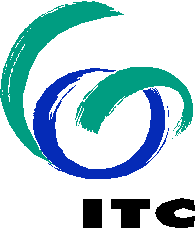
**GeoSearch: A Location-Based Product Search and Store Navigation Web Application**

**Integrated Geospatial Workflows Module -7 Project**

**Submitted by: Noor Ladkhan (Post Graduate Diploma 2022-2023)**



FACULTY OF GEO-INFORMATION SCIENCE AND EARTH OBSERVATION

UNIVERSITY OF TWENTE, ENSCHEDE, THE NETHERLANDS

***iirs***

INDIAN INSTITUTE OF REMOTE SENSING,

INDIAN SPACE RESEARCH ORGANISATION, DEHRADUN, INDIA

**Contents**

[1. Introduction 3](#_Toc138936303)

[1.1. Motivation 3](#_Toc138936304)

[1.2. Background 3](#_Toc138936305)

[2. Problem Statement 4](#_Toc138936306)

[2.1. Aim and Objectives 4](#_Toc138936307)

[3. Methodology 5](#_Toc138936308)

[3.1. Methodological Flowchart 5](#_Toc138936309)

[3.2. BPMN Diagram 9](#_Toc138936310)

[4. Softwares and tools used 11](#_Toc138936311)

[4.1. HTML and CSS 11](#_Toc138936312)

[4.2. PostgreSQL for Database 11](#_Toc138936313)

[4.3. Node.js 11](#_Toc138936314)

[4.4. Leaflet.js libraries for Map Visualization 12](#_Toc138936315)

[5. Results and Discussion 13](#_Toc138936316)

[6. References 14](#_Toc138936317)

# Introduction

## Motivation

The motivation behind this project stems from the need to address the challenges faced by consumers in finding specific products efficiently. Traditional methods of product search often involve browsing through numerous websites or physically visiting multiple stores, leading to time-consuming and sometimes frustrating experiences. Additionally, the lack of centralized platforms makes it difficult for users to compare prices and availability across different stores.

The proposed web application aims to alleviate these challenges by providing a convenient solution that integrates product search, store location visualization, and routing functionalities. By leveraging location-based services, users can effortlessly search for desired products and receive real-time information on the stores where those products are available. This streamlines the shopping experience and empowers users with the ability to make informed decisions regarding their purchases.

Furthermore, this web application also benefits store owners by offering them a platform to showcase their products and attract potential customers. By allowing store owners to register and update their product information, the application creates an ecosystem where both consumers and businesses can thrive.

## Background

Location-based services (LBS) have emerged as a transformative technology, revolutionizing various industries and enhancing user experiences. LBS utilizes geographical information to provide personalized services and tailored content based on a user's current or specified location. The proliferation of smartphones, coupled with advancements in mapping technologies and the availability of location data, has propelled the adoption of LBS in diverse domains.

In the context of the proposed web application, LBS plays a crucial role in delivering accurate and location-specific information to users. By leveraging LBS capabilities, the application can determine the user's current location or allow them to input a specific location of interest. This information serves as the foundation for product search, enabling the system to provide relevant results based on the user's preferences and proximity to stores.

Furthermore, the integration of mapping services in the web application, such as the utilization of Leaflet.js libraries for visualization, empowers users with an intuitive and interactive map interface. This interface allows users to visualize store locations, view additional information, and obtain routing directions from their current location to the selected store.

Overall, by harnessing the power of LBS and combining it with product search, store visualization, and routing capabilities, the proposed web application aims to enhance the efficiency and convenience of finding and purchasing desired products while improving the visibility and reach of stores.

# Problem Statement

The current problem addressed by this project lies in the lack of a centralized platform that offers comprehensive product search and store location visualization. Existing solutions often require users to navigate through multiple websites or physically visit various stores to find desired products. This project aims to overcome these limitations by providing a web application that combines product search functionality, store location visualization on a map, and routing capabilities.

## Aim and Objectives

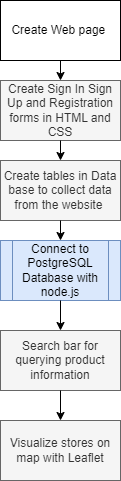
The aim of this project is to develop a web application that allows users to search for products based on their interests and visualize the locations of stores where the products are available. The specific objectives include:

* Designing an intuitive user interface that enables users to interact with the application seamlessly.
* Implementing an efficient search function that retrieves relevant products based on user input.
* Integrating a mapping service to visualize store locations on a dynamic map interface.
* Enabling user registration and providing store owners with the ability to update product data.
* Implementing routing and navigation functionality to guide users from their location to the selected store

# Methodology

The development of the web application followed a systematic methodology to ensure effective implementation of the proposed functionalities

