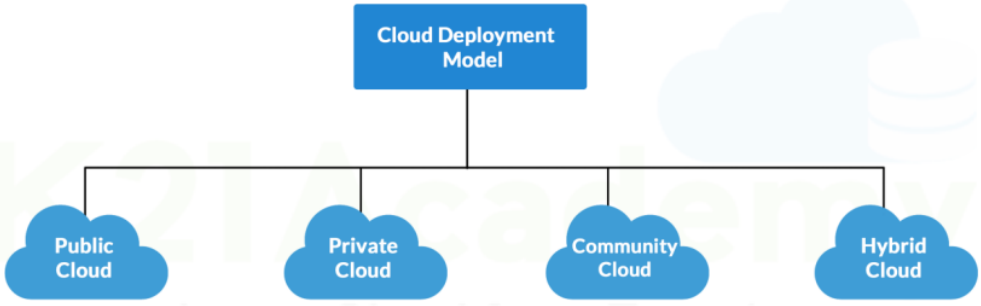


Figure 3: Cloud Computing Deployment Models

The Cloud Deployment Model runs on a virtual platform whose structure evolves depending on user access levels and data storage preferences (GeeksforGeeks, 2023). Cloud deployment helps businesses work more effectively and maintain their market advantage. Four deployment options exist for running cloud solutions. There are different kinds of deployment models, and they are Private Cloud, Public Cloud, Hybrid Cloud, Community Cloud, Multi-Cloud and Virtual Cloud (Patel, 2021). From these, three are most common:

a. **Public Cloud**

Public cloud is self-explanatory. It is accessible to the anyone on the internet. Third-party cloud providers deliver server, storage and network resources through the internet to many customers who pay only for what they use. Some of the examples are AWS (Amazon Web Services), Microsoft Azure, Google Cloud Platform.

Characteristics:

- Shared infrastructure among multiple tenants.
- Accessible over the internet.
- The provider is responsible for maintenance and management.
- Elastic scalability to accommodate varying workloads.

Suitability:

- This approach fits new organizations and businesses of medium scale that handle flexible work situations.
- Servers provide the perfect platform for web apps together with development/testing spaces and group tools.

Pros:

- The system requires no hardware investments at the start of the project.
- System grows or shrinks to match the ups and downs of customer demand.
- The system delivers worldwide service access.

- Users need minimal support for product maintenance.

Cons:

- Less secure
- Low customization

b. Private Cloud

Private cloud deployment follows an exact opposite approach to how public clouds work. Each private cloud instance serves just one client or user individually. A private cloud offers an exclusive cloud service to a single organization that gives complete management and protection of their resources. Their cloud infrastructure can operate either locally or through service providers as they maintain private control over the system. Some of the examples are VMware vSphere, OpenStack, Microsoft Azure Stack.

Characteristics:

- Dedicated resources for a single organization.
- Greater control over data security and compliance.
- Customizable infrastructure to meet specific business needs.

Suitability:

- Large organizations with tight security standards especially in healthcare, finance, and government should pick this option.
- Applications with complete data and resource control and best configuration can use this system.

Pros:

- Delivers both stronger protection and enhanced data confidentiality.
- Creates IT systems that match user's business goals precisely.
- Managed compliance and how well the system works.

Cons:

- Need to invest a large sum right from the start to set up hardware and infrastructure.
- The system requires an ongoing qualified staff presence to keep things running smoothly.
- The growth potential in private cloud can be less than from public cloud providers.

c. Hybrid Cloud

Proprietary software enables hybrid cloud computing to unite public and private cloud networks effectively. The hybrid cloud lets organization store data in both public and private systems through simple data sharing (Cloudfare, n.d.). Organizations select this design to get the advantages of public and private cloud solutions while managing their costs and expansion needs. Some of the examples are AWS Hybrid Cloud, Azure Hybrid Cloud, Google Anthos.

Characteristics

- Combines private and public cloud environments.
- Provides flexibility in resource allocation based on business needs.
- Enables seamless data transfer and integration between environments.
- Allows organizations to keep sensitive data on-premises while leveraging the public cloud for scalability.

Suitability:

- Small businesses and organizations need this when demand changes frequently and quickly.
- This system lets companies restore all important information and resume normal business activities during unexpected events.

Pros:

- Flexible to move workload placement where it suits the business best.
- It lets to access affordable public cloud services on demand.
- The solution allows to keep business operations running through failover mechanisms.

Cons:

- Complex to manage and integrate different environments.
- Issues to keep data consistent and compliant between the two clouds.
- The expenses rise when combining a public cloud with a private one.

Comparison of Cloud Deployment Models

Feature	Public Cloud	Private Cloud	Hybrid Cloud
Ownership	Cloud provider	Organization	Combination
Security	Standard security measures	High security and control	Moderate to high security
Cost	Pay-as-you-go	High upfront cost	Balanced
Scalability	High	Limited	High
Maintenance	Handled by provider	Requires in-house expertise	Shared responsibility
Customization	Limited	High	Moderate

2.4 Cloud computing providers

The Coud Service Providers (CSP) are the IT businesses that provide scalable online services to deliver computer resources including processing power networking, cloud-based storage, and platform (Google Cloud, 2024). From the data of 2024, top 10 CSP are Amazon Web Services (AWS), Microsoft Azure, Kamatera, Alibaba Cloud, Oracle Cloud, IBM Cloud (Kyndryl), Tencent Cloud, OVHcloud, DigitalOcean, and Linode (GeeksforGeeks, 2024). The three of them are in detailed overview below:

A. Amazon Web Services (AWS)

AWS is known for its powerful scalability and reliability with extensive global infrastructure. It also offers a broad spectrum of cloud services, which consist of machine learning, analytics, storage, databases, and networking.

Key Services:

- **Compute:** Amazon EC2, AWS Lambda
- **Storage:** Amazon S3, EBS, Glacier
- **Databases:** Amazon RDS, DynamoDB
- **Networking:** Amazon VPC, CloudFront
- **AI/ML:** AWS SageMaker, Rekognition

Pricing Model:

- Pay-as-you-go pricing with free-tier options.
- Reserved instances for long-term cost savings.
- Spot instances for cost-efficient, non-critical workloads.

Service Level Agreement (SLA):

- 99.99% uptime for EC2 and S3 services.
- Compensation is provided in case of downtime exceeding SLA guarantees.

Pros:

- AWS is cost-effective as it works on a pay-as-you-go pricing model.
- Efficiently manage and secure Windows workloads.

Cons:

- Complex pricing structure that can be difficult to predict.
- Steeper learning curve for beginners.

B. Microsoft Azure

Microsoft Azure, also known as Windows Azure, is the fastest-growing CSP (Yasar and Bigelow, 2022). It is a cloud platform that emphasizes hybrid cloud capabilities, enterprise integrations, and smooth compatibility with Microsoft products like Windows Server, Active Directory, and Office 365.

Key Services:

- **Compute:** Azure Virtual Machines, Azure Functions
- **Storage:** Azure Blob Storage, Disk Storage
- **Databases:** Azure SQL Database, Cosmos DB
- **Networking:** Azure Virtual Network, CDN
- **AI/ML:** Azure Machine Learning, Cognitive Services

Pricing Model:

- Pay-as-you-go, reserved instances, and hybrid licensing options.
- Free tier available for select services.
- Discounts for enterprises through volume agreements.

Service Level Agreement (SLA):

- 99.95% uptime for most services.
- Compensation is credited for unplanned downtime beyond SLA guarantees.

Pros:

- Excellent hybrid cloud capabilities.
- Enterprise-grade security and compliance.

Cons:

- Some services may be region dependent.
- Complexity in managing hybrid cloud solutions.

C. **Oracle Cloud**

Oracle Cloud Platform, offered by the Oracle Corporation, helps to build, deploy and manage workloads in the cloud or on-premises (Barney, n.d.). It supplies full cloud technology services that cover computer processing, data space and internet connectivity alongside powerful Oracle Autonomous Database systems.

Key Services:

- **Compute:** OCI Compute Instances, Functions
- **Storage:** OCI Object Storage, Block Storage, Archive Storage
- **Databases:** Oracle Autonomous Database, Exadata Cloud Service
- **Networking:** Virtual Cloud Networks (VCN), Load Balancers
- **AI/ML:** OCI Data Science, AI Services

Pricing Model:

- Pay-as-you-go with predictable pricing.
- Discounts for reserved capacity and long-term commitments.
- Always Free Tier, which includes access to Autonomous Database and compute resources.

Service Level Agreement (SLA):

- 99.995% uptime for critical services, ensuring high availability.
- Financially backed guarantees for performance and reliability.

Pros:

- Best-in-class database solutions, ideal for businesses using Oracle products.
- High performance and low-latency architecture optimized for enterprise workloads.

Cons:

- Limited global reach compared to AWS and Azure.
- Smaller range of general-purpose cloud services.

Comparison of Cloud Providers

Feature	AWS	Azure	Oracle
Market Share	Largest	Second Largest	Smaller, but expanding
Compute Services	EC2, Lambda	Virtual Machines, Functions	Compute Instances, Functions
Storage Options	S3, EBS, Glacier	Blob Storage, Disk Storage	Object Storage, Block Storage
Database Services	RDS, DynamoDB	SQL Database, CosmosDB	Autonomous Database, Exadata
Security Features	Strong security compliance	Seamless enterprise security	Strong database security features

2.5 Selection and justification of the cloud provider, the service model and implementation technology

After exploring the multiple cloud providers, their features, prices, infrastructure, AWS has been chosen as the cloud provider for this project. AWS is chosen because it supports different workloads and allows to pay for what is used without spending money upfront. AWS delivers complete documentation and community assistance to simplify efficient application setup and administration. IaaS is used in order to get more control over cloud environment since IaaS enables to customize operating system, software stack, and application installation options. The decision is fully driven due to the perfect solution for configuration authority and unrestricted software component deployment that IaaS offers. In IaaS settings user can better manage how the resources work and check their status to keep web application at its best (Google Cloud, n.d.).

The web application which is developed for this project, uses HTML, CSS and JavaScript for its frontend, to create an interactive user experience. The backend development depends on Node.js because its event-driven structure makes the application faster and more extensible than other runtimes (freeCodeCamp, 2020). Node.js manages multiple client requests at once making it perfect for developing fast and scalable web applications. The website uses MongoDB as its database which lets it handle both structured and partially structured data while maintaining flexibility. The document structure in MongoDB works best when user needs quick data operations and can expand across applications (MongoDB, n.d.).

The development process is done using Visual Studio Code (VS Code) to build projects because this popular integrated development environment supports web technologies and backend features through added extensions. For quicker setup Amazon Linux 2 is used which provides a secure runtime environment tailored to Node.js and MongoDB applications. System uses Apache as the web server to deliver static files promptly while handling user requests without delay.

3. Development and Deployment of the Cloud Service

3.1 Service and Client Development

3.1.1 Service specification

The service developed for this project is "BodyMetric" which is a fitness-oriented website that combines multiple tools to let users see their fitness development while making better health choices. The platform combines several service options through API functions while primarily focusing on two calculation types plus workout plan generation and user contact. Users can enter their height and weight to learn their body mass index using BMI index measurement tool. Through this information the system assigns body mass index categories to users then explains their health status. Users can check travel time and distance using distance measurement tool for walking, cycling and driving modes. The tool helps users schedule their workouts and transportation routes better.

Alongside the fitness calculators this service creates custom workouts that meet user requirements for their fitness goals. Using the website people can tell the service about their starting abilities and what they want to achieve so it creates the best workout plan for them. Through a feedback mechanism users can interact with the platform to tell their experience and ask for assistance while also helping for improvement of the service.

BodyMetrics Fitness Website UML Use Case Diagram

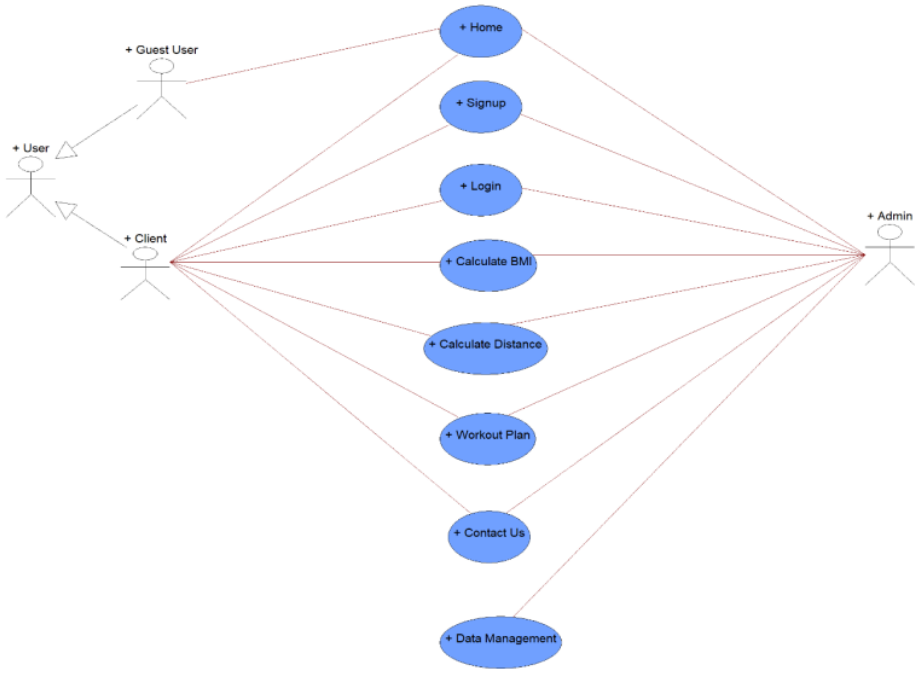


Figure 4: Use Case Diagram: Interaction between users and the core functions of the system

3.1.2 Service development and implementation

The following code screenshots and flowchart demonstrate the core logic and workflows for the system:

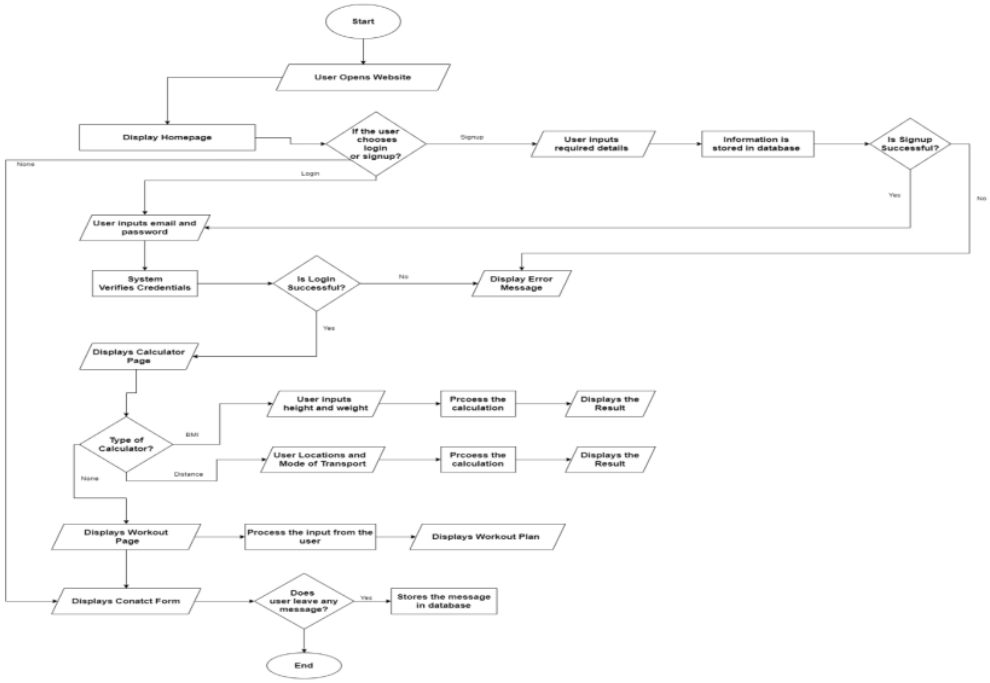


Figure 5: Flowchart of the working mechanism

As the above flowchart represents, after users open the website, the home page will be displayed. Users won't have access to the calculator and workout plan pages unless they log in.

