# EXPENSE TRACKER

## A MICRO PROJECT REPORT

Submitted by

NAFILA NASRIN K

(Reg. No: 24MCR071)

RAJARAM S

(Reg. No: 24MCR080)

## SANTHOSH S

## (Reg. No: 24MCR092)

SOWMIYA SREE M

(Reg. No: 24MCR105)

THARUN PRAKASH S

(Reg. No: 24MCR116)

***in partial fulfilment of the requirements for the***

***award of the degree***

***of***

# MASTER OF COMPUTER APPLICATIONS

# DEPARTMENT OF COMPUTER APPLICATIONS

**KONGU ENGINEERING COLLEGE**

(Autonomous)

PERUNDURAI, ERODE – 638 060

MAY 2025

DEPARTMENT OF COMPUTER APPLICATIONS

KONGU ENGINEERING COLLEGE

**(Autonomous)**

**PERUNDURAI ERODE-638 060**

**May 2025**

BONAFIDE CERTIFICATE

Name : NAFILA NASRIN K (24MCR071)

RAJARAM S (24MCR080)

SOWMIYA SREE M (24MCR105)

SANTHOSH S(24MCR092)

THARUN PRAKASH S(24MCR116)

Course Code : **24MCT22**

Course Name : **CLOUD COMPUTING TECHNOLOGY**

Semester **II**

Certified that this is a bonafide record of work for application project done by the above students for **24MCT22- CLOUD COMPUTING TECHNOLOGY** during the academic year **2024-2025.**

Course Incharge HEAD OF THE DEPARTMENT (Signature with seal)

**Date:**

Submitted for the viva-voce examination held on

INTERNAL EXAMINER EXTERNAL EXAMINER

TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TITLE** | **PAGE NO.** |
| **1.** | INTRODUCTION | 1 |
|  | 1.1 OBJECTIVE | 2 |
|  | 1.2 SCOPE | 2 |
|  | 1.3 TECHNOLOGY USED | 3 |
| **2.** | SYSTEM REQUIREMENTS | 4 |
|  | 2.1 SOFTWARE REQUIREMENTS | 4 |
|  | 2.2 TOOLS AND PLATFORMS | 5 |
| **3.** | WEBPAGE DESIGN | 6 |
|  | 3.1 HTML STRUCTURE | 7 |
|  | 3.2 CSS STYLE | 8 |
| **4.** | DEPLOYMENT ON AWS S3 | 10 |
|  | 4.1 CREATING AN S3 BUCKET | 10 |
|  | 4.2 CONFIGURING BUCKET FOR STATIC  WEBSITE HOSTING | 12 |
|  | 4.3 UPLOADING FILES TO S3 | 13 |
|  | 4.4 TESTING THE DEPLOYMENT | 16 |
| **5.** | VERSION CONTROL WITH GIT | 17 |
|  | 5.1 INITIALIZING GIT REPOSITORY | 17 |
|  | 5.2 COMMITTING AND PUSHING CODE | 17 |
|  | 5.3 REPOSITORY STRUCTURE | 18 |
| **6.** | CONCLUSION | 19 |
| **7.** | REFERENCES | 20 |

# 1.INTRODUCTION

In today’s fast-paced world, financial management plays a crucial role in both personal and business domains. Tracking expenses efficiently ensures better budgeting, financial stability, and informed decision-making. Traditional expense tracking methods, such as manual record-keeping, spreadsheets, and paper-based receipts, often prove to be tedious, error-prone, and difficult to manage over time. The advent of cloud computing technology has revolutionized this process, offering automated, scalable, and secure solutions for expense management that are accessible from anywhere with an internet connection.

Cloud-based expense tracking systems utilize advanced computational resources and data storage facilities provided by cloud platforms such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud. These platforms enable users to record expenses in real time, categorize financial transactions, generate insightful reports, and integrate their financial data with other services such as banking systems and accounting tools. The ability to store data remotely ensures that users do not have to worry about losing physical records or experiencing device failures.

Furthermore, cloud computing enhances security measures by implementing encryption, multi-factor authentication, and secure access protocols to protect sensitive financial data from unauthorized access or cyber threats. These features make cloud-based solutions more reliable compared to traditional methods, where data can be misplaced or compromised due to insufficient security safeguards.

Cloud-based expense trackers are also highly adaptable, allowing users to customize features based on their specific financial goals. Businesses can leverage cloud solutions for expense monitoring across multiple departments, ensuring compliance with financial regulations while reducing administrative overhead. Individuals, on the other hand, can use these systems to track personal expenditures, set financial goals, and receive AI-driven insights on spending patterns.

This report delves into the implementation of cloud computing technology in expense tracking, highlighting its advantages, challenges, and future implications in the realm of financial management. By understanding the role of cloud computing, users can harness its full potential to achieve efficiency, accuracy, and security in expense tracking.

# 1.1 OBJECTIVE

The primary objective of this micro project is to design and develop a **static web page** that effectively presents comprehensive information about a specific country in a structured and visually engaging manner. Using **HTML and CSS**, the project aims to create an aesthetically appealing layout that showcases key details such as the country's name, national flag, capital city, population, official language(s), currency, geographic coordinates, and cultural highlights. Additionally, the webpage integrates an **embedded map** or location reference using services like Google Maps, enhancing the geographical context for users. A strong emphasis is placed on **responsive design**, ensuring that the webpage adapts seamlessly to various screen sizes, including desktops, tablets, and mobile devices, to provide an optimal user experience. To make the webpage publicly accessible, it is hosted using **Amazon Web Services (AWS) S3**, offering a scalable and reliable static site hosting solution. Furthermore, the project's **source code** is version-controlled using Git and uploaded to a **GitHub repository**, enabling open collaboration, efficient code management, and future scalability. By fulfilling these objectives, the project demonstrates the practical application of **web development and cloud hosting** in presenting structured and visually appealing geographic data while utilizing modern web technologies for enhanced accessibility and user engagement.

# SCOPE

The scope of project focuses on designing and developing a **static web page** that presents detailed information about a specific country in a structured and visually appealing format using **HTML and CSS**. The webpage includes essential details such as the country's name, national flag, capital city, population, official languages, currency, geographic coordinates, and cultural highlights, along with an **embedded map** for location reference using services like Google Maps. Additionally, the project’s **source code is version-controlled** using Git and uploaded to a GitHub repository, facilitating open collaboration, efficient code management, and future scalability. While the project does not involve complex functionalities like dynamic content updates or database integration, it establishes a strong foundation for future enhancements, including interactive elements and expanded data visualization, demonstrating the practical application of **web development and cloud hosting** in delivering structured and user-friendly geographic information.