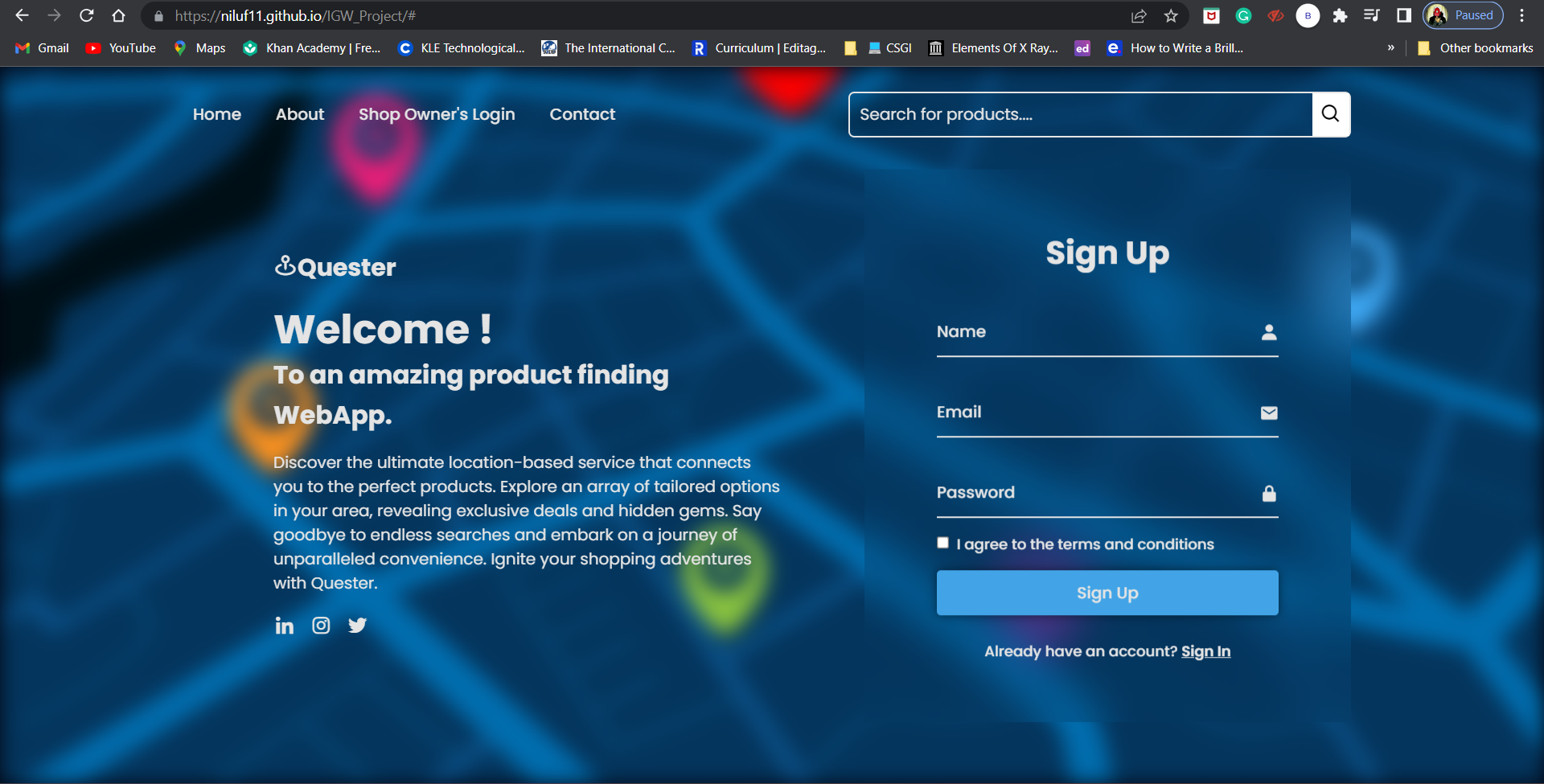
## Methodological Flowchart



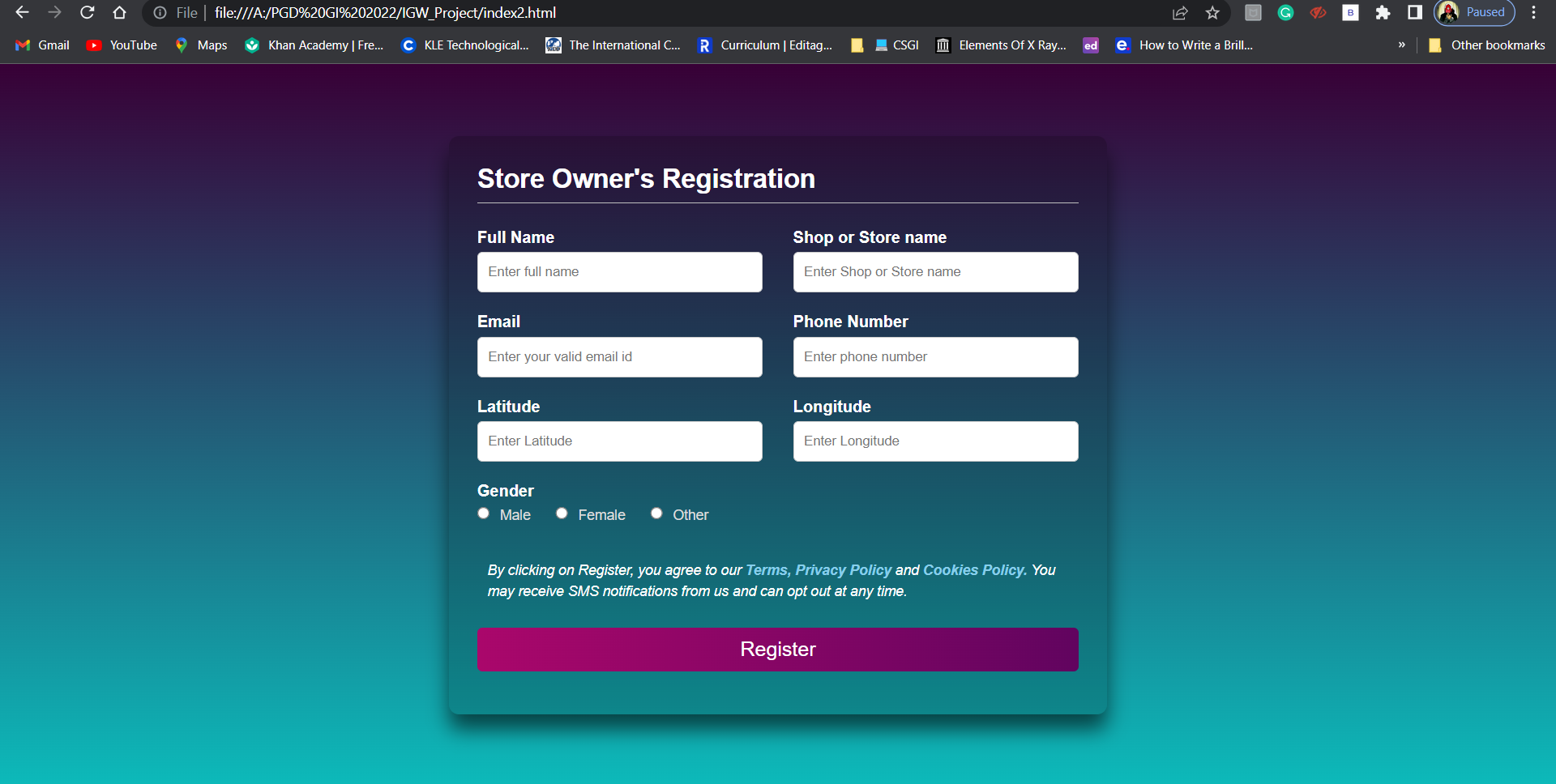
*Fig.1: Methodological flowchart*

A flowchart was designed to illustrate the overall flow of the web application, outlining the major steps and interactions between components. The flowchart depicted the sequence of actions, user interactions, and system