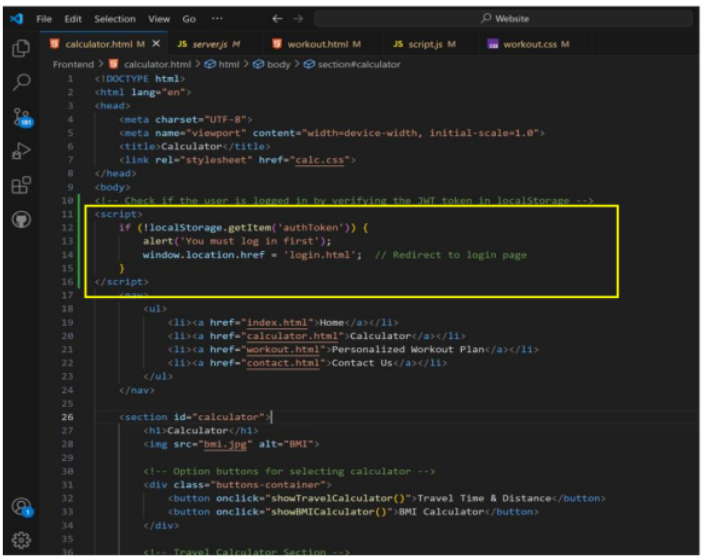


Figure 6: Screenshot of authorization code in calculator.html

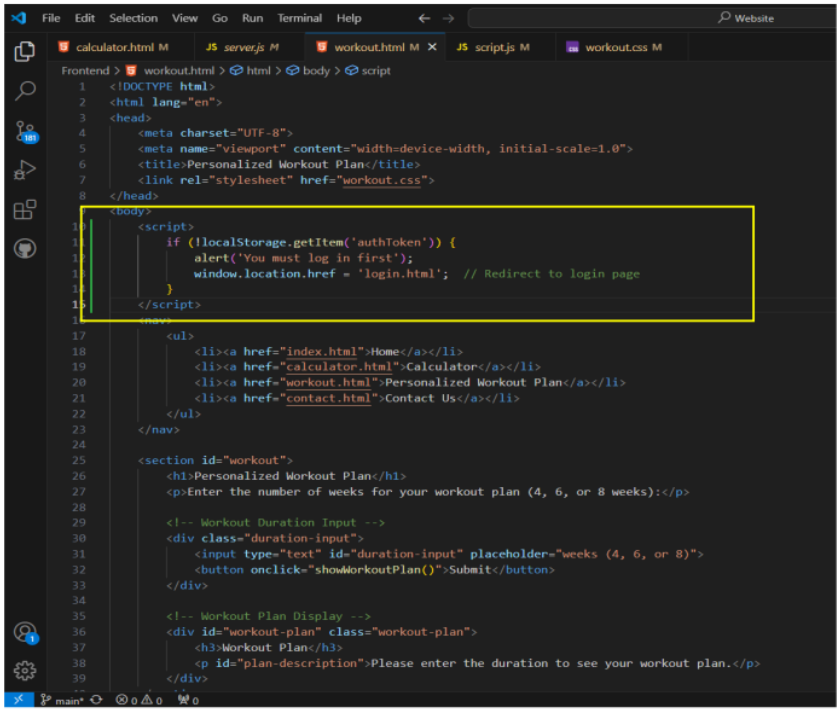


Figure 7: Screenshot of authorization code in workout.html

In the above screenshots, the yellow rectangle highlights the code that requires the login before accessing those pages.

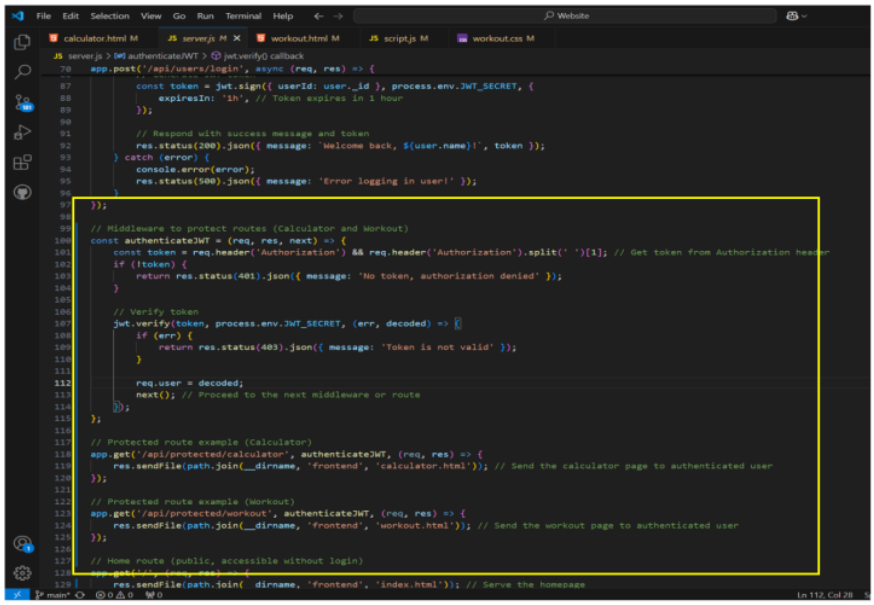


Figure 8: Screenshot of authorization logic in server.js

The above screenshot shows that the server.js handles the backend authentication token.

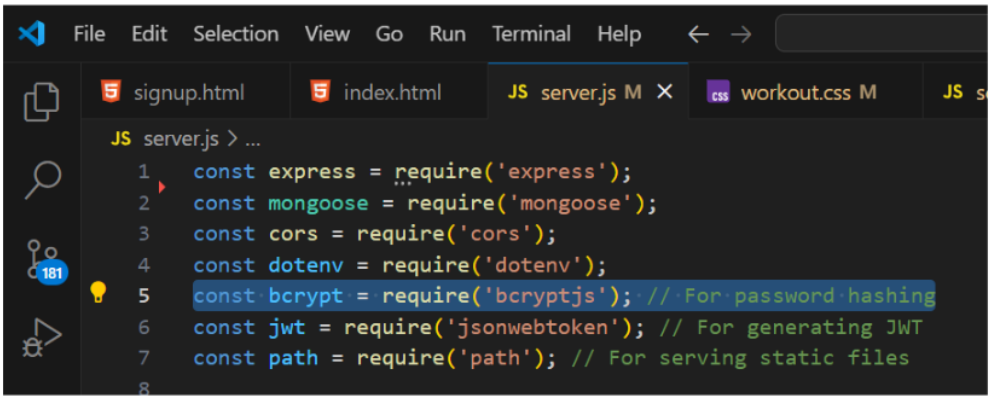


Figure 9: Screenshot of password hashing code in server.js

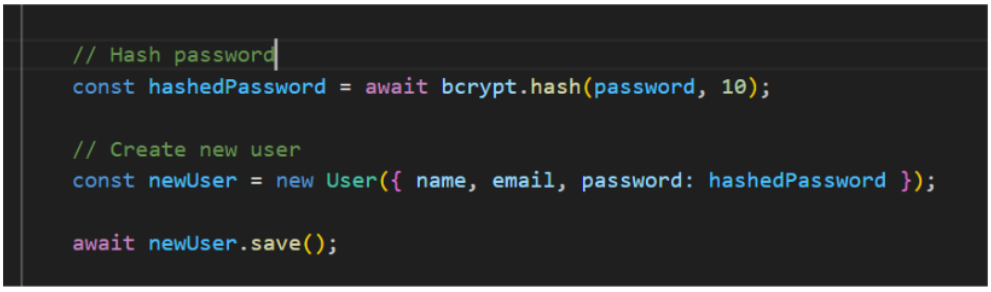


Figure 10: Screenshot of password hashing code in server.js

The above screenshot shows the password hashing logic when new users sign up for the website.

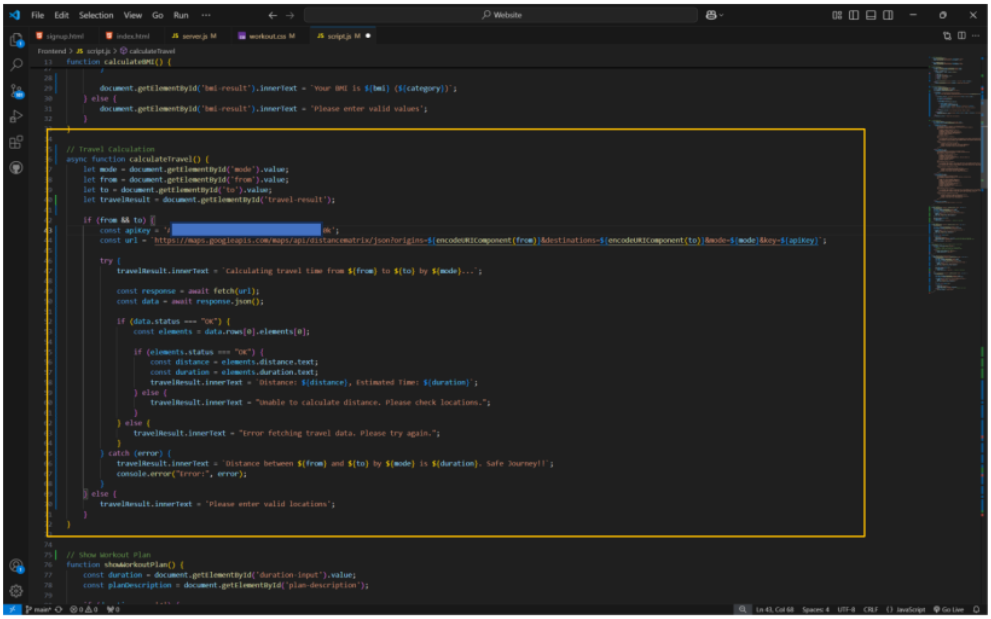


Figure 11: Screenshot of travel & distance calculation logic in server.js

The above screenshot shows the travel and distance calculations logic. Fetched API key from google cloud console so that location calculations can be done through google maps. API key is hidden in this screenshot due to security concerns.

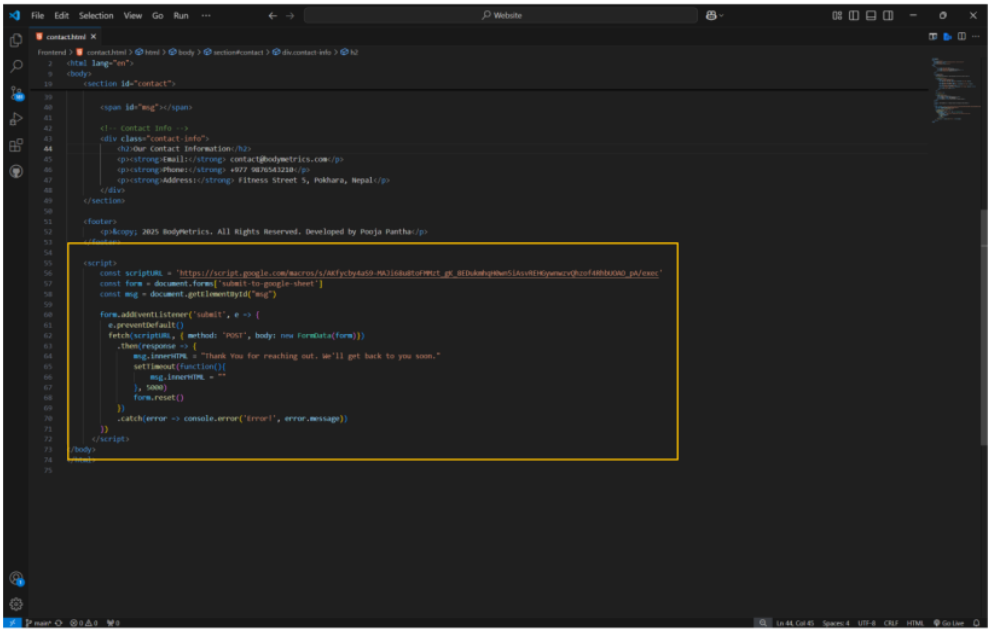


Figure 12: Screenshot of contact form data store code in frontend/contact.html

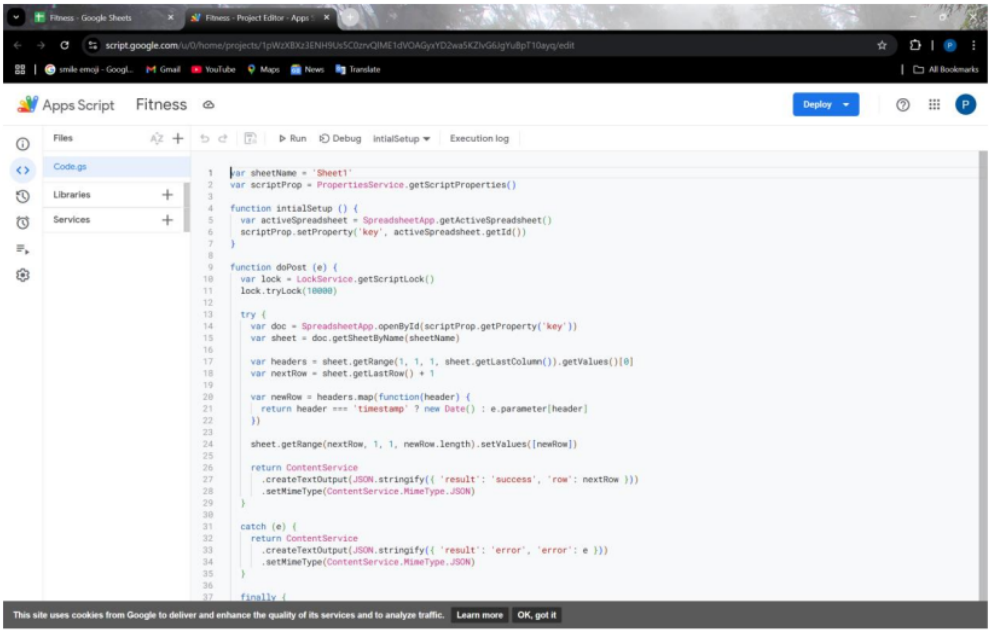


Figure 13: Screenshot of contact form data store code in google sheets

The above screenshot is the JavaScript logic for the contact us form which is submitted by users. It is stored in a google sheet. Even google sheet has a JavaScript in the Apps Script. The Google sheet screenshot storing data is in the next section below.

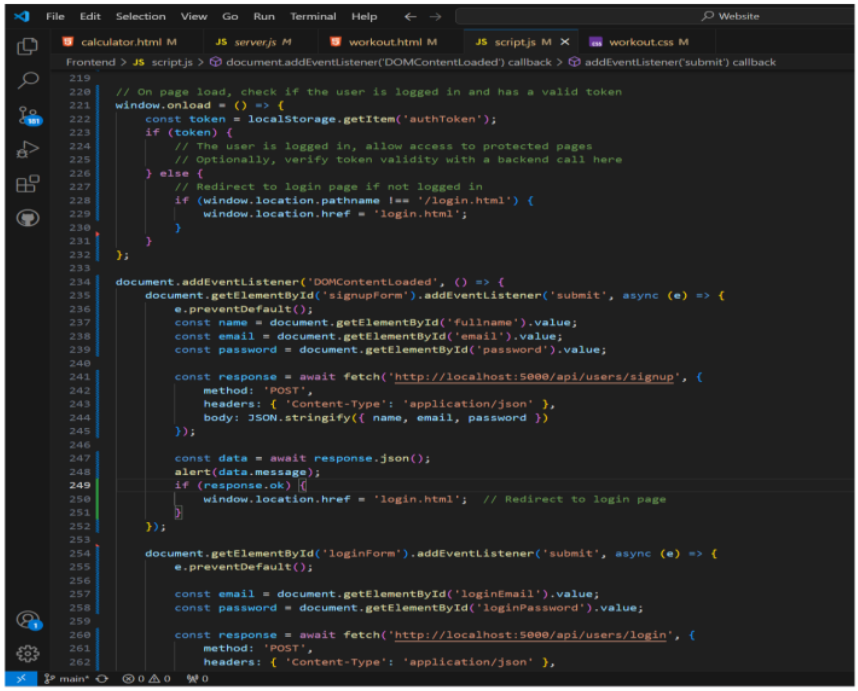


Figure 14: Screenshot of JavaScript code in frontend/script.js

The above screenshot displays script.js which has the frontend logic for the website that handles login and signup.