# 1.3 TECHNOLOGY USED

## HTML (HyperText Markup Language)

* + Acts as the backbone of the web page.
  + Used to structure the content and elements of the country profile.
  + Includes semantic elements such as <header>, <section>, <article>, <nav>, and

<footer> for better accessibility and SEO.

## CSS (Cascading Style Sheets)

* + Provides styling to the web page.
  + Used for colors, fonts, layouts, spacing, alignment, and responsiveness.
  + Media queries ensure that the webpage works well across different screen sizes.
  + CSS Grid and Flexbox are used for dynamic layouts.

## AWS S3 (Simple Storage Service)

* + Acts as the cloud-based hosting platform.
  + Offers static website hosting capabilities with high availability and low cost.
  + Configuration includes:
    - Bucket creation
    - Static website hosting setup
    - Public access permission
    - Uploading HTML/CSS files
    - Configuring index document (e.g., index.html)
    - Generating a public URL for global access

## Git & GitHub

* + Git is used for version control and change tracking during development.
  + GitHub is used as a remote repository to:
    - Back up the code
    - Collaborate with others
    - Document the project
    - Share the code publicly

# SYSTEM REQUIREMENTS

The system requirements for the Country Profile Info Page are minimal and accessible to any user with a basic development environment. With tools like Visual Studio Code, Git, GitHub, and AWS S3, the project illustrates a complete workflow—from development and testing to deployment and public hosting—using widely adopted, beginner-friendly technologies and platforms.

# SOFTWARE REQUIREMENTS

## Text Editor / IDE:

* Required for: Writing and editing HTML and CSS code.

## Recommended Options:

[Visual Studio Code](https://code.visualstudio.com/)

Notepad++

## Web Browser

* Required for: Testing and previewing the web page locally and after deployment.

## Recommended Browsers:

Google Chrome (preferred for developer tools)

Microsoft Edge

## Version Control System

* Software: [Git](https://git-scm.com/)
* Required for: Managing versions of the project, tracking changes, and pushing code to GitHub.

## Static Website Files

* HTML5: For creating the structure and content of the webpage.
* CSS3: For styling and layout, including responsive design through media queries.

# 2.2 TOOLS AND PLATFORMS

## GitHub

* **Purpose**: Hosting the project repository, managing source code, enabling collaboration, and maintaining version history.
* **Benefit**: Allows for public access to the code and documentation, and acts as a backup.

## AWS S3 (Amazon Simple Storage Service)

* **Purpose**: Hosting the static webpage and making it publicly accessible via the internet.

## Setup Includes:

* Creating an S3 bucket
* Enabling static website hosting
* Uploading HTML and CSS files
* Configuring index document and permissions for public read access

## AWS Management Console

* **Purpose**: User interface for setting up and managing AWS S3 services, permissions, and configurations.
* **Alternative**: AWS CLI (Command Line Interface) can be used for automated deployment (optional for advanced users).

## Image Hosting (Optional)

* If external images like country flags or maps are used, they can either be hosted within the S3 bucket or loaded from trusted URLs.

# WEBPAGE DESIGN

For static web page, the design should be clean, structured, and visually engaging to effectively present country-specific information. Below are some design elements to consider:

Layout & Structure:

* Header: Displays the country's name and flag prominently.
* Main Content Sections: Includes information such as capital city, population, official language(s), currency, geographic coordinates, and cultural highlights.
* Map Integration: Embeds Google Maps for location reference.
* Responsive Design: Ensures adaptability across different screen sizes (desktop, tablet, and mobile).
* Footer: Contains credits, links, and a brief note about the data sources.

Color Scheme & Typography:

* Choose a color palette that aligns with the country's identity, such as national colors.
* Use clear, readable fonts (e.g., sans-serif for modern, clean styling).
* Maintain consistent spacing and contrast for readability.

Hosting & Performance Optimization:

* Host the webpage using AWS S3 for reliable access.
* Optimize images and code to ensure fast load times.
* Implement meta tags for SEO to make the page easily discoverable

# HTML STRUCTURE

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Country Information</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<header>

<h1 id="country-name">Country Name</h1>

<img src="flag.png" alt="Country Flag">

</header>

<main>

<section class="info">

<h2>Basic Information</h2>

<p><strong>Capital:</strong> <span id="capital">City Name</span></p>

<p><strong>Population:</strong> <span id="population">XX Million</span></p>

<p><strong>Official Language(s):</strong> <span id="language">Language Name(s)</span></p>

<p><strong>Currency:</strong> <span id="currency">Currency Name</span></p>

<p><strong>Geographic Coordinates:</strong> <span id="coordinates">Lat, Long</span></p>

</section>

<section class="map">

<h2>Location</h2>

<iframe

src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3173.593571908572!2d-122.08424958503842!3d37.42206537982048!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x808fbb5df222e399%3A0xb8d9bc4df0c34016!2sGoogleplex!5e0!3m2!1sen!2sus!4v1651234567890"

width="600" height="450" style="border:0;" allowfullscreen="" loading="lazy">

</iframe>

</section>

</main>

<footer>

<p>Data sourced from reliable references | Hosted on AWS S3</p>

</footer>

</body>

</html>

# CSS STYLE

body {

font-family: Arial, sans-serif;

text-align: center;

margin: 0;

padding: 0;

background-color: #f4f4f4;

}

header {

background-color: #003366;

color: white;

padding: 20px;

}

header img {

width: 150px;

}

.info, .map {

background-color: white;

padding: 20px;

margin: 20px auto;

width: 80%;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

footer {

background-color: #222;