responses, providing a comprehensive overview of the application's functionality.

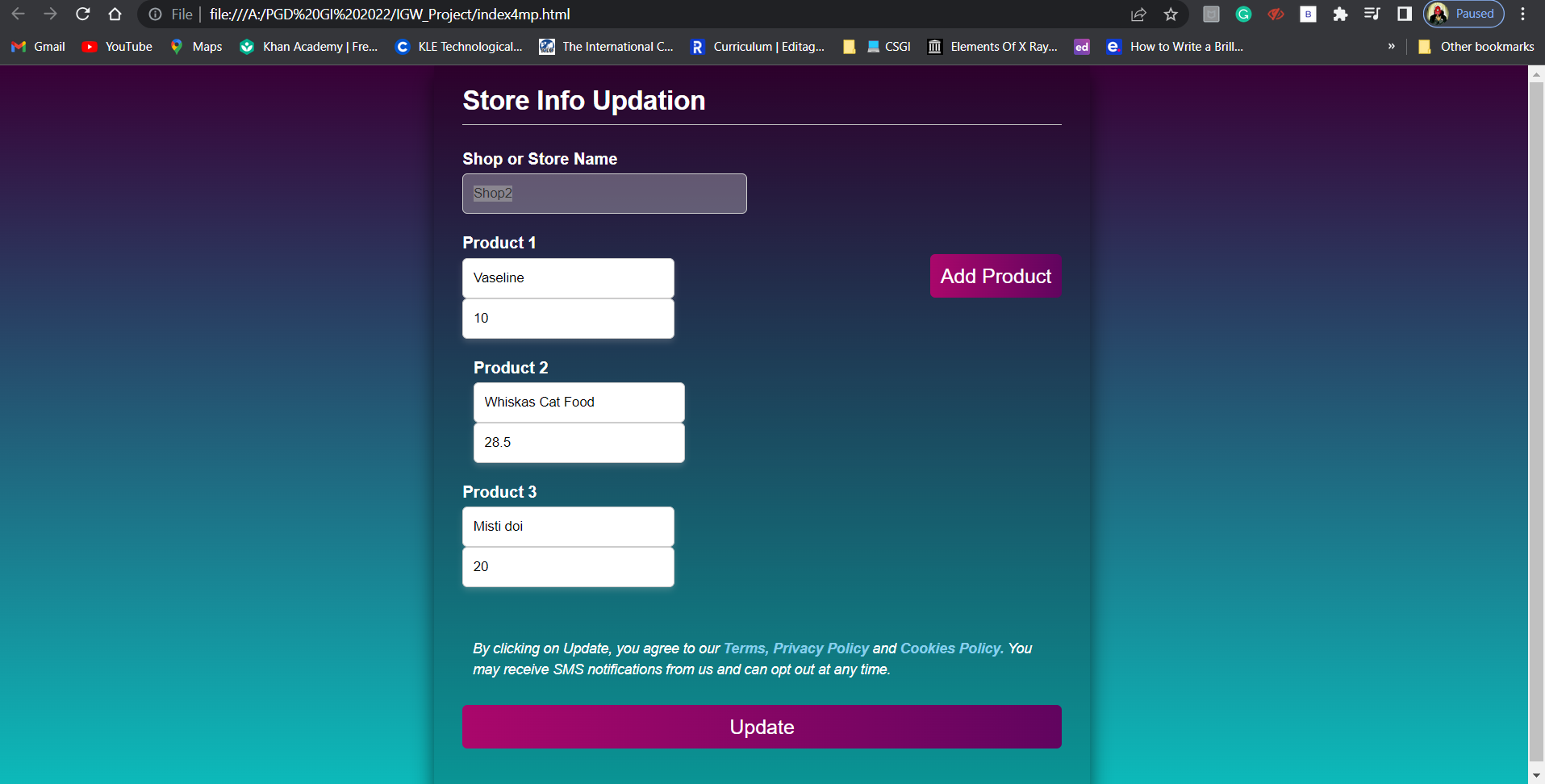
***Create Web Pages:*** The first step involves creating web pages using HTML and CSS to establish the user interface. This includes designing the landing page and incorporating features such as sign-in, sign-up, and registration forms.