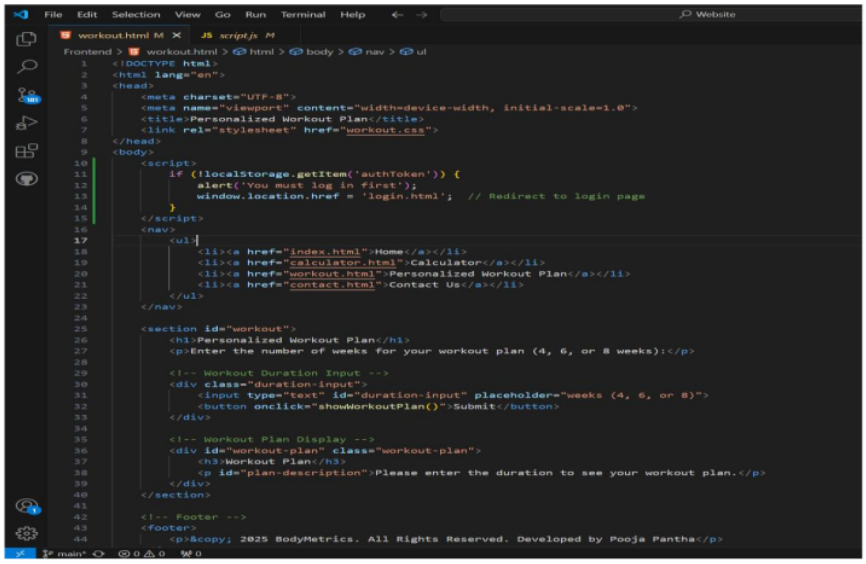


Figure 15: Screenshot of workout plan code in frontend/workout.html

The above code is the workout.html which displays a personalized workout plan after users input their choice.

3.1.3 Client development and implementation

A web-based application built with HTML CSS JavaScript for the frontend interface makes easy for users to interact with the service.

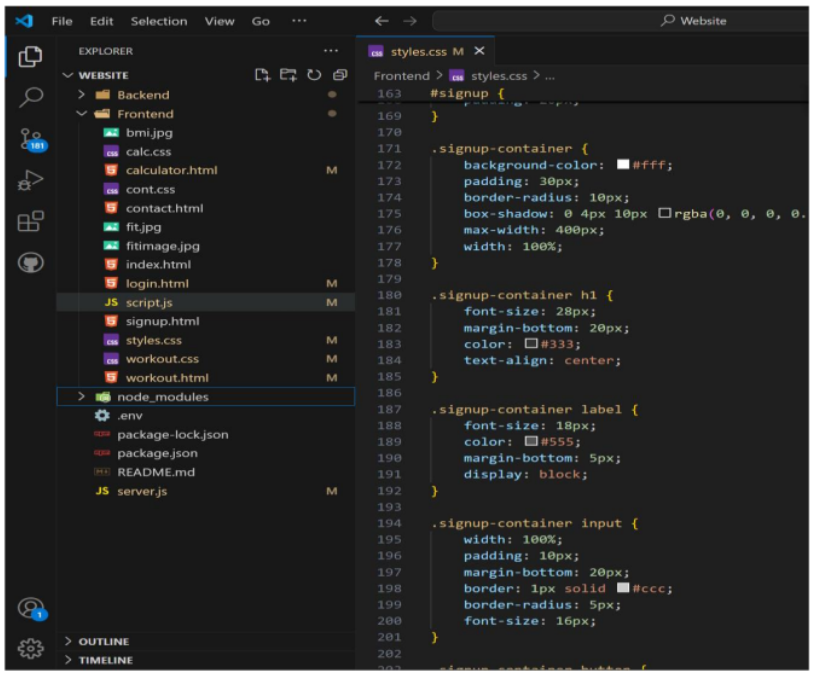
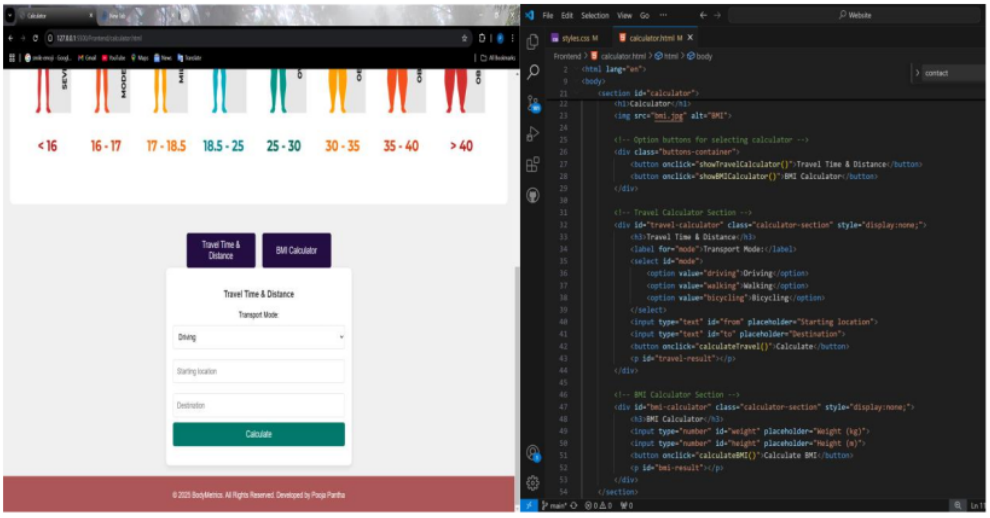
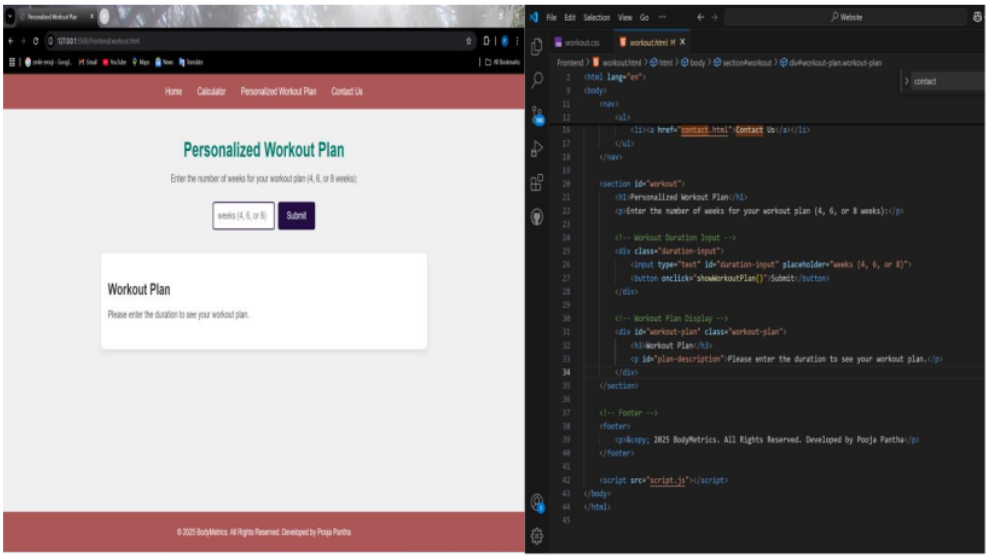
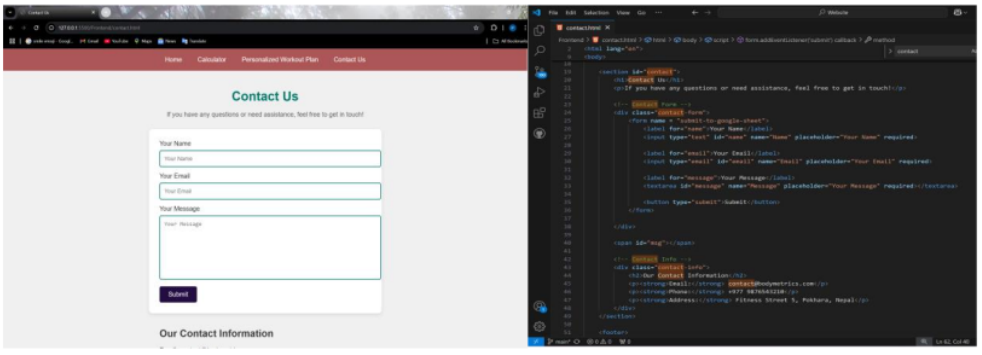


Figure 16: Screenshot of files created during development

The above screenshot has the files that handles UI for the website. Different CSS, HTML and JS files are present for interactive interface.







The above screenshots show a part of the HTML code with its corresponding UI. This shows how user actions are processed within the system.

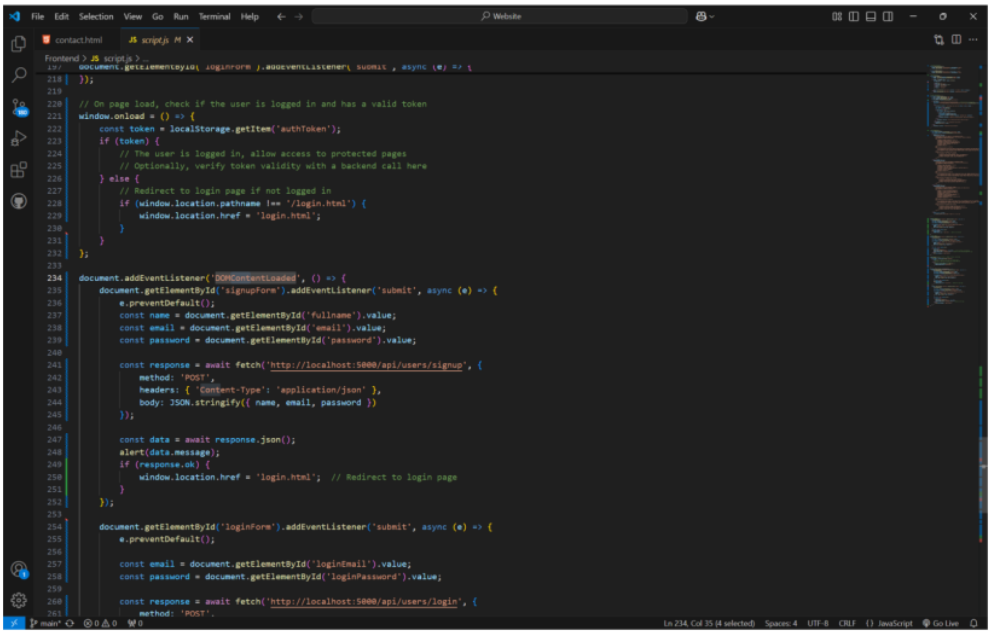


Figure 17: Screenshot of JavaScript code for the frontend operation in frontend/script.js

Even though the full JavaScript code is not present in the snippet above, this script.js handles the frontend logic for the website.

3.1.4 Local service testing

These screenshots below validate that service works as intended during local tests. The tests were done in the local server using node server.js.

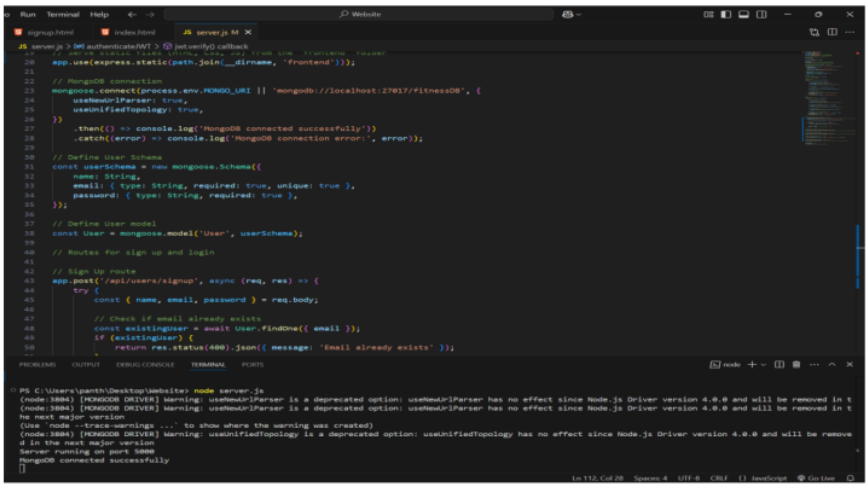


Figure 18: Screenshot of the terminal confirming database connection and server running