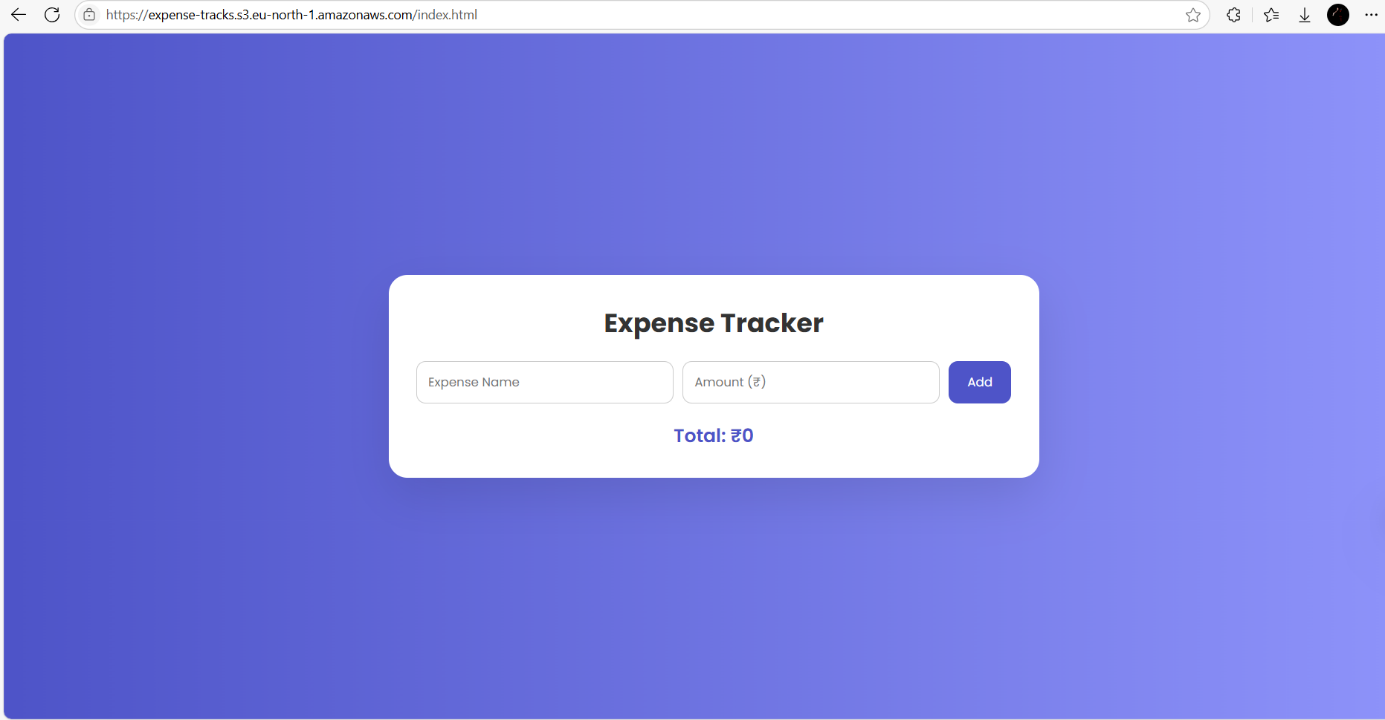
color: white;

padding: 10px;

position: relative;

bottom: 0;

width: 100%;