*Fig.2: Screen shot of the landing page*

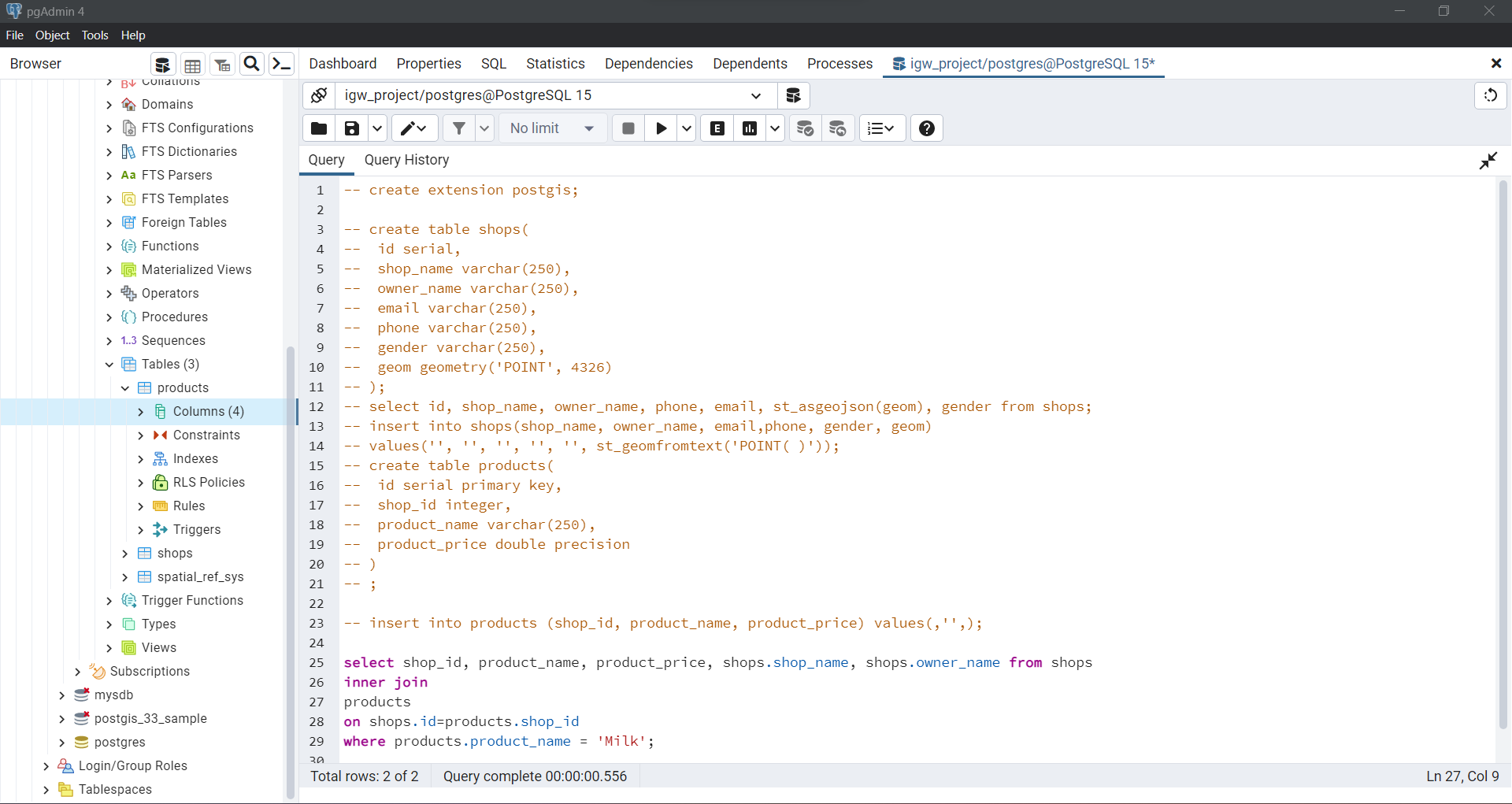


*Fig.3: Screen shot of the Store Owner’s Registration form*



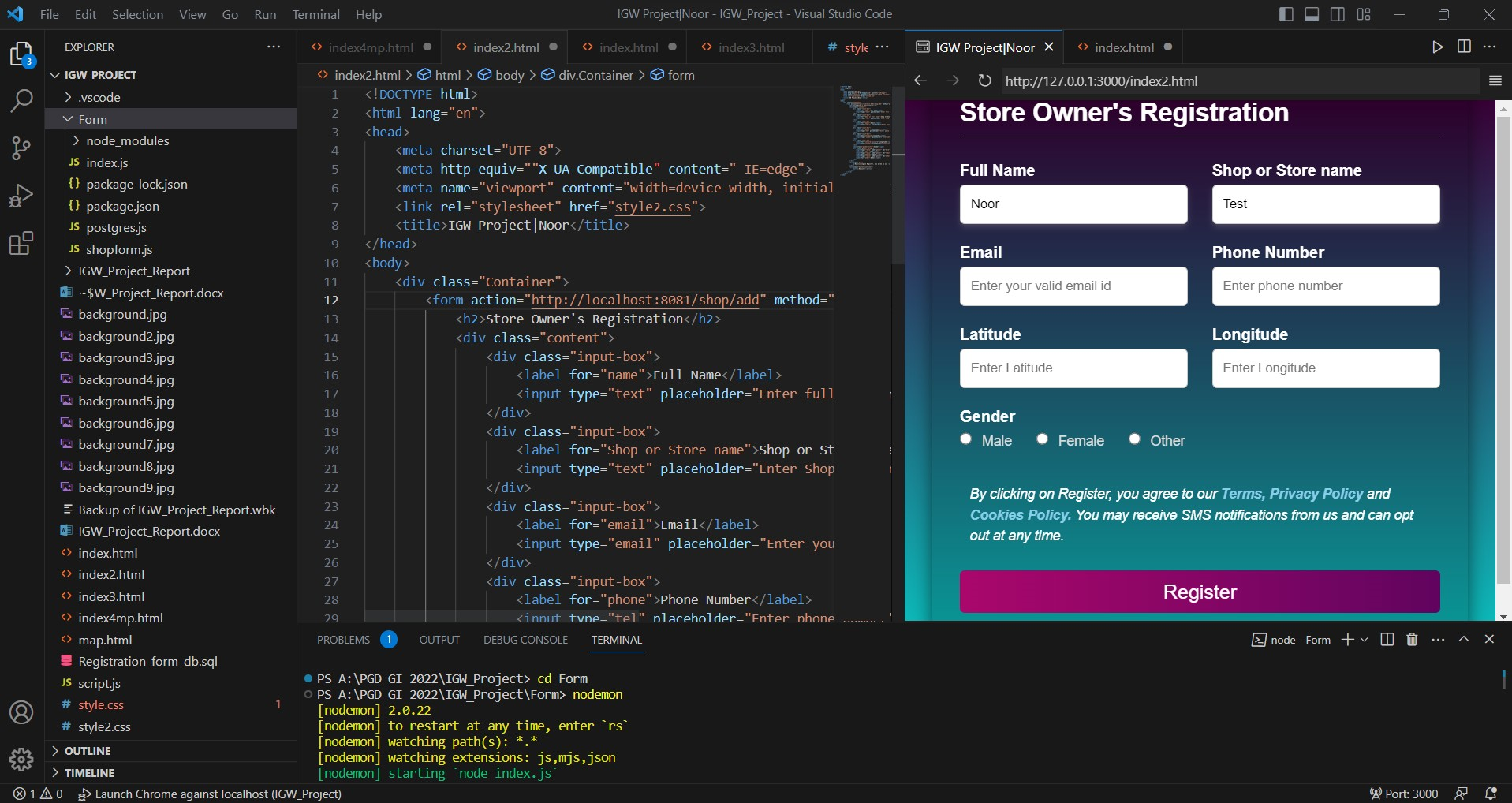
*Fig.4:* Screen shot of the page for Updating Store Info

***Create Tables in Database***: Next, database tables are created in PostgreSQL to store the data collected from the website. These tables are designed to efficiently store information related to products, stores, and user details.