Home Page

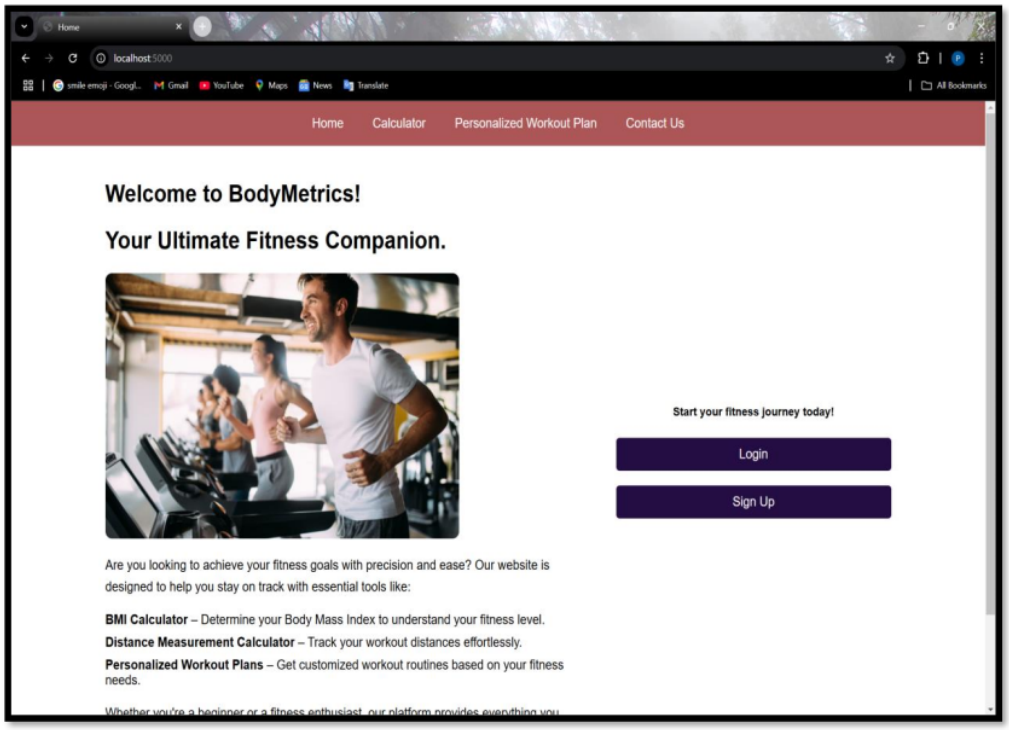


Figure 19: Screenshot of home page in localhost

Calculator page before login

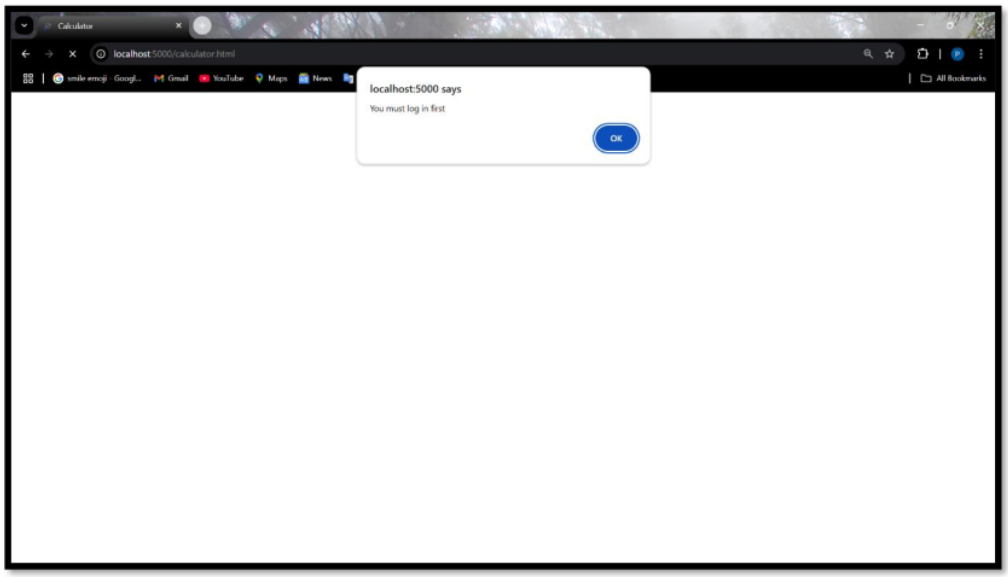


Figure 20: Screenshot of calculator page in localhost

Workout page before login

Figure 21: Screenshot of workout page in localhost

Sign up page

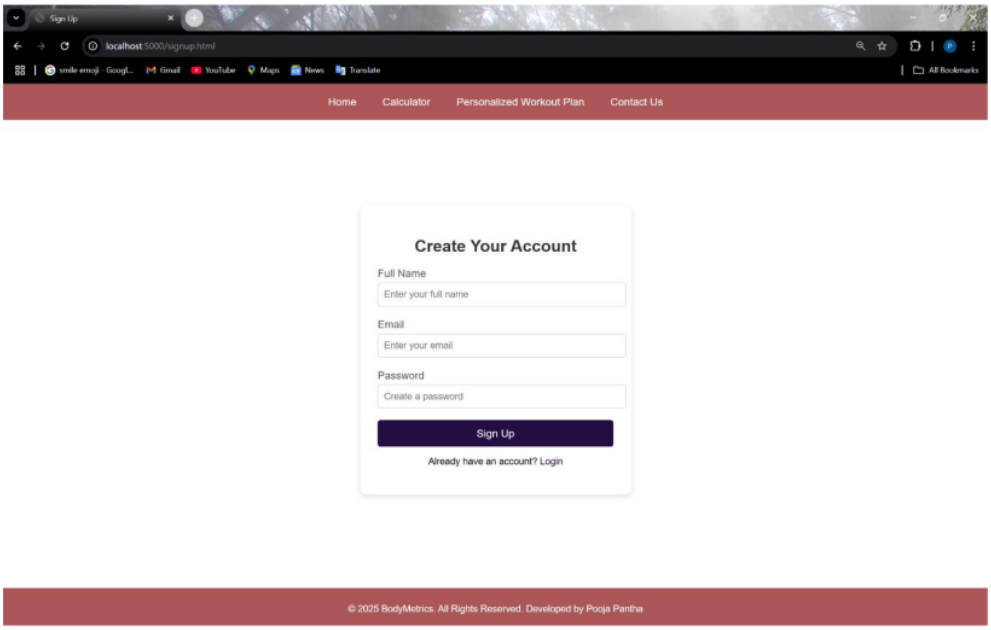


Figure 22: Screenshot of signup page in localhost

Filled up details for sign up

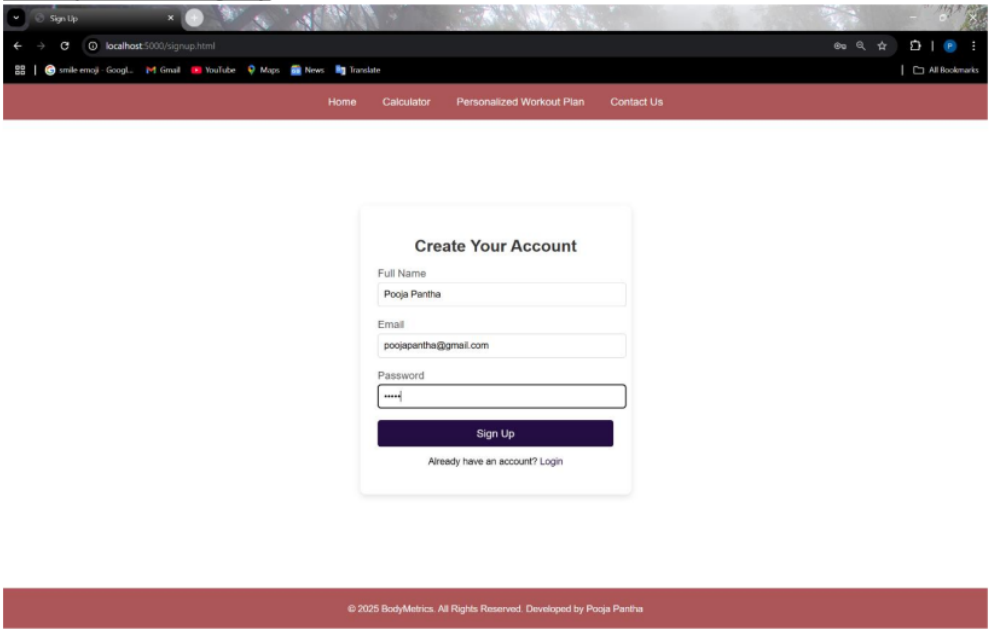


Figure 23: Screenshot of signup page in localhost

Signed up!