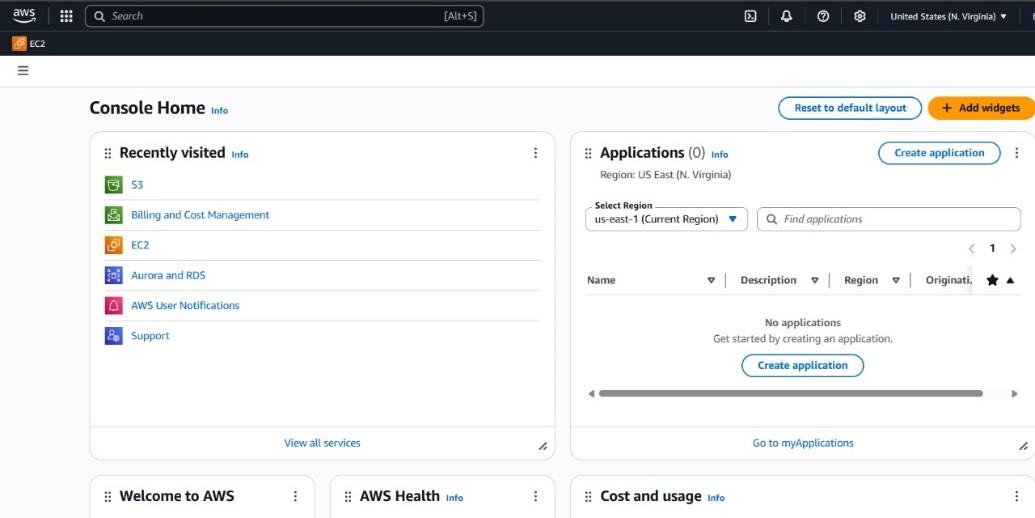


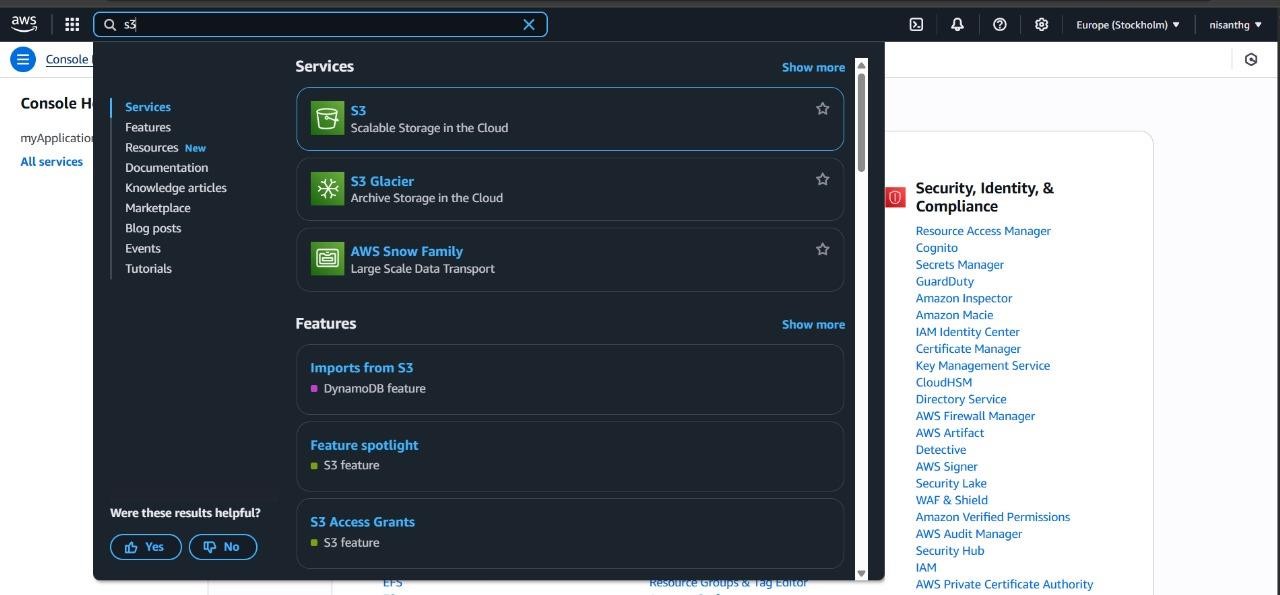
# DEPLOYMENT ON AWS S3

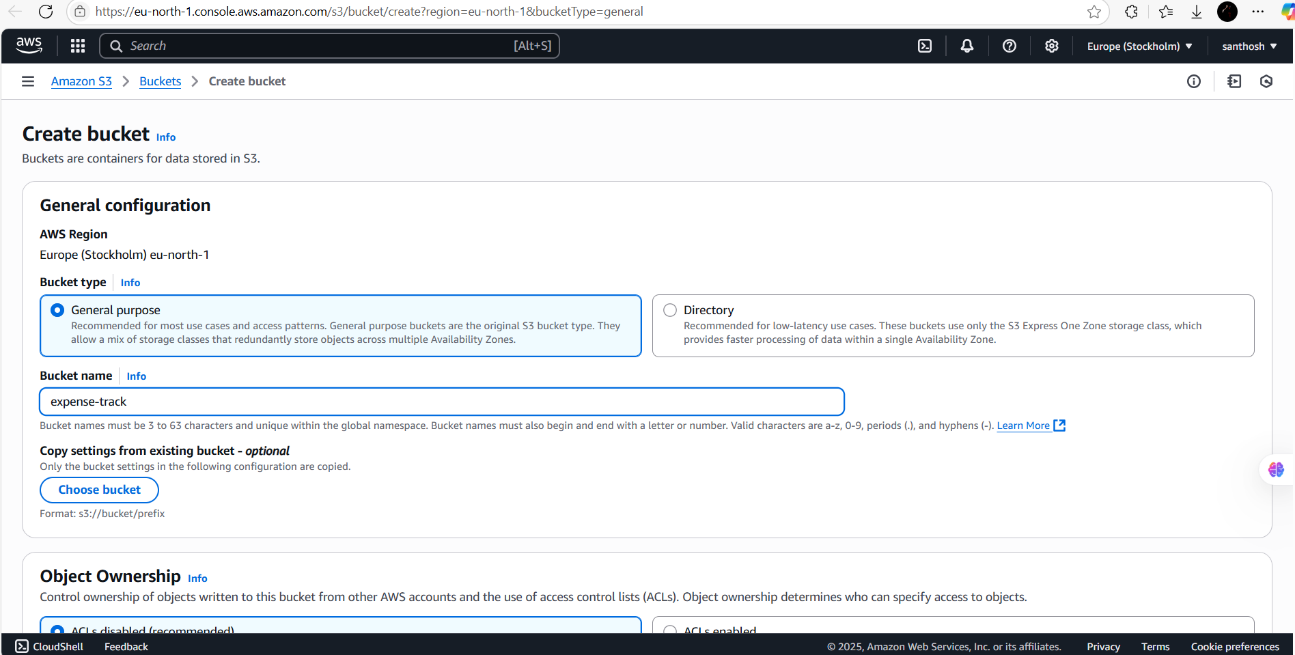
This section outlines the step-by-step process of deploying the Country Profile Info Page on Amazon Web Services (AWS) using S3 (Simple Storage Service), which is a cost- effective and reliable solution for hosting static websites.

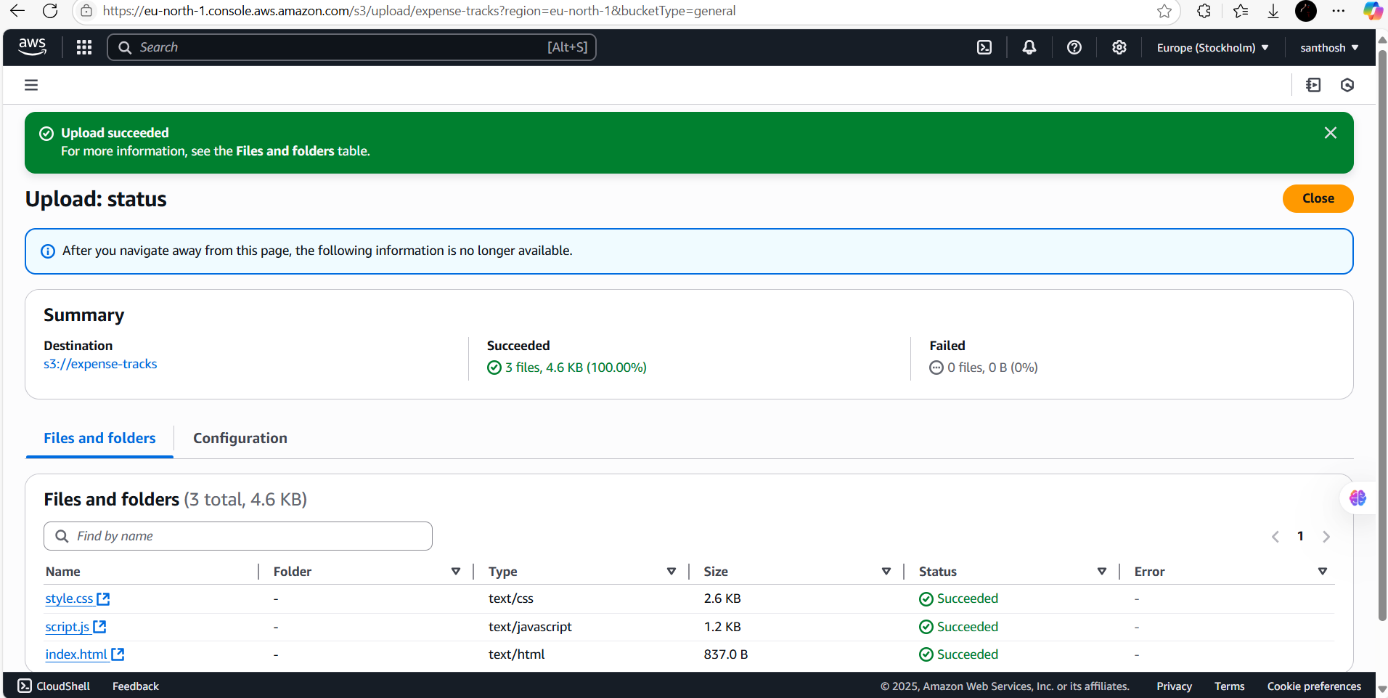
# CREATING AN S3 BUCKET

* Log in to your [AWS Management Console](https://aws.amazon.com/console/).
* Navigate to the S3 service from the Services menu.
* Click on the "Create bucket" button.
* Enter a unique bucket name and select your preferred AWS Region.
* Keep default settings for most options, but uncheck the box that blocks all public access (since static website hosting requires public access).
* Acknowledge the warning for public access and click "Create bucket".