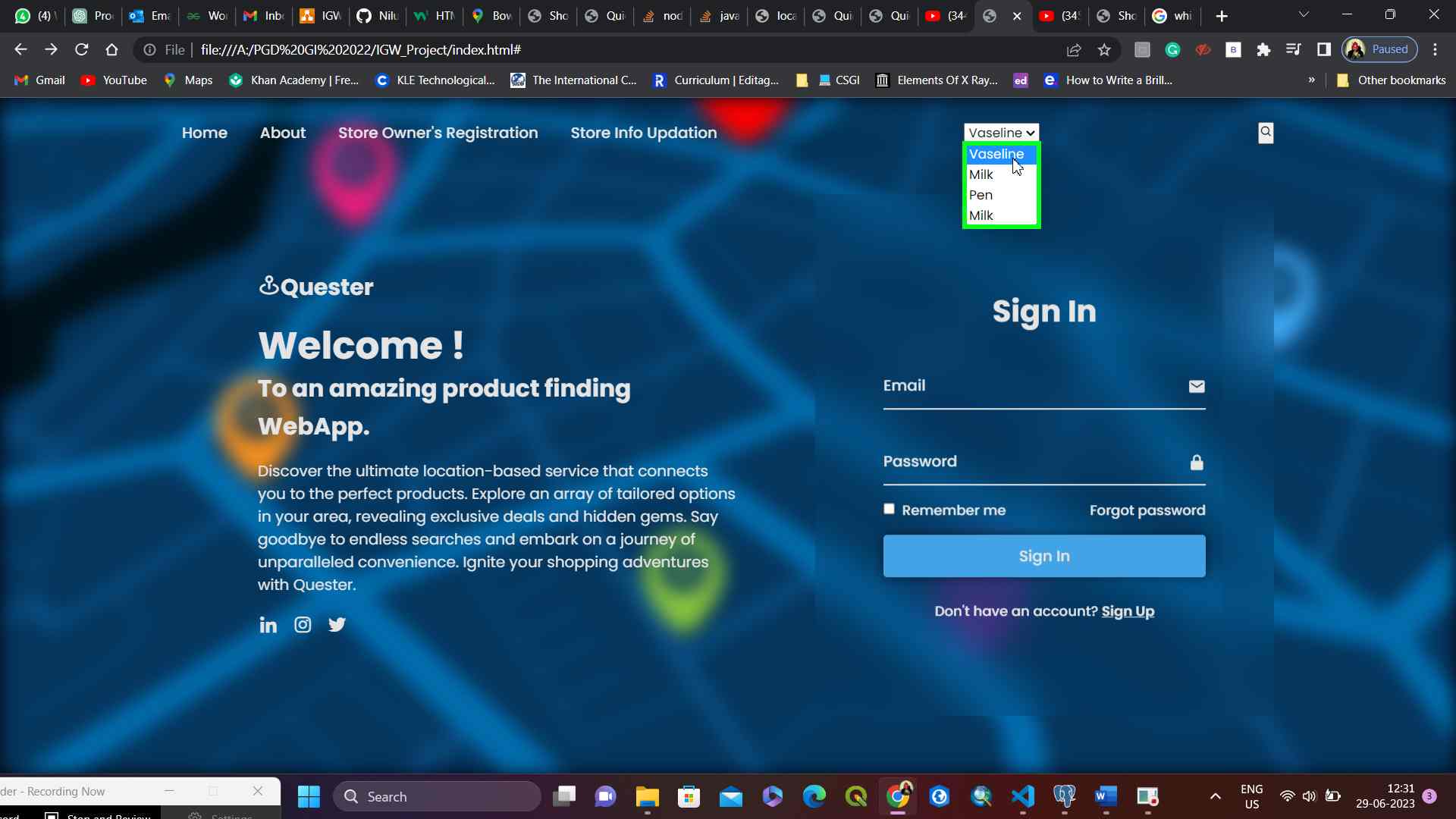
*Fig.5:* Screen shot of the code for creating tables in PostgreSQL

***Connect to PostgreSQL Database with Node.js:*** Node.js is utilized to establish a connection between the web application and the PostgreSQL