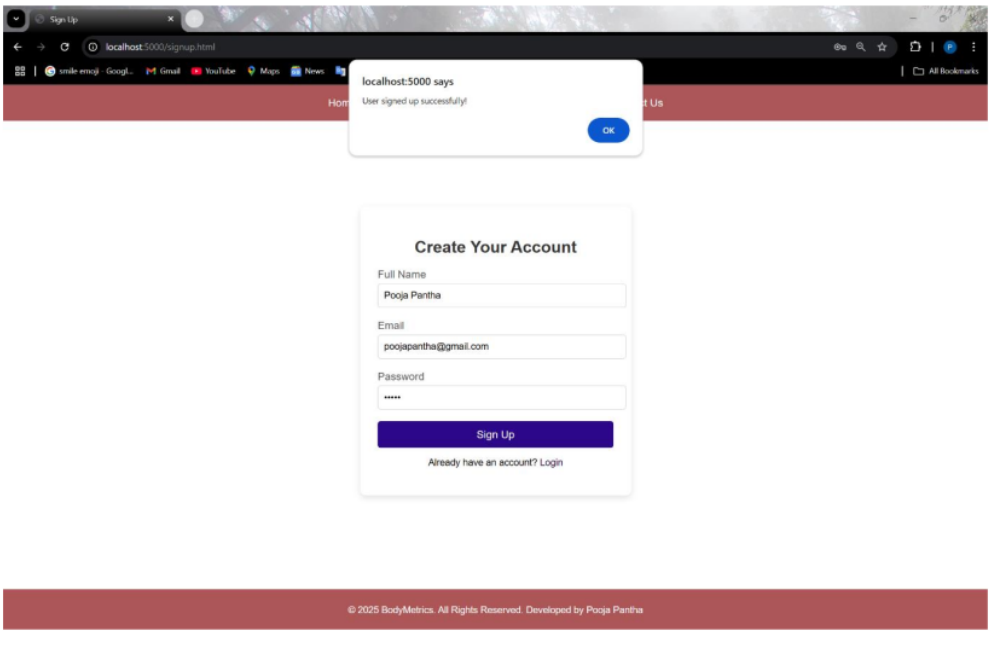


Figure 24: Screenshot of successful signup in localhost

Login page

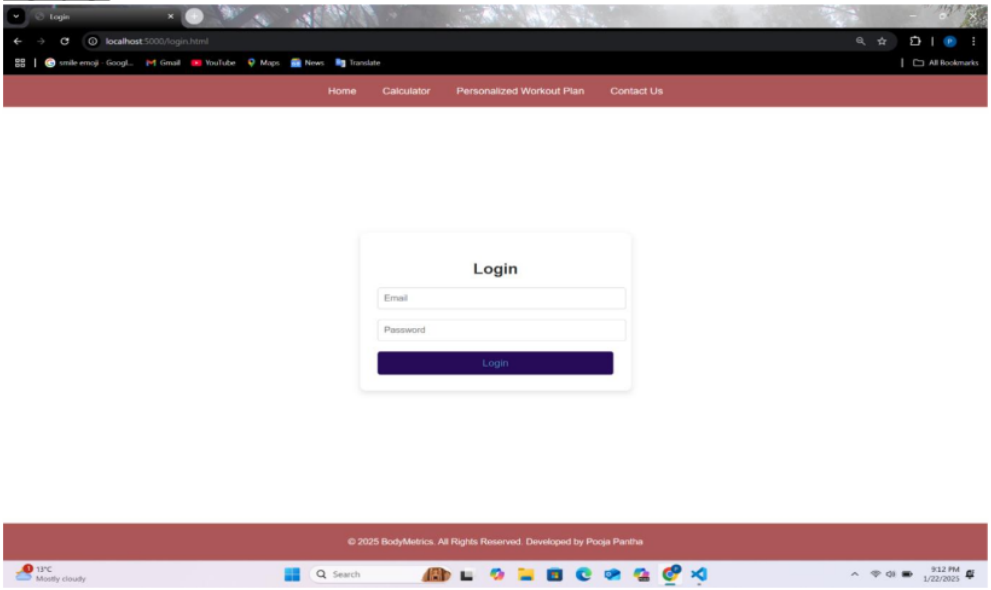


Figure 25: Screenshot of login page in localhost

Filled up details for log in

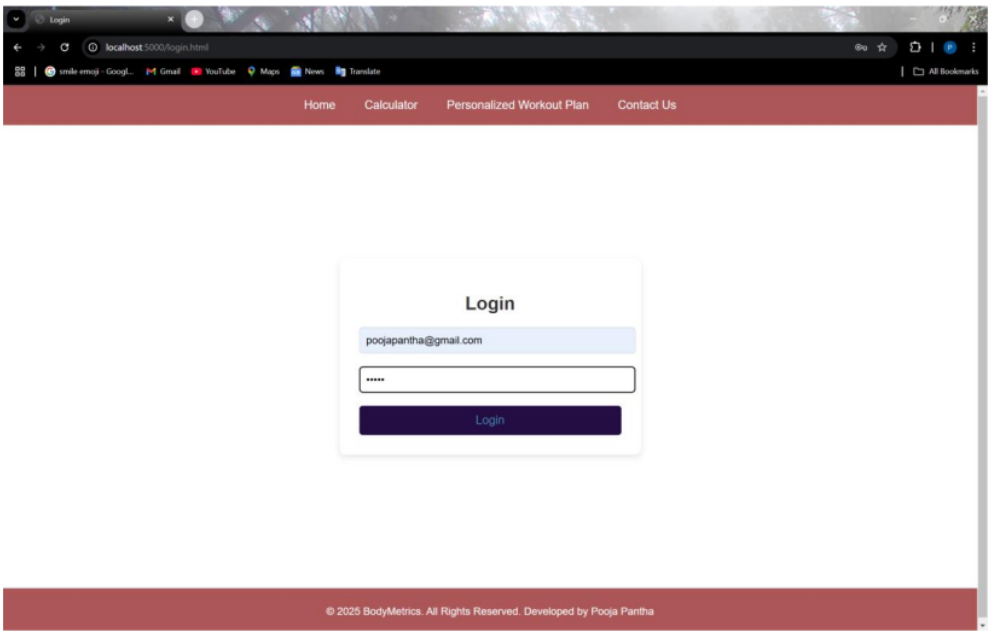


Figure 26: Screenshot of login page in localhost

After logging in, redirected to the calculator page

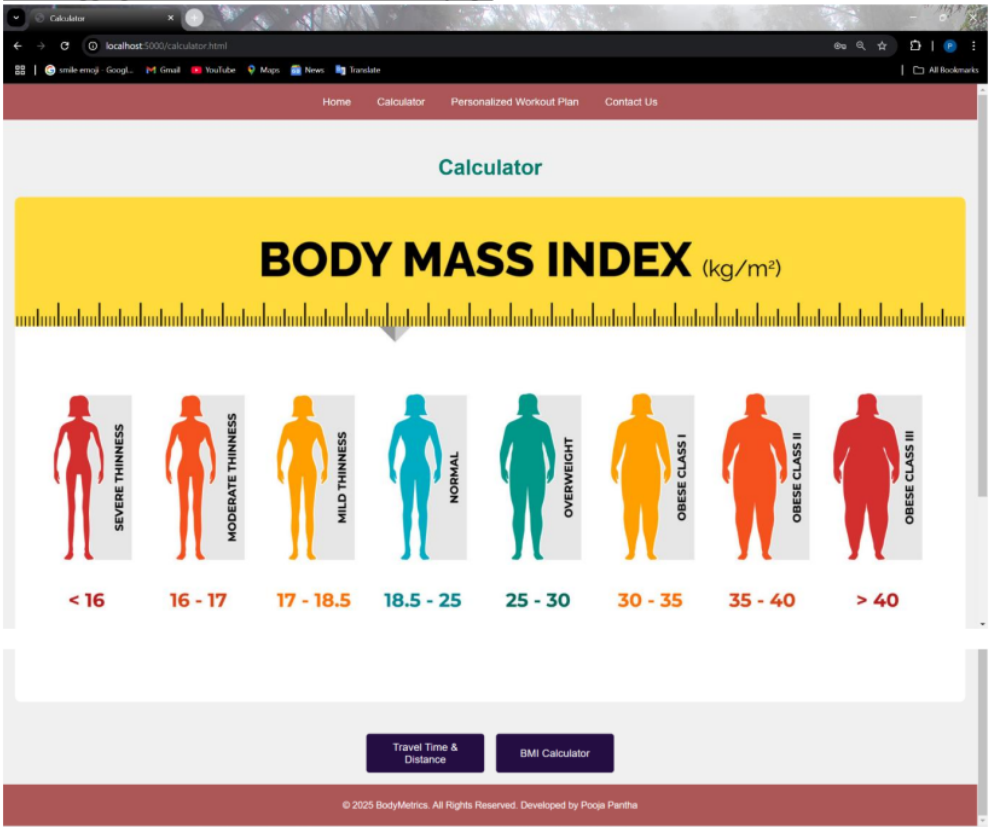


Figure 27: Screenshot of calculator page in localhost

BMI Calculator

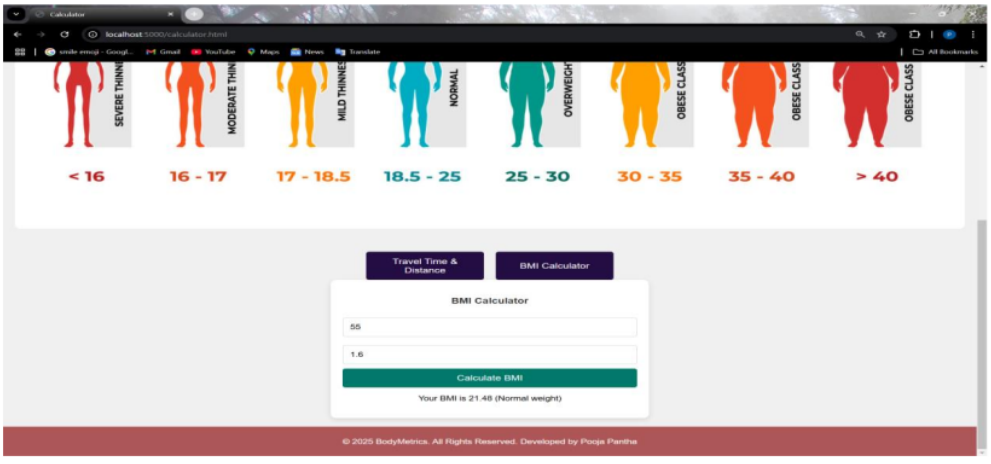


Figure 28: Screenshot of calculator page in localhost

Travel time & distance calculator with invalid location

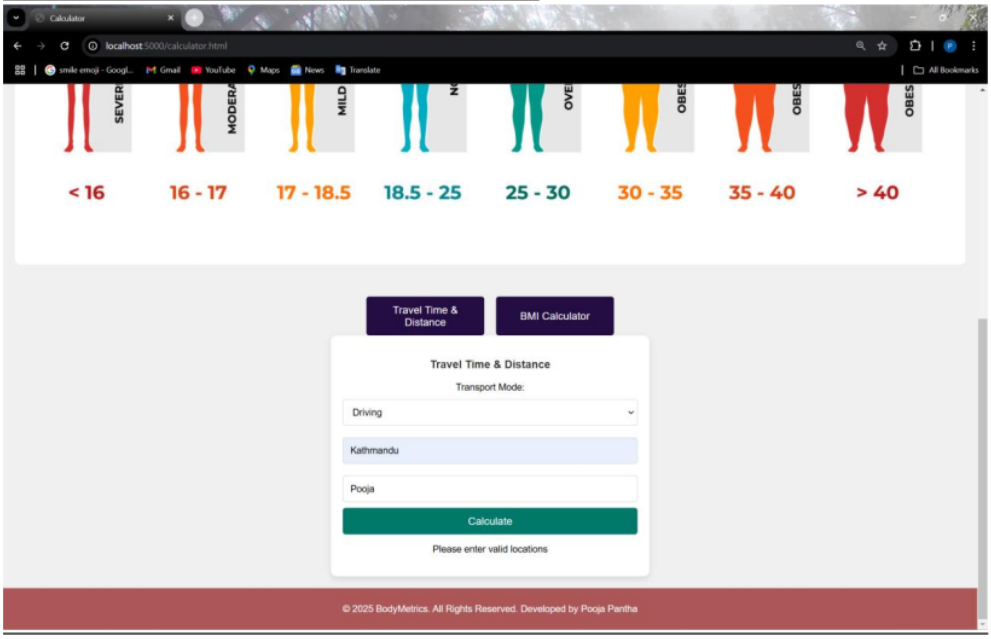


Figure 29: Screenshot of calculator page in localhost

Travel time & distance calculator with valid location

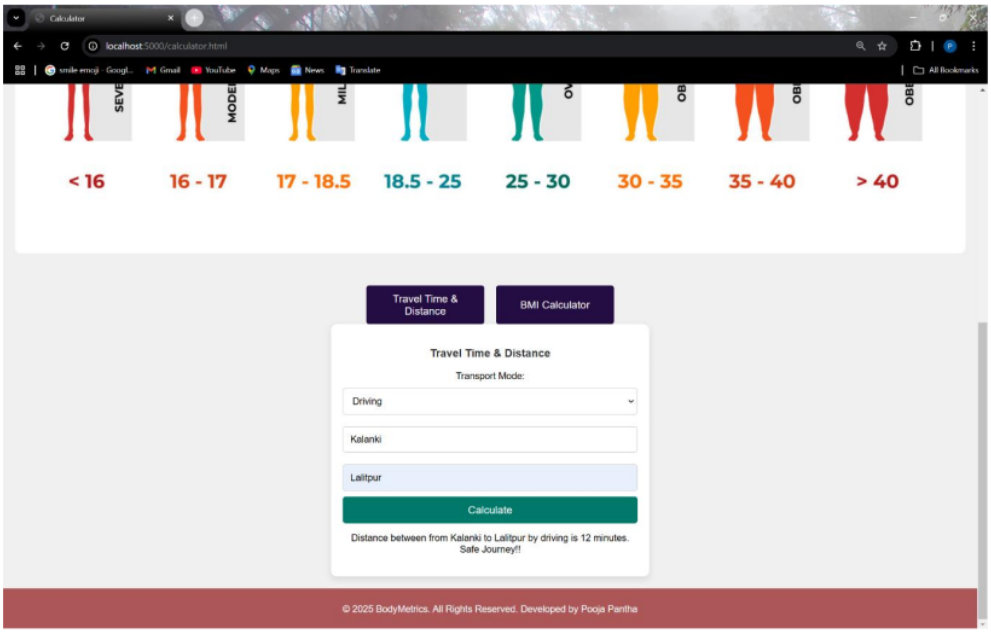


Figure 30: Screenshot of calculator page in localhost

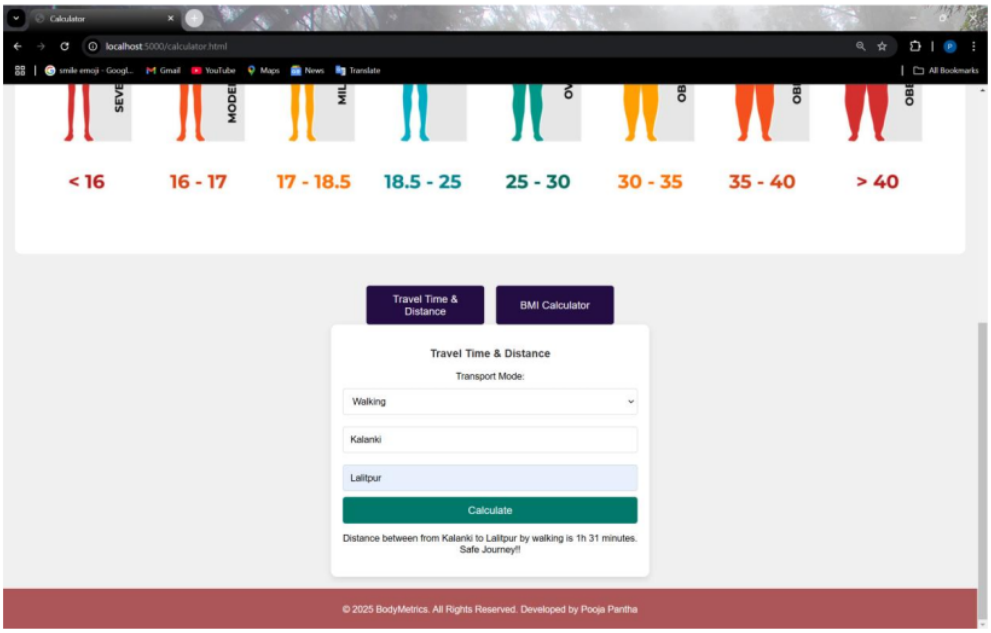


Figure 31: Screenshot of calculator page in localhost

Workout plan page

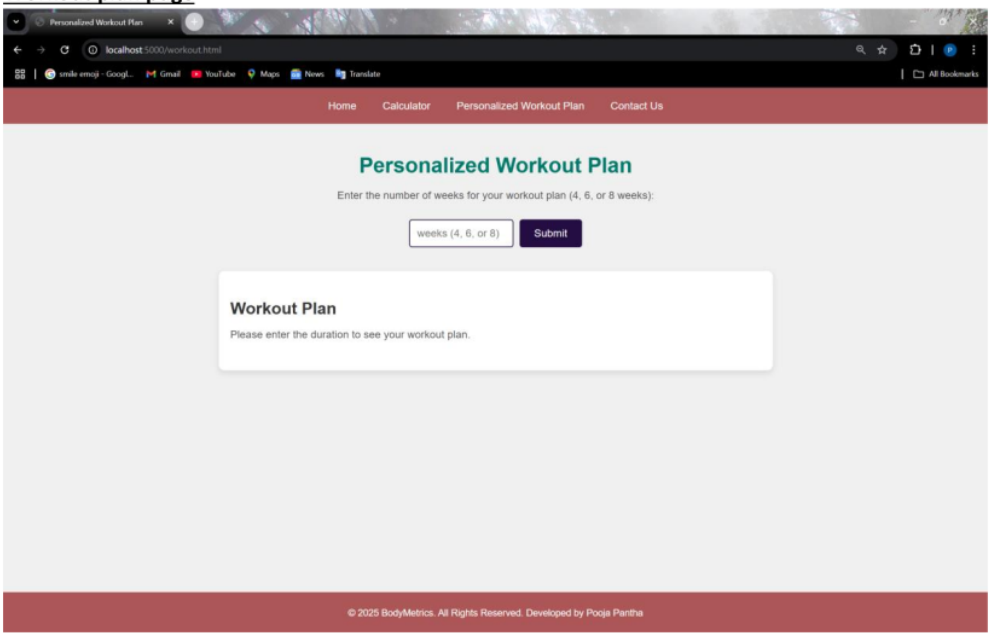


Figure 32: Screenshot of workout page in localhost

4 weeks plan

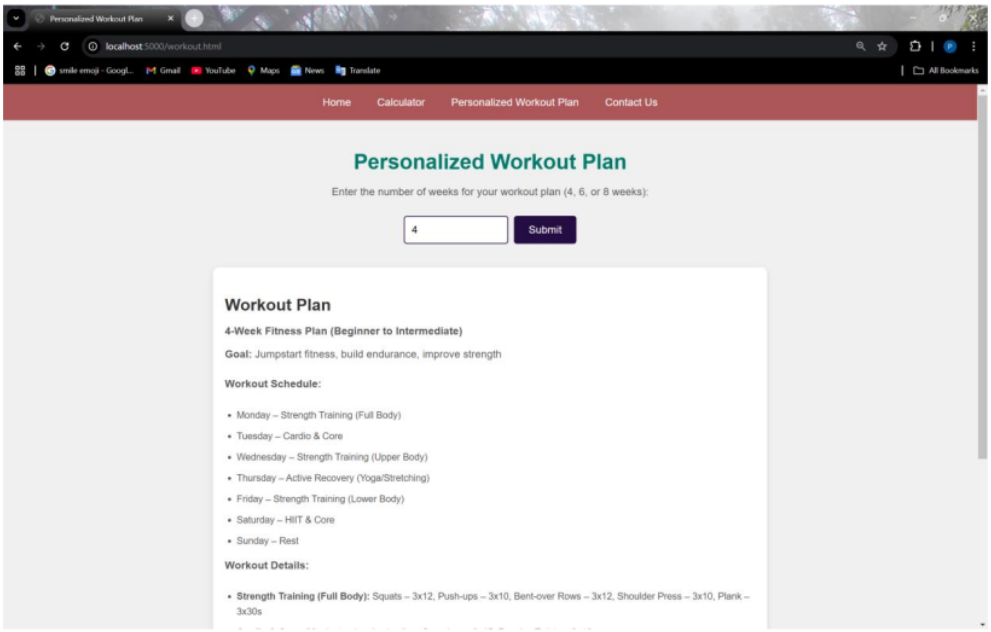


Figure 33: Screenshot of workout page in localhost

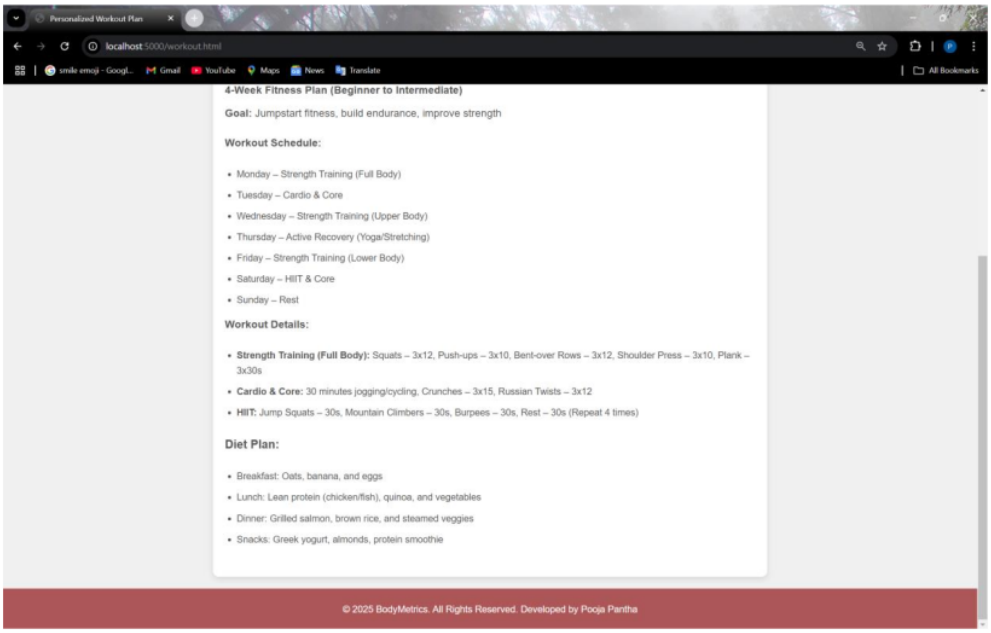


Figure 34: Screenshot of workout page in localhost

6 weeks plan

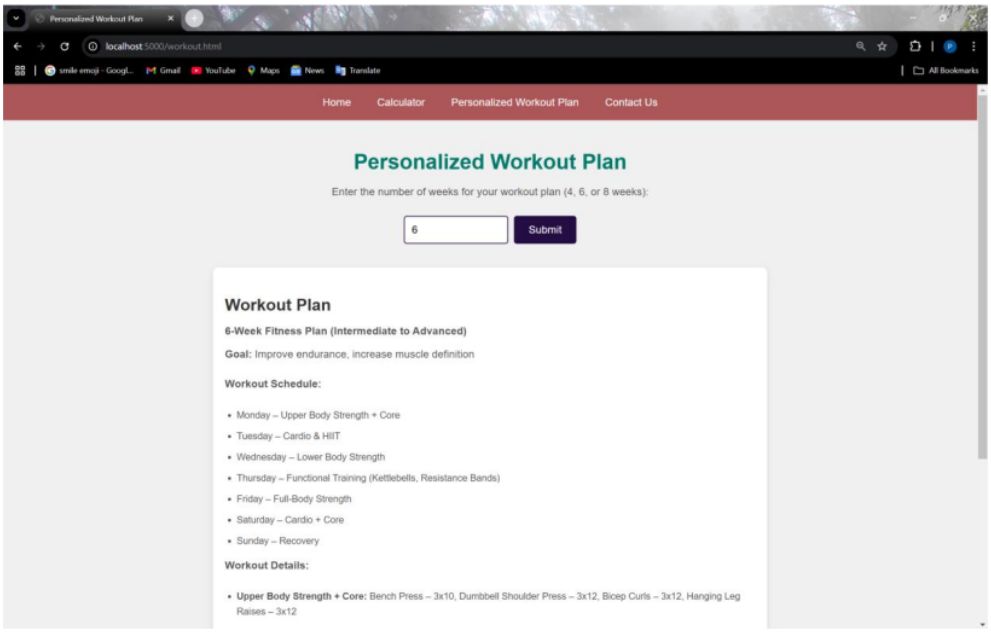


Figure 35: Screenshot of workout page in localhost

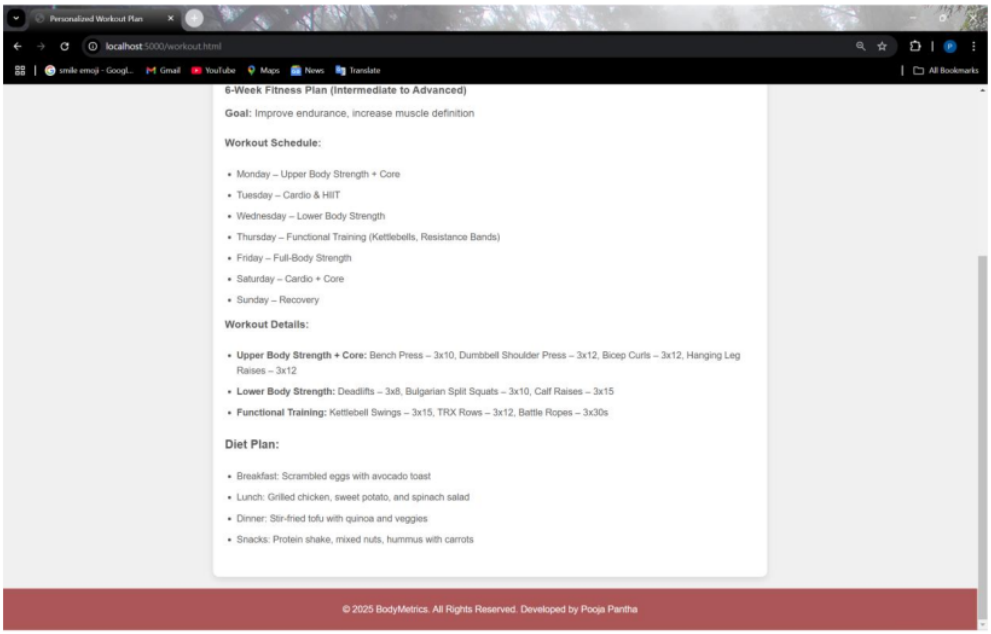


Figure 36: Screenshot of workout page in localhost

8 weeks plan

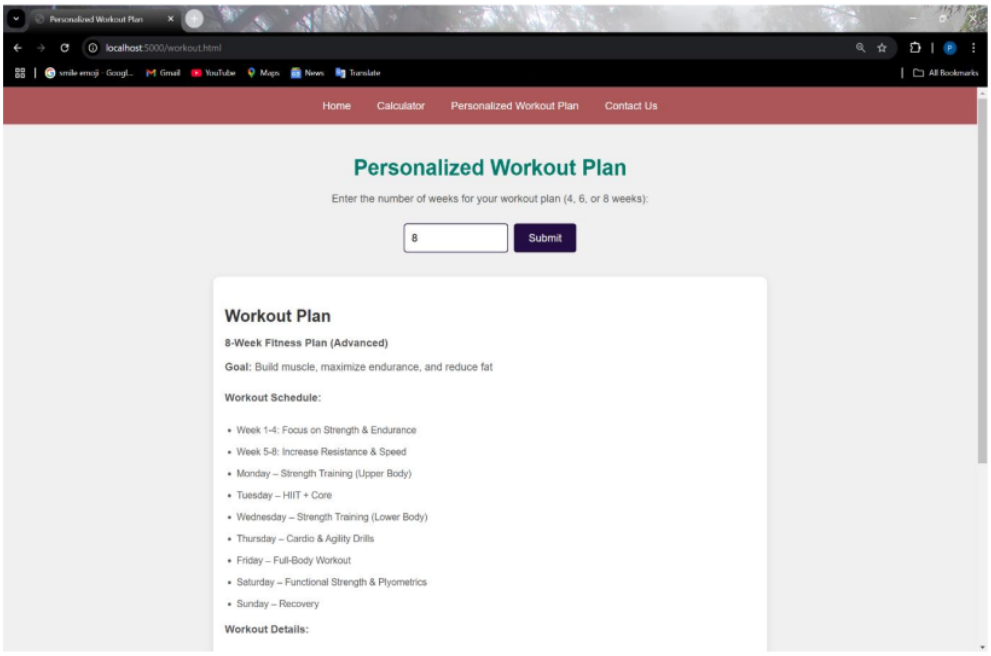


Figure 37: Screenshot of workout page in localhost

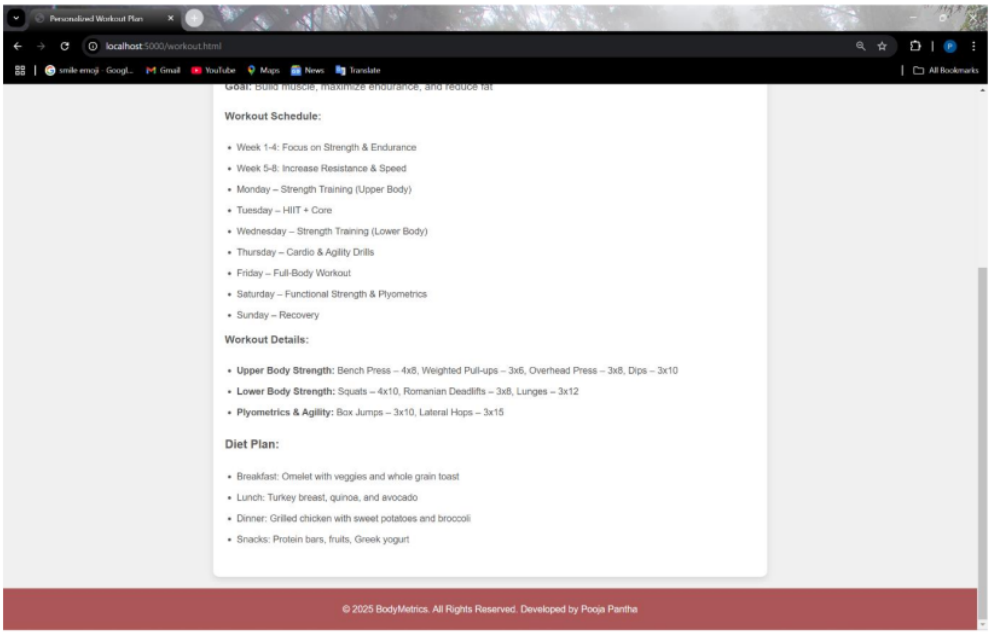


Figure 38: Screenshot of workout page in localhost

5

Contact us page

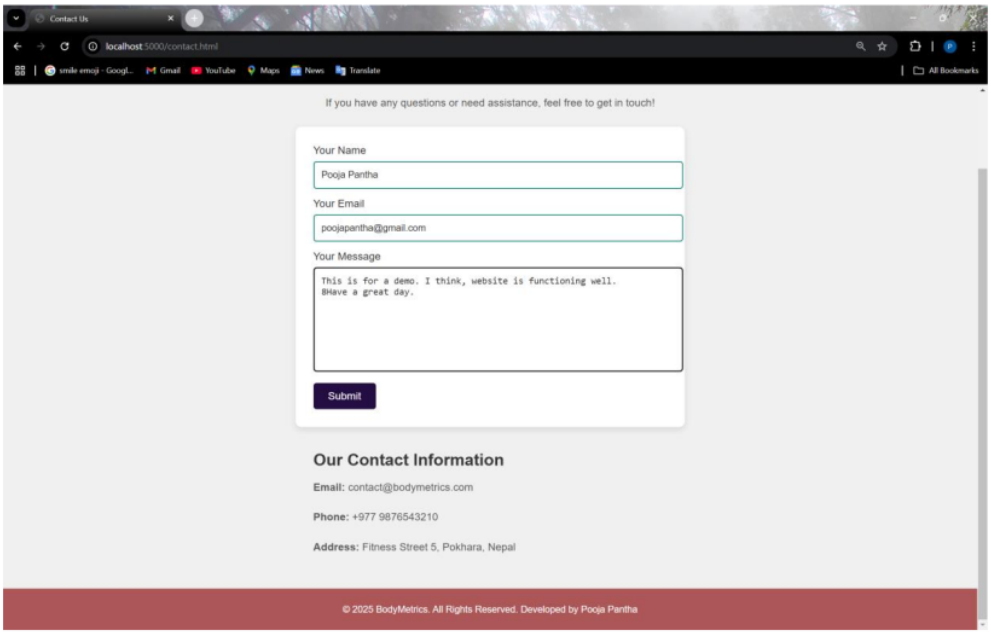


Figure 39: Screenshot of contact us page in localhost

The message, name and email address which is filled up in above contact form is stored in google sheets. The screenshot for the logic behind it is shared in the next section below.

After submitting contact us form

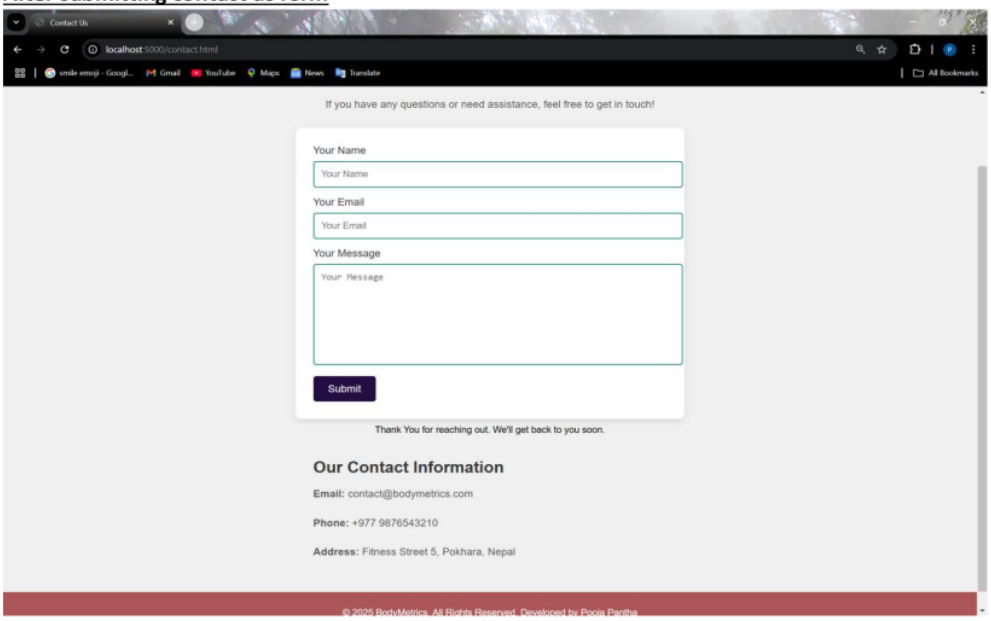