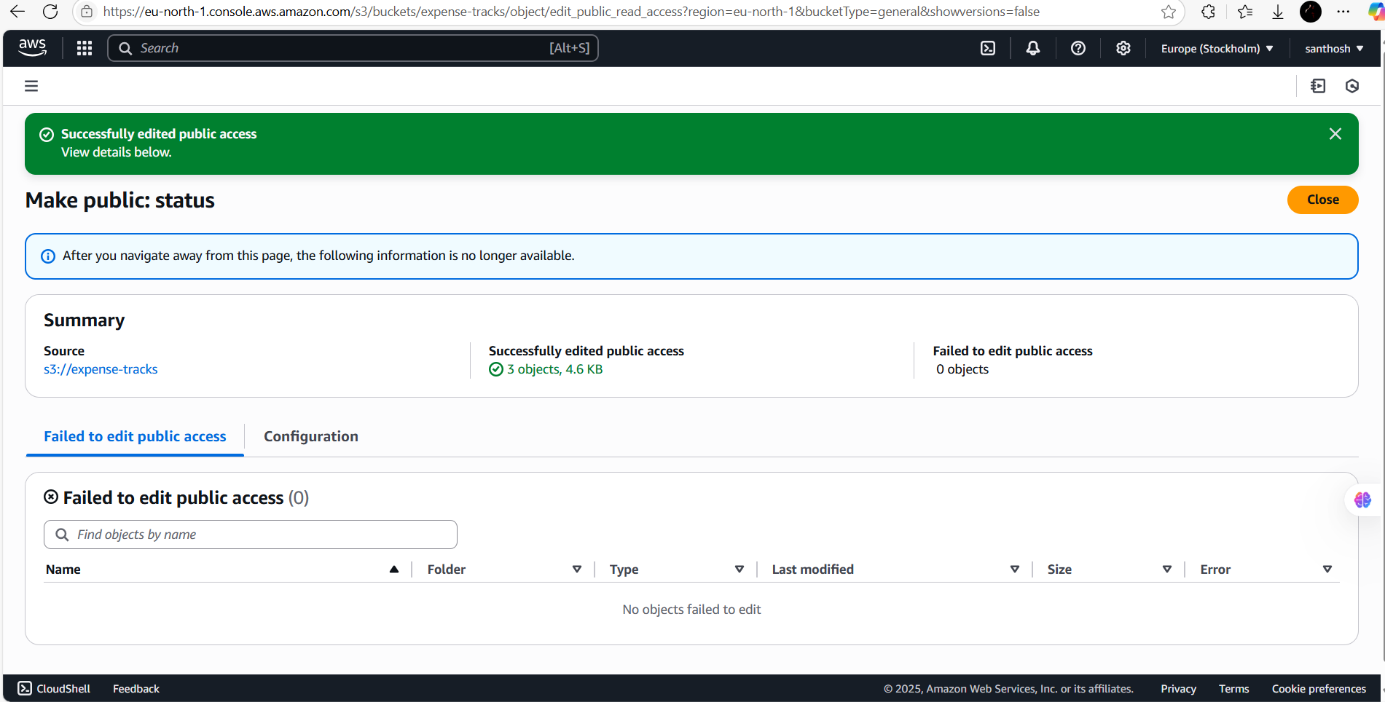
# 4.2 CONFIGURING BUCKET FOR STATIC WEBSITE HOSTING

* Once the bucket is created, click on the bucket name to open it.
* Go to the "Properties" tab.
* Scroll down to find the "Static website hosting" section.
* Click "Edit" and enable static website hosting.
* Choose “Host a static website”.
* In the fields provided:

Index document: index.html

Error document (optional): error.html

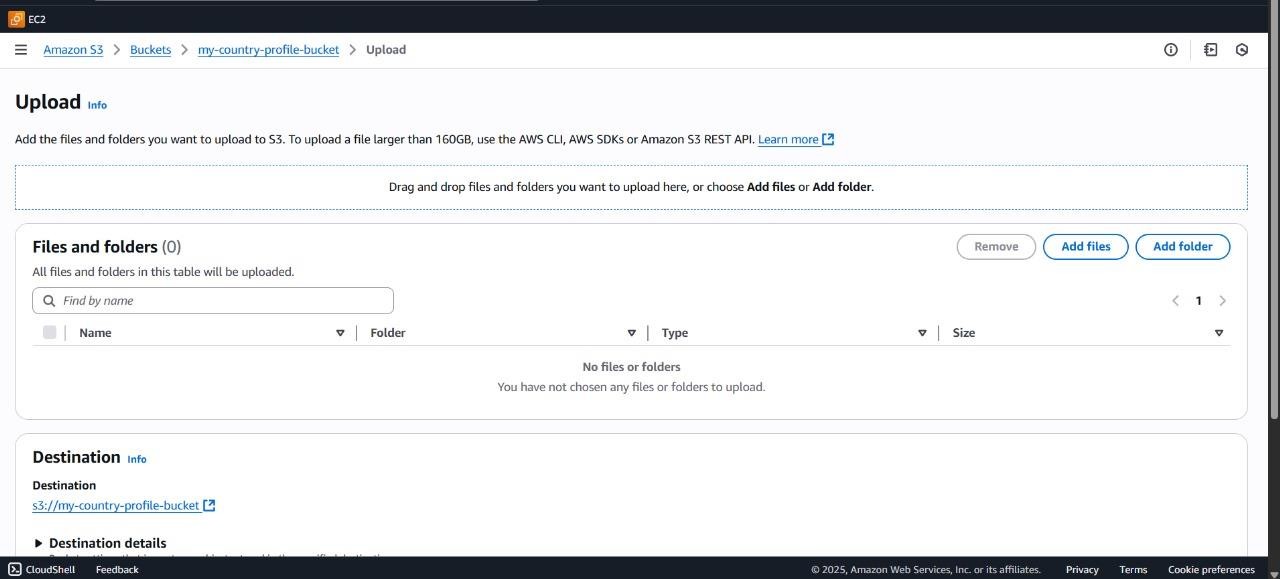
* Save the changes.