database. This connection allows for seamless interaction between the web pages and the database, enabling data retrieval and manipulation.



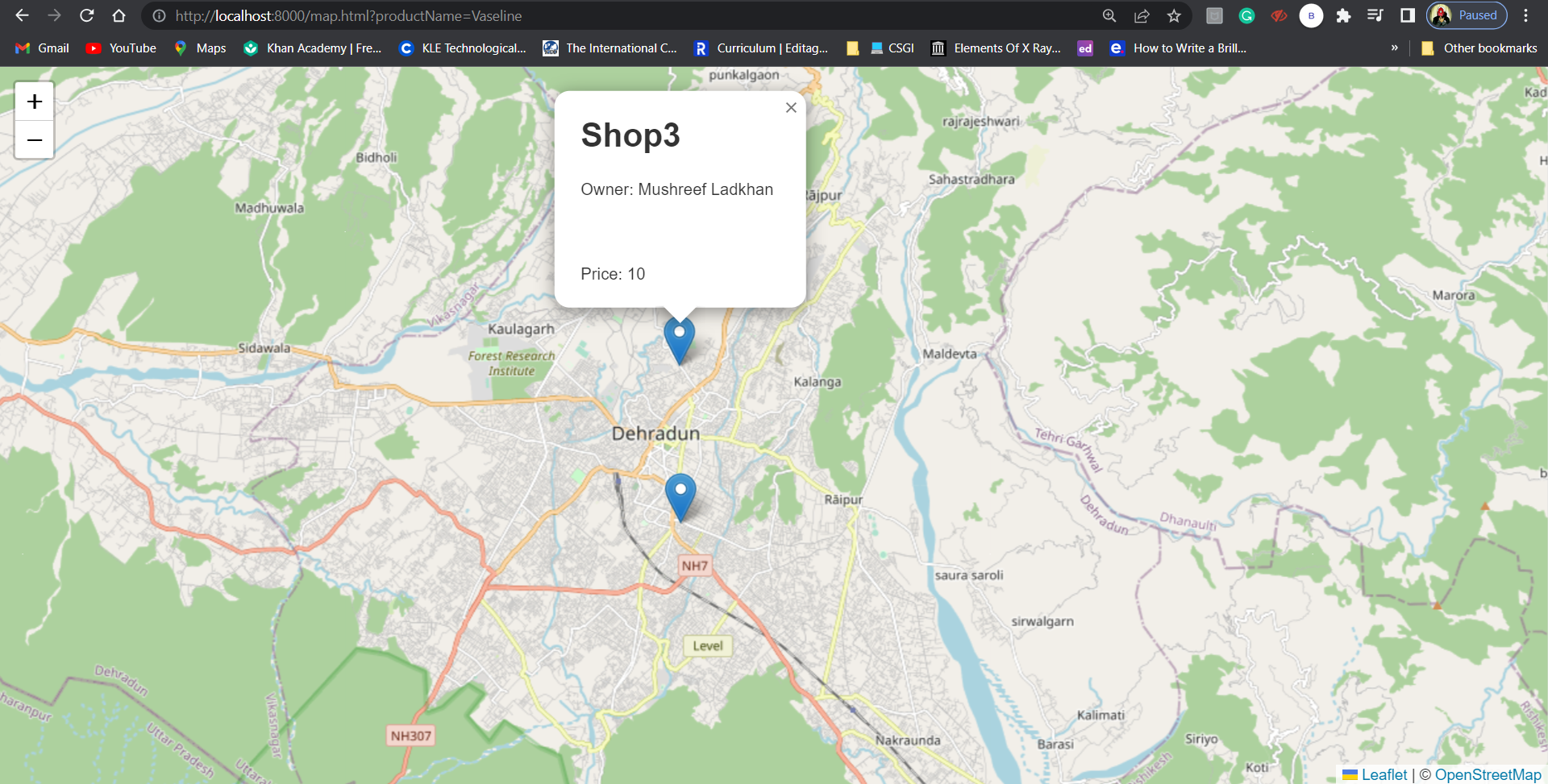
*Fig.6:* Screen shot of the code for connecting forms to database