Figure 40: Screenshot of contact us page in localhost

When signing up with an existing email

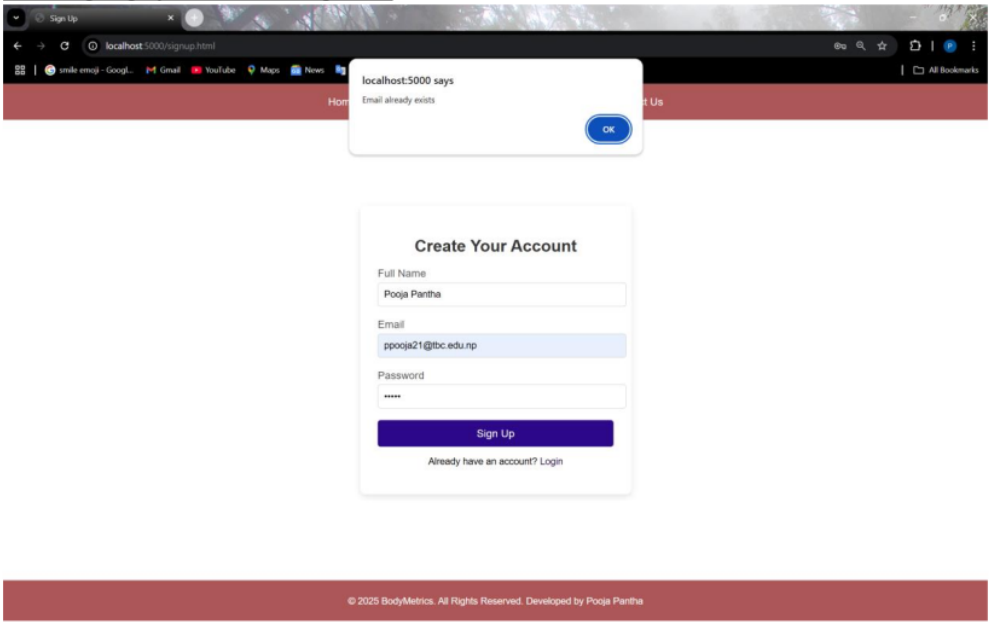


Figure 41: Screenshot of signup page in localhost

Backend data store and management

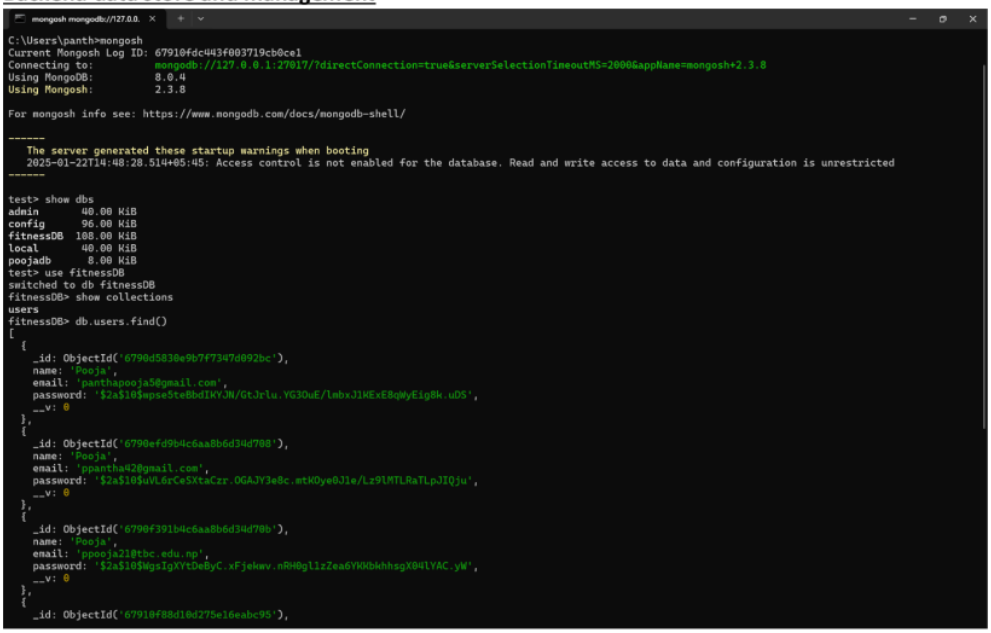


Figure 42: Screenshot of data stored in MongoDB

```
fitnessDB> db.users.find()
[
  {
    _id: ObjectId('6790d5830e9b7f7347d092bc'),
    name: 'Pooja',
    email: 'panthapooja5@gmail.com',
    password: '$2a$10$wpse5teBbdIKYJN/GtJrlu.YG3OuE/lmbxJ1KExE8qWyEig8k.uDS',
    __v: 0
  },
  {
    _id: ObjectId('6790efd9b4c6aa8b6d34d708'),
    name: 'Pooja',
    email: 'ppantha42@gmail.com',
    password: '$2a$10$uVL6rCeSXtaCzr.OGAJY3e8c.mtKOye0J1e/Lz9lMTLRaTLpJIQju',
    __v: 0
  },
  {
    _id: ObjectId('6790f391b4c6aa8b6d34d70b'),
    name: 'Pooja',
    email: 'ppooja21@tbc.edu.np',
    password: '$2a$10$WgsIgXYtDeByC.xFjekwv.nRH0gl1zZea6YKKbkhhsgX04lYAC.yW',
    __v: 0
  },
  {
    _id: ObjectId('67910f88d10d275e16eabc95'),
    name: 'Pawan',
    email: 'panth.pawan@gmail.com',
    password: '$2a$10$qkUxitZidacN2cFT9P1g2eh31Wi8VY/JCEDP6aXs7nCLuAWFYxmhS',
    __v: 0
  },
  {
    _id: ObjectId('679110afd10d275e16eabc98'),
    name: 'Pooja Pantha',
    email: 'poojapantha@gmail.com',
    password: '$2a$10$Ka.XNdINnOBZeFO4N9SPNejANu1728uIwEOiK2SG4jp6BaJfPf4Rm',
    __v: 0
  }
]
fitnessDB>
```

Figure 43: Screenshot of data stored in MongoDB

When contact us form is submitted it is stored in a google sheet for easy access to the admin unlike email, password. In above contact us form whatever it was submitted, it is stored here.

Note: First row of the data was a trial.

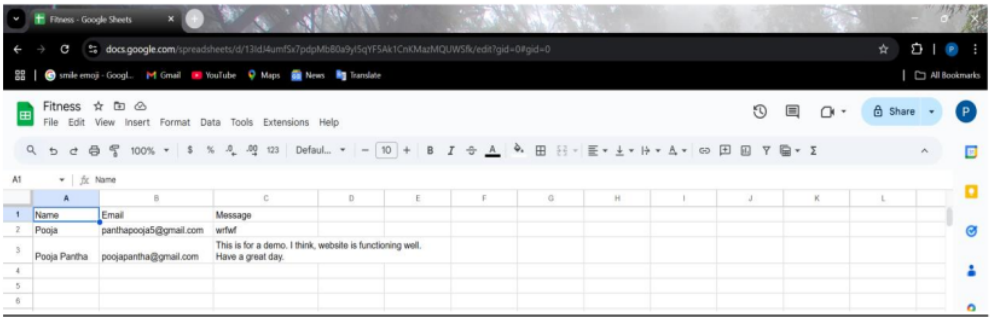


Figure 44: Screenshot of data stored in Google Sheets

3.2 Service Deployment with the Cloud Provider

The service is built for deployment through Amazon Web Services (AWS) EC2 to use IaaS capabilities. A step-by-step deployment guideline below will instruct the process of setting up instances before successfully operating the web application on the cloud environment:

Step 1: Navigated to EC2 Dashboard.