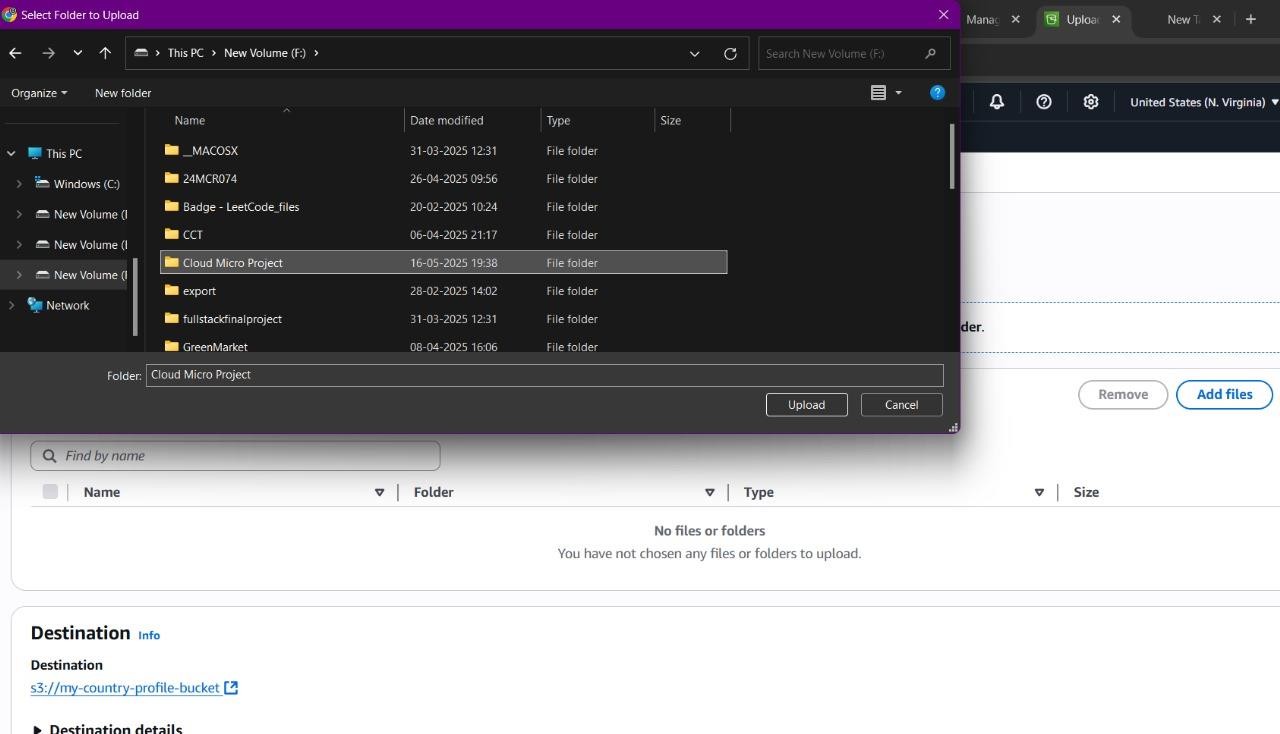


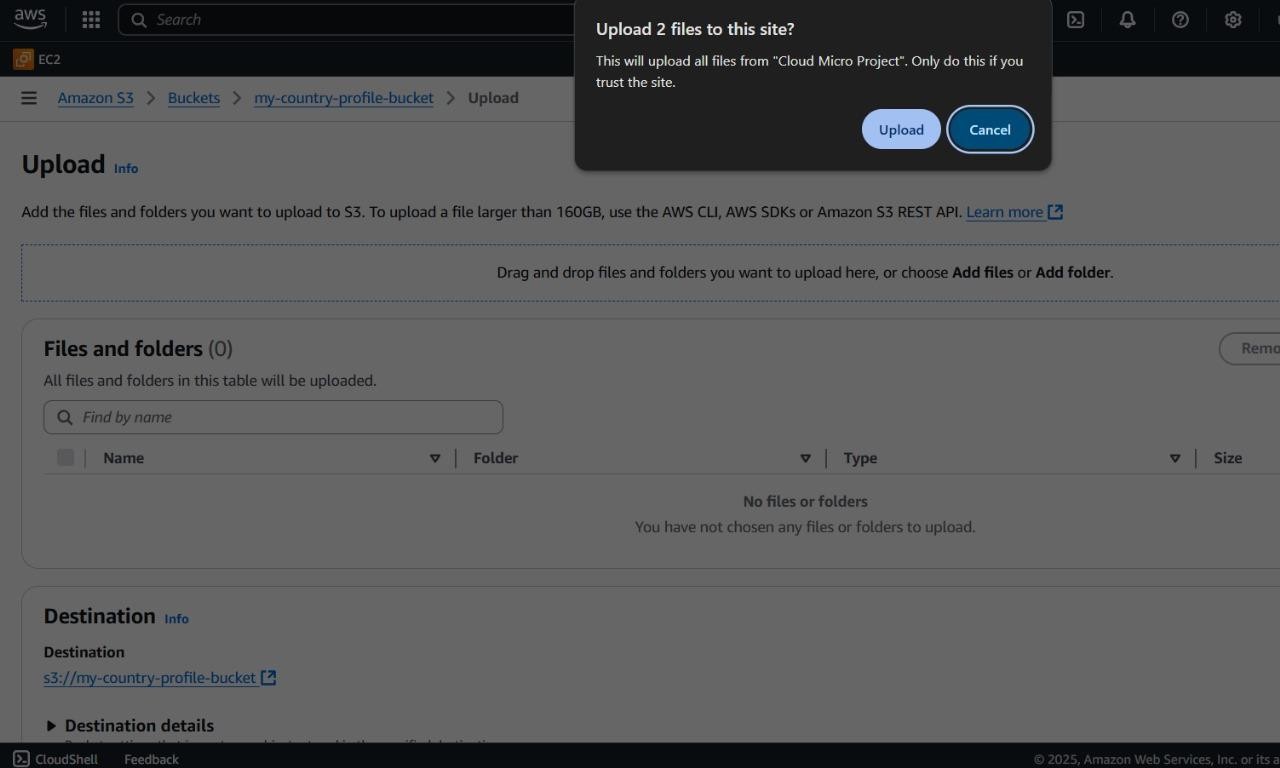
# UPLOADING FILES TO S3

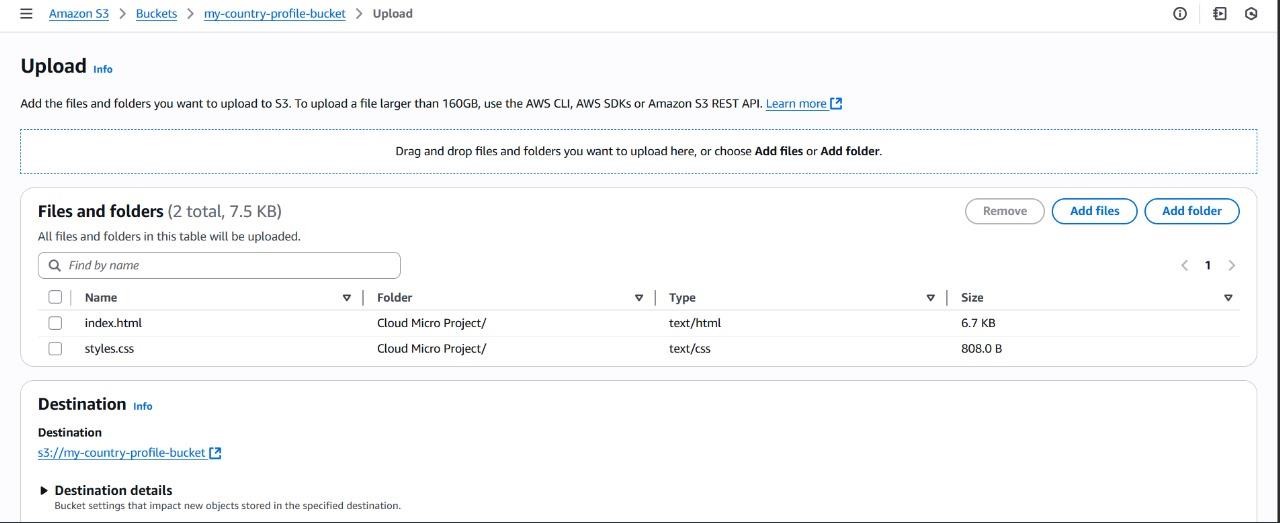
* Go to the "Objects" tab of your S3 bucket.
* Click "Upload" and then "Add files”.
* Select your project files:
* index.html
* style.css
* Any images or assets (e.g., flag, map image, etc.)
* Click "Upload" to finish the process.

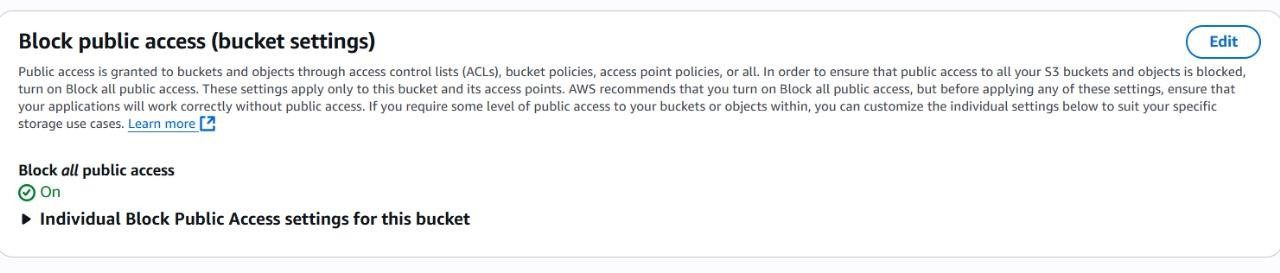
After upload, click on each file (especially index.html) and choose “Make public” (unless you’ve set bucket-level public access).