***Implement Search Bar:*** A search bar is integrated into the web application, allowing users to query product information. The search functionality is implemented using appropriate programming techniques in conjunction with the database, ensuring accurate and efficient retrieval of relevant products.



*Fig.7:* Screen shot of page where User can search for the Product in the search bar

***Visualize Stores on Map with Leaflet:*** Leaflet.js libraries are utilized to visualize store locations on an interactive map. The store data retrieved from the database is mapped onto the Leaflet.js map, displaying markers or other visual elements to represent the stores. This visualization feature enhances the user experience by providing a clear and intuitive representation of store locations.

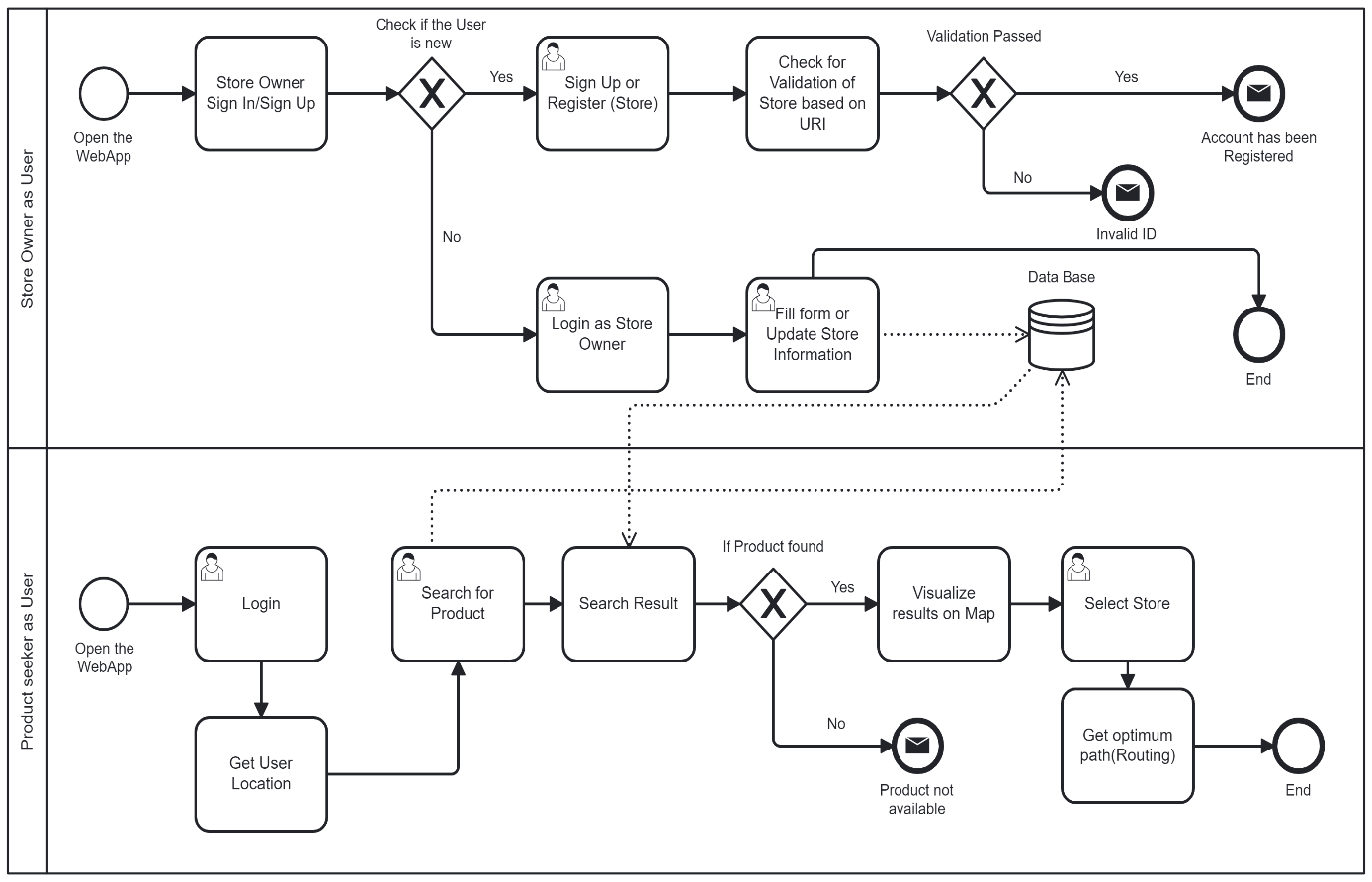


*Fig.8:* Screen shot of page where the Query result is visualized using Leaflet

A Repository is created in github which has the complete code for the application along with necessary documents in the github which can be accessed from the below link:

<https://github.com/Niluf11/IGW_Project>

## BPMN Diagram



The BPMN diagram consists of various elements such as start events, tasks, gateways, and end events, which define the flow and behaviour of the processes.