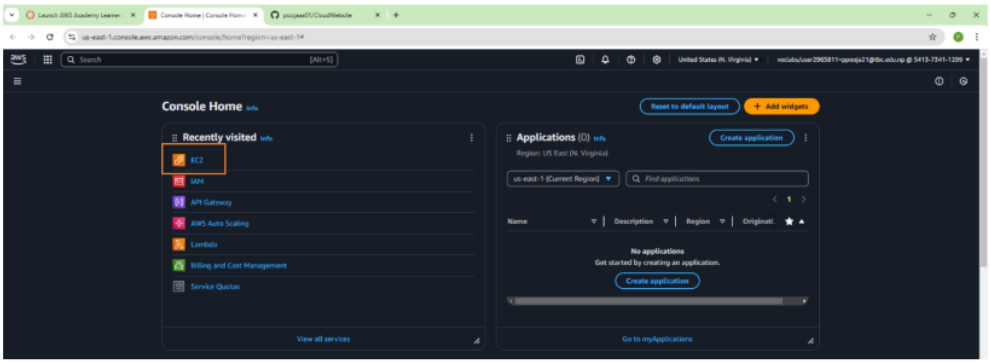


Figure 45: Screenshot of the EC2 instance creation

Step 2: Clicked Launch instance.

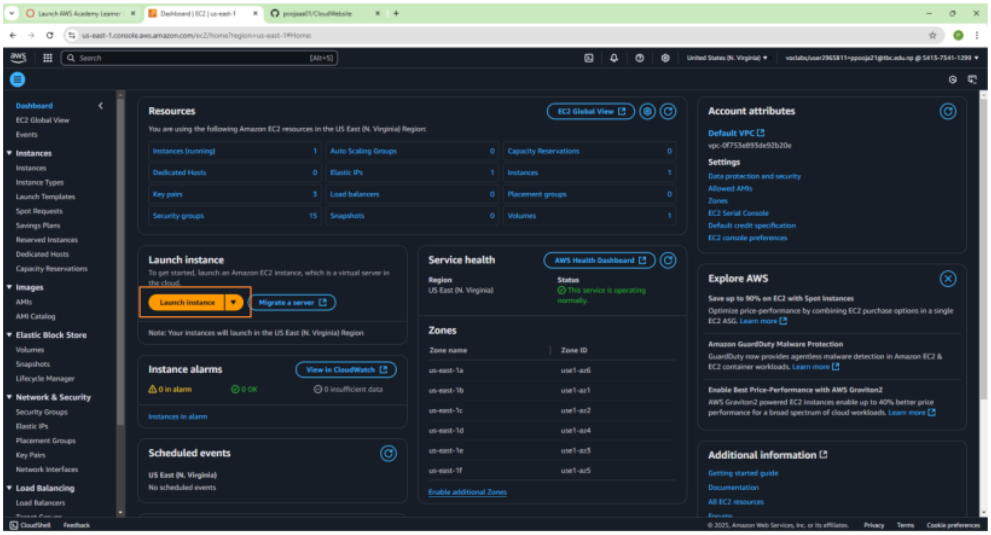


Figure 46: Screenshot of the EC2 instance creation

Step 3: Gave name to the server and selected Linux.

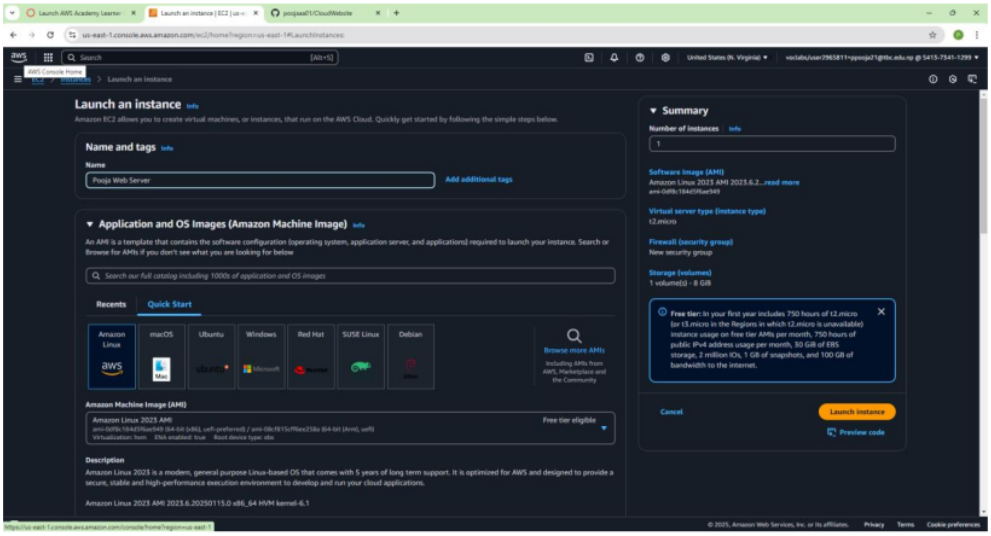


Figure 47: Screenshot of the EC2 instance creation

Step 4: Selected free tier instance type, key pair and customized network settings and launched the instance.

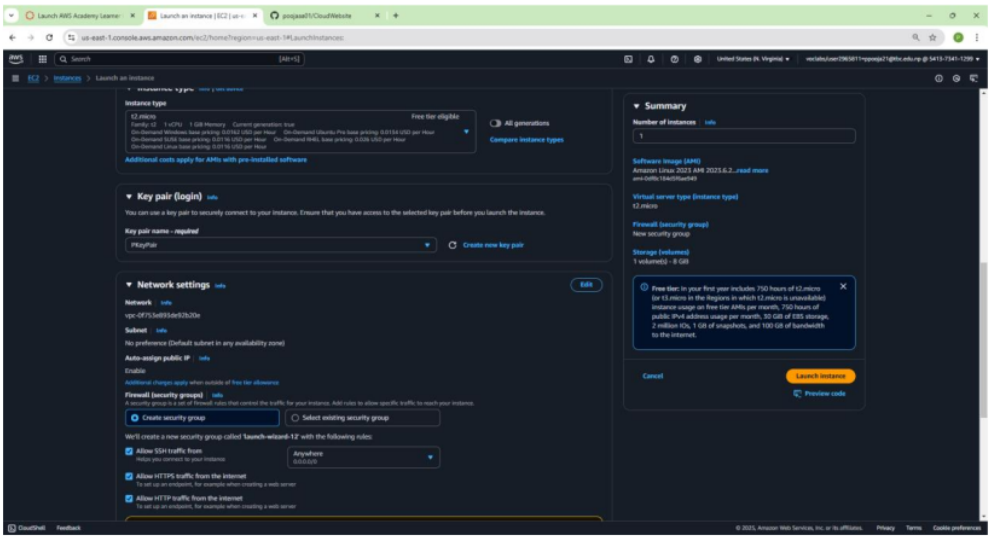


Figure 48: Screenshot of the EC2 instance creation

Step 5: Instance is launched. The most important part to note here is Public IPv4 address. And then connect.

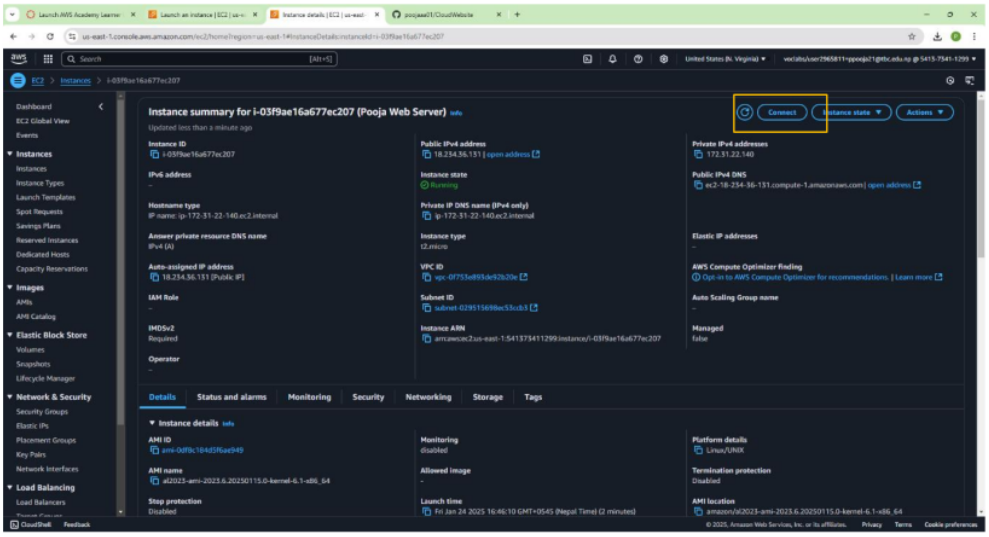


Figure 49: Instance Summary

Step 6: After pressing the connect button, Amazon Linux was connected.

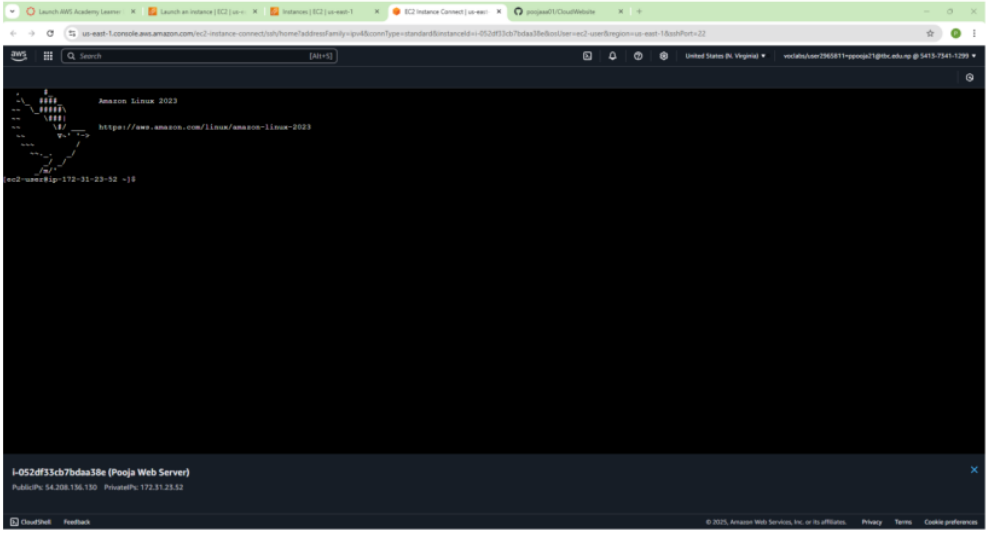


Figure 50: Connected to the Amazon Linux

Step 7: Passed these commands for the deployment

```
sudo su -
yum update -y
yum install -y httpd
systemctl status httpd
mkdir aws_assignment
cd aws_assignment
wget https://github.com/poojaaa01/CloudWebsite.git
ls -lrt
wget https://github.com/poojaaa01/CloudWebsite/archive/refs/heads/main.zip
ls -lrt
unzip main.zip
cd CloudWebsite-main
ls -lrt
mv * /var/www/html/
cd /var/www/html
systemctl enable httpd
systemctl start httpd
```

Figure 51: Commands for the connection and deployment

The initial step for this connection was committing code to a GitHub repository. The code deployment process started when integrated the GitHub link into the Amazon Linux server.

Step 8: Check the public IPv4 address which was in an instance details page.

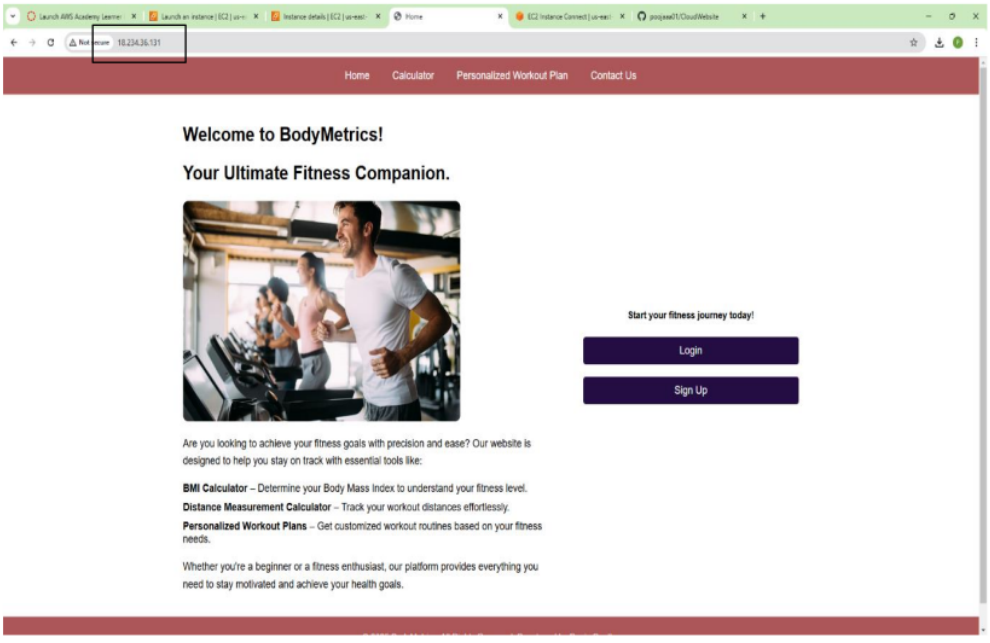


Figure 52: Home page in cloud server

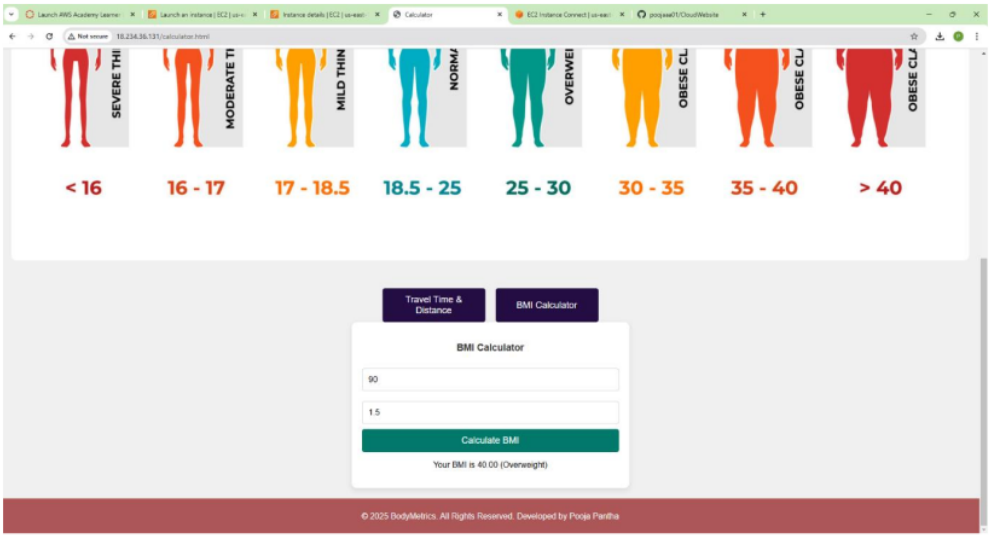


Figure 53: Calculator page with execution in cloud server

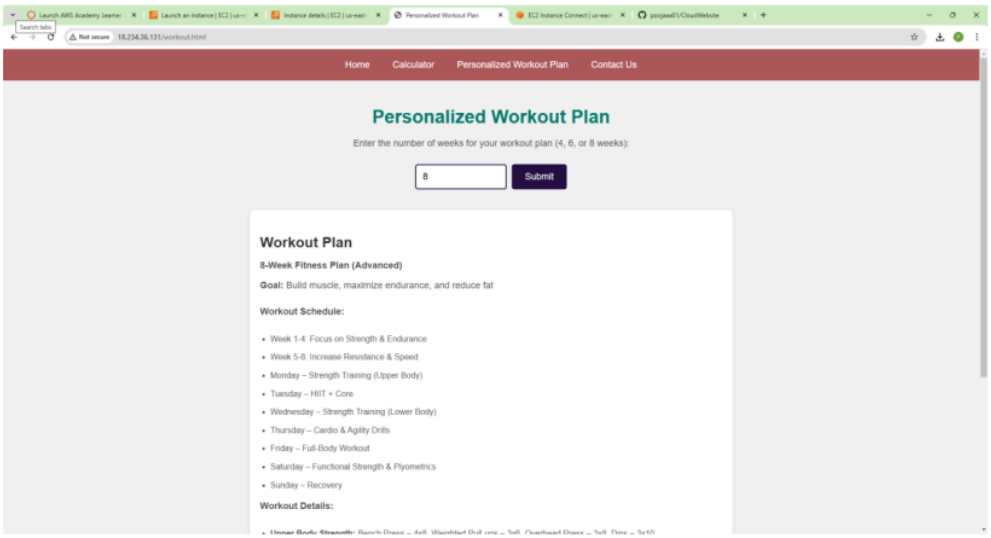


Figure 54: Workout page in cloud server

The deployed web application demonstrates successful execution of all the pages and features.

An error message is displayed when entering an invalid email address.

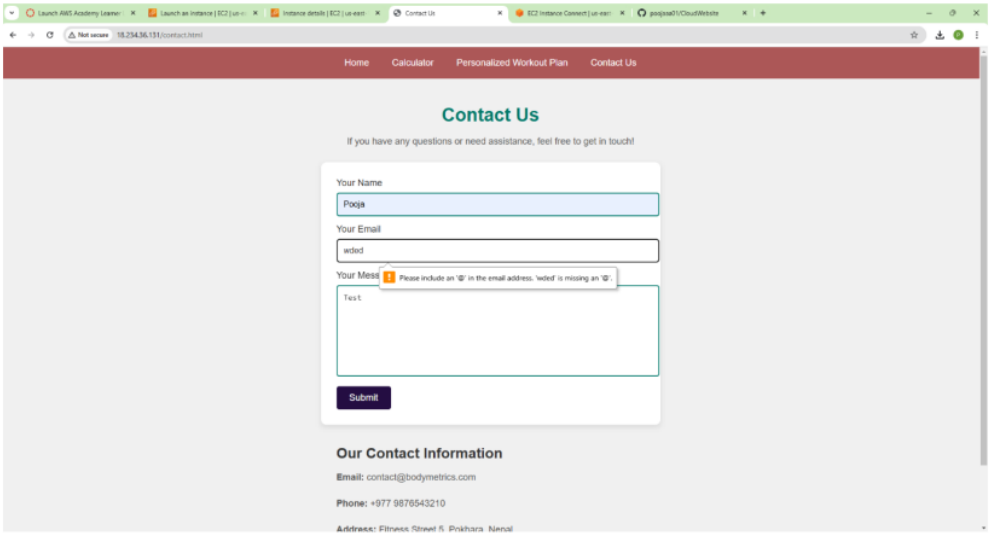


Figure 55: Contact Page with an error message in cloud server

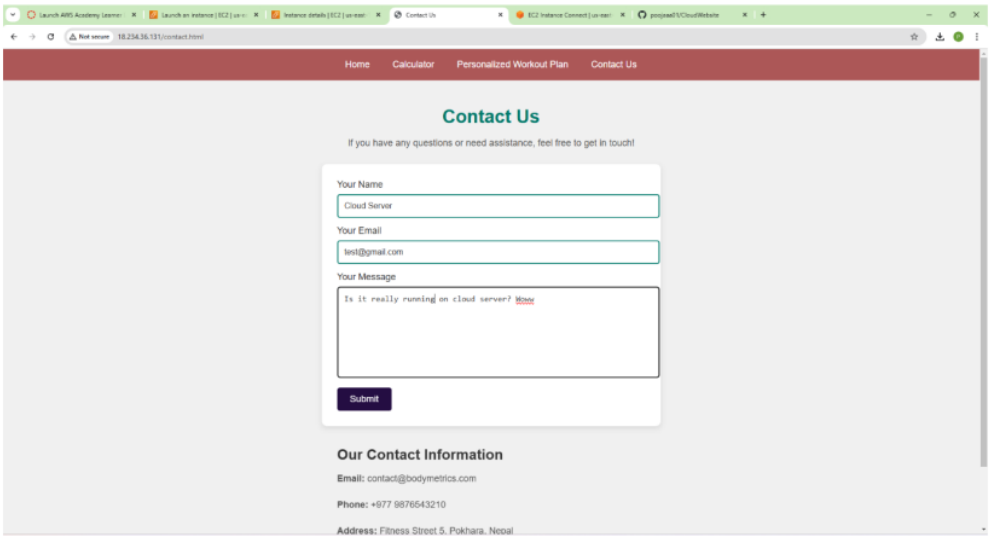


Figure 56: Contact page in cloud server

The web application deploys successfully on AWS EC2 by following these steps while maintaining user access to all the features of the website.

3.3 Service management

AWS delivers multiple observation interfaces to manage and track the deployed service operations on EC2 instances. The AWS Management Console creates a simple interface for users to trigger instant functions like starting and stopping and rebooting and terminating their instances (Amazon Web Services, 2025). This gives users total control over their service operations. The service monitoring capability of Amazon CloudWatch delivers real-time usage details about CPU performance and memory storage and disk events and network bandwidth information. A flexible alerting system, together with automated responses, helps users maintain service performance and ensure availability of resources through threshold-based actions (IBM, 2023). Service scaling becomes more efficient through Auto Scaling Groups that perform automatic schema adjustments based on traffic patterns. This results in optimized resource distribution. Through its System Manager interface AWS offers secure remote access services in combination with automated task processing which eliminates the need for manual SSH authentication (AWS User Guide, 2024). AWS services provide capabilities for both version-control management and distribution and failure-prevention functions during deployment operations. Through AWS management and monitoring solutions organizations benefit from systems which deliver consistent reliability alongside capabilities for further scalability and simplified maintenance practices.

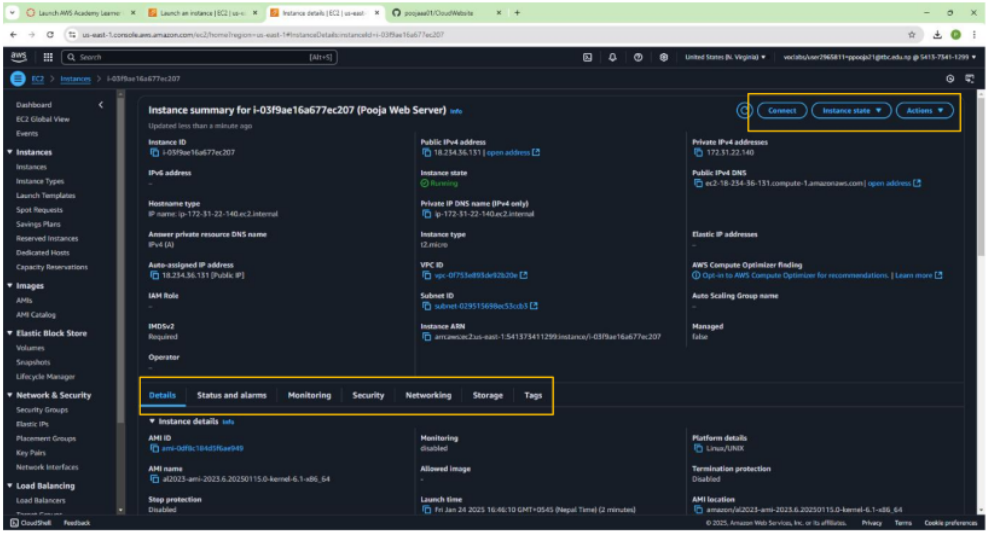


Figure 57:Instance Summary

Here in the instance summary page, multiple essential tools can be seen for efficient service management. Connect and Instance State within the EC2 dashboards allow managed control and secure SSH access to instances through various connection methods. Users can enhance their deployed service's management and reliability through monitoring tools combined with configuration options which help track performance metrics to make necessary changes.

4. Conclusion

This report provided an in-depth overview of the development and deployment of a web-based service utilizing Amazon Web Services (AWS) EC2, an Infrastructure as a Service (IaaS) model. Users can access platform features such as BMI calculator and distance measurement tools as well as personalized workout plans and a contact form. The website development combined HTML, CSS and

JavaScript for the frontend with Node.js and MongoDB for its database powered backend structure to support responsive interfaces. The deployment required creating an EC2 instance and implementing environmental configuration while establishing web-based access for users to use the service.

Through its extensive management and monitoring capability AWS allows users to monitor their services efficiently with AWS Management Console, CloudWatch, and Auto Scaling and maintain their operations smoothly. The available tools provide essential capabilities for maintaining application deployment reliability alongside maintaining superior performance levels along with reaching maximum scalability objectives.

In conclusion, application deployment strategies gain effectiveness from cloud technologies because these solutions deliver scalable infrastructure with affordable services and improved reliability features. Through AWS EC2 users can develop applications right away while their infrastructure management needs are handled directly by cloud providers. The research demonstrates how cloud solutions efficiently streamline operations and deliver reliable system performance.

5. References

Administrator (2024) Difference between IaaS, PaaS, and SaaS. Examples of Cloud Service Models | LitsLink Blog. [Online]. Available from: <https://litslink.com/blog/iaas-paas-saas> [Accessed 20 January 2025].

Alam, T. (2020) Cloud Computing and Its Role in the Information Technology. [Online]. Rochester, NY: papers.ssrn.com. Available from: <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3639063> [Accessed 19 January 2025].

Amazon Web Services (2024) Amazon EC2 [Online]. Amazon Web Services, Inc. Available from: <https://aws.amazon.com/ec2/> [Accessed 20 January 2025].

Amazon Web Services (2021) What is Cloud Computing? | Amazon Web Services. [Online video]. July 15. Available from: <https://www.youtube.com/watch?v=mxT233EdY5c> [Accessed 20 January 2025].

Amazon Web Services (n.d.) Stop and Start Your Instance - Amazon Elastic Compute Cloud [Online]. docs.aws.amazon.com. Available from: <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Stop_Start.html> [Accessed 20 January 2025].

AWS User Guide (2024) User Guide AWS Systems Manager. [Online]. Available from: <https://docs.aws.amazon.com/pdfs/systems-manager/latest/userguide/systems-manager-ug.pdf> [Accessed 24 January 2025].

Barney, N. (n.d.) What Is Oracle Cloud? | Definition from TechTarget. [Online]. Available from: <https://www.techtarget.com/whatis/definition/Oracle-Cloud> [Accessed 22 January 2025].

Cloudflare (n.d.) What Is Hybrid Cloud? | Hybrid Cloud Definition. [Online]. Available from: <https://www.cloudflare.com/en-gb/learning/cloud/what-is-hybrid-cloud/> [Accessed 20 January 2025].

freeCodeCamp.org (2020) NodeJS vs Python: How to Choose the Best Technology to Develop Your Web App's Back End. [Online]. Available from: <https://www.freecodecamp.org/news/nodejs-vs-python-choosing-the-best-technology-to-develop-back-end-of-your-web-app/> [Accessed 22 January 2025].

GeeksforGeeks (2023) Cloud Deployment Models. [Online]. Available from: <https://www.geeksforgeeks.org/cloud-deployment-models/> [Accessed 22 January 2025].

GeeksforGeeks (2024) Top 10 Cloud Platform Service Providers in 2024. [Online]. Available from: <https://www.geeksforgeeks.org/top-cloud-platform-service-providers/> [Accessed 20 January 2025].

Google Cloud (n.d.) What Is a Cloud Service Provider? [Online]. Available from: <https://cloud.google.com/learn/what-is-a-cloud-service-provider> [Accessed 20 January 2025].

Google Cloud (n.d.) What Is IaaS (Infrastructure as a Service)? [Online]. Available from: <https://cloud.google.com/learn/what-is-iaas?hl=en> [Accessed 20 January 2025].

Gupta, P. (2013) A Technical Seminar Report on Cloud Computing. [Online]. Available from: <https://www.slideshare.net/slideshow/report-on-cloud-computing-by-prashant-gupta/24192696> [Accessed 19 January 2025].

IBM (2023) Infrastructure Monitoring. [Online]. Available from: <https://www.ibm.com/think/topics/infrastructure-monitoring> [Accessed 24 January 2025].

JavatPoint (n.d.) Cloud Service Provider Companies. [Online]. Available from: <https://www.javatpoint.com/cloud-service-provider-companies> [Accessed 20 January 2025].

Pandey, H. (2018) Cloud Based Services. [Online]. Available from: <https://www.geeksforgeeks.org/cloud-based-services/> [Accessed 24 January 2025].

Patel, H. B. (2021) Cloud Computing Deployment Models: A Comparative Study. [Online]. Available from: <https://www.researchgate.net/publication/350721171_Cloud_Computing_Deployment_Models_A_Comparative_Study>.

Scribd (n.d.) *Cloud Computing.* [Online]. Available from: <https://www.scribd.com/document/487483812/Cloud-computing-Wikipedia> [Accessed 21 January 2025].

Simplilearn (2014) Essential Characteristics of Cloud Computing | Exin Cloud Computing. [Online video]. February 7. Available from: <https://www.youtube.com/watch?v=FDMrqES3pwo> [Accessed 20 January 2025].

Yasar, K. and Bigelow, S.J. (2022) What Is Microsoft Azure and How Does It Work? [Online]. Available from: <https://www.techtarget.com/searchcloudcomputing/definition/Windows-Azure> [Accessed 22 January 2025].

CCD-Report-PoojaPantha-77356785.docx

ORIGINALITY REPORT

16%	9%	4%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Leeds Beckett University Student Paper	4%
2	www.coursehero.com Internet Source	2%
3	Submitted to PSB Academy (ACP eSolutions) Student Paper	2%
4	Submitted to Divine Word Univresity Student Paper	1%
5	Submitted to University of Wolverhampton Student Paper	1%
6	Submitted to University of Maryland, Global Campus Student Paper	1%
7	www.ijastems.org Internet Source	<1%
8	w3cschoool.com Internet Source	<1%
9	www.mobt3ath.com Internet Source	<1%

10	Submitted to Manipal University Student Paper	<1 %
11	Submitted to i-CATS University College Student Paper	<1 %
12	Submitted to The British College Student Paper	<1 %
13	Submitted to Shinas College of Technology Student Paper	<1 %
14	Submitted to Info Myanmar College Student Paper	<1 %
15	ijarse.com Internet Source	<1 %
16	ebookreading.net Internet Source	<1 %
17	Submitted to Webster University Student Paper	<1 %
18	prosimo.io Internet Source	<1 %
19	qspace.qu.edu.qa Internet Source	<1 %
20	technology.eurekajournals.com Internet Source	<1 %
21	virtualizationreview.com Internet Source	<1 %

22	Submitted to Southampton Solent University Student Paper	<1%
23	Submitted to University of Melbourne Student Paper	<1%
24	blog.teamascend.com Internet Source	<1%
25	www.heart.org Internet Source	<1%
26	Submitted to Liverpool John Moores University Student Paper	<1%
27	Naresh Kumar Sehgal, Manoj Saxena, Dhaval N. Shah. "AI on the Edge with Security", Springer Science and Business Media LLC, 2025 Publication	<1%
28	John R. Vacca. "Cloud Computing Security - Foundations and Challenges", CRC Press, 2020 Publication	<1%
29	Debashis De. "Mobile Cloud Computing - Architectures, Algorithms and Applications", Chapman and Hall/CRC, 2019 Publication	<1%
30	K. Chandrasekaran. "Essentials of Cloud Computing", Chapman and Hall/CRC, 2019	<1%

Publication

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		