***Process: Store Owner as User***

* Start Event: Represents the start of the process when the WebApp is opened.
* Task: Store Owner Sign In/Sign Up: Represents the task of signing in or signing up for a store owner.
* Exclusive Gateway: Check if the User is new: Determines if the user is new or existing.
* User Task: Sign Up or Register (Store): Represents the task of signing up or registering as a store owner.
* Task: Check for Validation of Store based on URI: Validates the store based on a URI.
* Exclusive Gateway: Validation Passed: Determines if the store validation passed.
* User Task: Login as Store Owner: Represents the task of logging in as a store owner.
* User Task: Fill form or Update Store Information: Represents the task of filling a form or updating store information.
* Data Store Reference: Represents a reference to a data store (e.g., a database).
* End Event: Invalid ID: Represents the end event when the ID is invalid.
* End Event: Account has been Registered: Represents the end event when the account has been successfully registered.
* End Event: End: Represents the end event of the process.

***Process: Product seeker as User***

* Task: Search Result: Represents the task of searching for results.
* Exclusive Gateway: If Product found: Determines if the product was found.
* End Event: Product not available: Represents the end event when the product is not available.
* Task: Visualize results on Map: Represents the task of visualizing the results on a map.
* End Event: End: Represents the end event of the process.
* Task: Get optimum path(Routing): Represents the task of getting the optimum path or routing.
* User Task: Select Store: Represents the task of selecting a store.
* User Task: Search for Product: Represents the task of searching for a product.
* Task: Get User Location: Represents the task of getting the user's location.
* User Task: Login: Represents the task of logging in.
* Start Event: Open the WebApp: Represents the start of the process when the WebApp is opened.

# Softwares and tools used

## HTML and CSS

For creating and designing web pages, two key technologies are HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets). While CSS offers visual styling and layout flexibility, HTML provides the structural framework for structuring material on web pages, allowing developers to design aesthetically pleasing and user-friendly interfaces.

HTML and CSS were utilised to generate the landing page and design the user interface for the web application created for this project. The headings, paragraphs, forms, and other parts of the web pages' content were all defined using HTML. CSS was used to improve the web sites' aesthetic appeal, including their colours, fonts, layouts, and responsive design. Using HTML and CSS together, a user-friendly and visually beautiful website could be made.

## PostgreSQL for Database

PostgreSQL is an open-source object oriented relational database management system (ORDBMS) that provides a robust and feature-rich platform for storing and managing data. Its adherence to the SQL standard, scalability, and support for advanced data types make it a popular choice for various applications, including web development (https://www.postgresql.org/).

In the context of the web application developed in this project, PostgreSQL is employed as the database management system to store and manage data related to products, stores, and user information. It offers a reliable and efficient storage solution, ensuring data integrity, security, and seamless data retrieval. Using SQL queries and transactions allowed for flexible data manipulation and facilitated complex operations within the web application.

## Node.js

Node.js is a JavaScript runtime environment built on Chrome's V8 JavaScript engine. It allows developers to execute JavaScript code on the server-side, providing a powerful and efficient platform for server-side scripting and building scalable network applications. Its non-blocking I/O model and event-driven architecture make it well-suited for handling concurrent requests and building real-time applications. (<https://nodejs.org/>)

In the web application developed in this project, Node.js was utilized for server-side scripting and interaction with the PostgreSQL database. It facilitated the implementation of server-side logic, handling HTTP requests, and managing the flow of data between the web application and the database. The use of asynchronous programming techniques in Node.js ensured the responsiveness and scalability of the web application, allowing multiple users to interact with the system concurrently.

## Leaflet.js libraries for Map Visualization

Leaflet.js is a widely adopted JavaScript library for interactive map visualization. It provides a lightweight and versatile platform for creating dynamic and visually appealing maps on web pages. Leaflet.js offers comprehensive features, including the display of markers, overlays, and various map controls, allowing developers to create intuitive and interactive map interfaces.( <https://leafletjs.com/>)

In the web application developed in this project, Leaflet.js libraries are integrated to visualize the locations of stores on a map interface. The libraries facilitated the display of markers representing store locations, enabling users to identify and explore nearby stores easily. Additionally, the interactive capabilities of Leaflet.js allowed for seamless zooming, panning, and information display on the map, enhancing the user experience.

# Results and Discussion

The development of the web application resulted in the successful implementation of the most of the proposed functionalities. Through screenshots and descriptions, the results showcased the application in action, illustrating the seamless product search, store location visualization however routing capabilities couldn’t be completed due to time constraints. The development process encountered challenges, such as integrating external APIs and ensuring data security, which were to be addressed to overcome through rigorous testing and efficient solutions. The project provided valuable insights into the practical implementation of basic Web development in the context of location-based services for product search and store visualization.

# References

PostgreSQL: The World's Most Advanced Open Source Relational Database. (n.d.). Retrieved from <https://www.postgresql.org/>

Node.js. (n.d.). Retrieved from <https://nodejs.org/>

HTML: The Markup Language. (n.d.). Retrieved from <https://html.spec.whatwg.org/>

Leaflet.js. (n.d.). Retrieved from <https://leafletjs.com/>