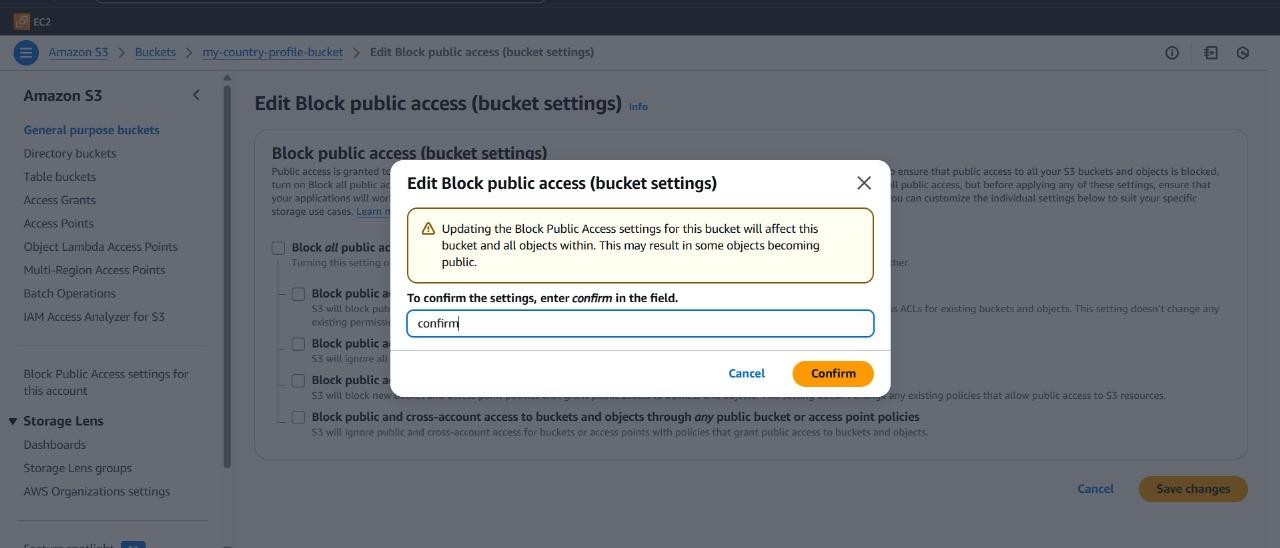


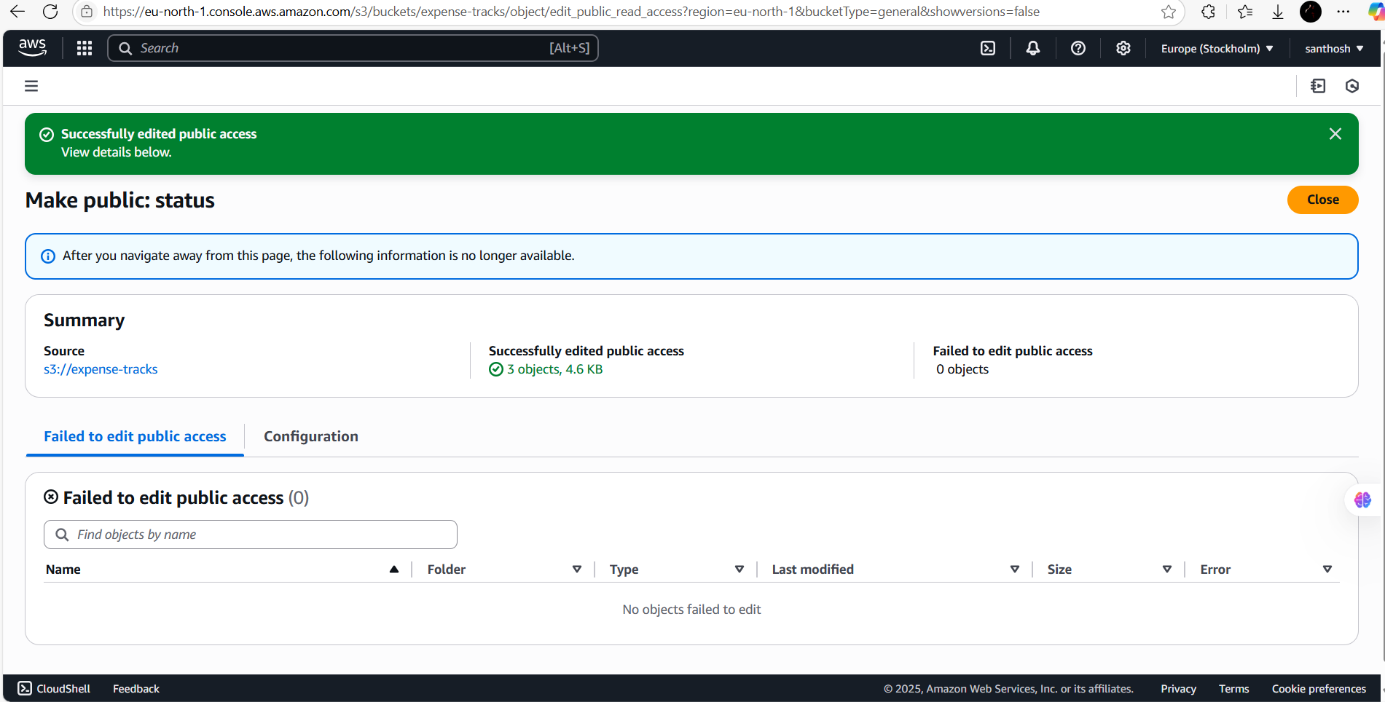






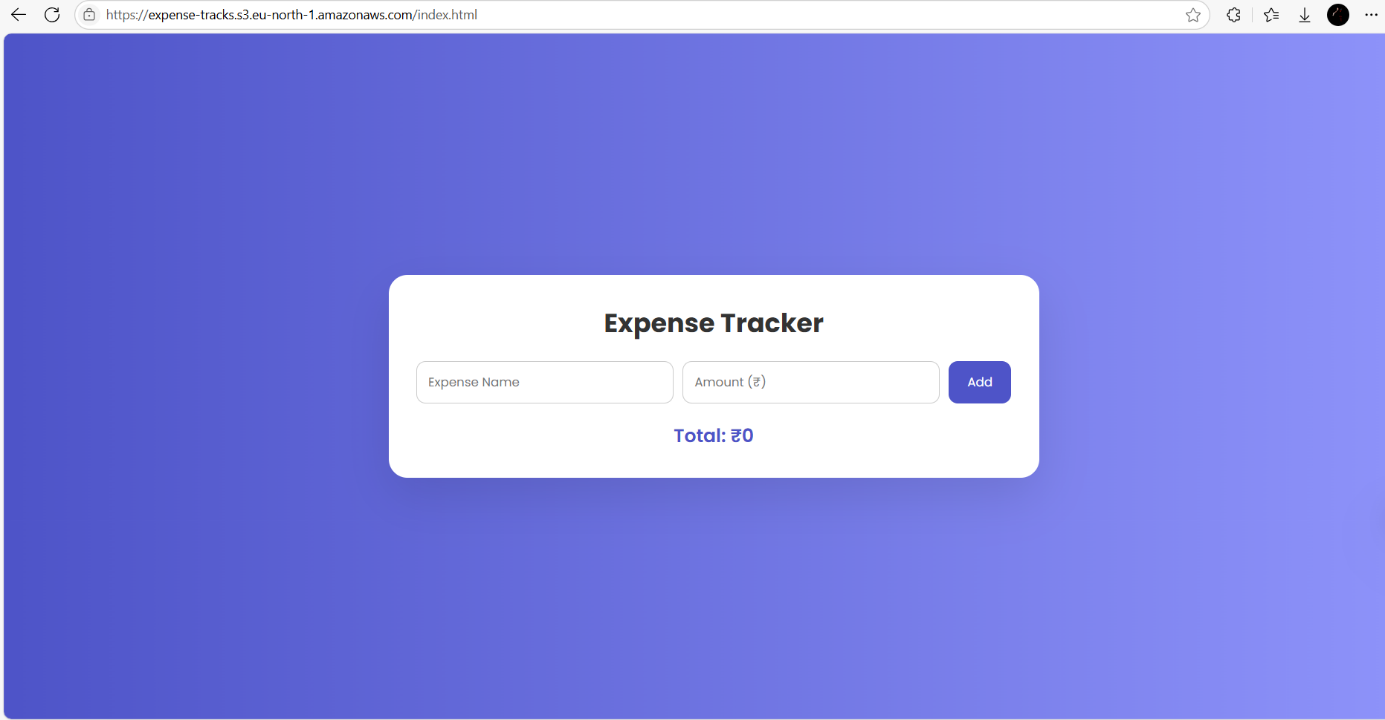






# TESTING THE DEPLOYMENT

* Return to the "Properties" tab of your bucket.
* In the Static website hosting section, find the “Bucket website endpoint” (e.g., [http://your-bucket-name.s3-website-region.amazonaws.com).](http://your-bucket-name.s3-website-region.amazonaws.com/)
* Click or copy this URL and open it in your browser.

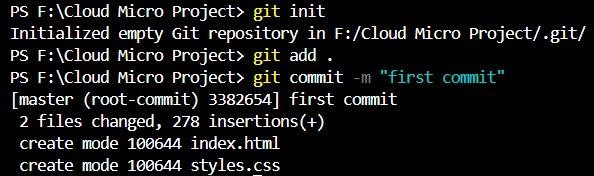


# VERSION CONTROL WITH GIT

Version control is a crucial part of modern software development, allowing developers to manage changes, track progress, and collaborate efficiently. In this project, Git is used as the version control system, and GitHub is the remote hosting platform. This section explains the process of initializing the Git repository, committing and pushing code, and the final structure of the repository.

# 5.1 INITIALIZING GIT REPOSITORY

The first step in implementing version control with Git is initializing a Git repository in the local project directory. This process involves creating a hidden directory (.git) within the project folder that will store metadata and object information needed to track changes to the files. This enables the local environment to recognize and monitor all modifications made to the project’s source code. Initialization ensures that the entire development process is documented and that changes can be reverted or reviewed at any stage of the project lifecycle.

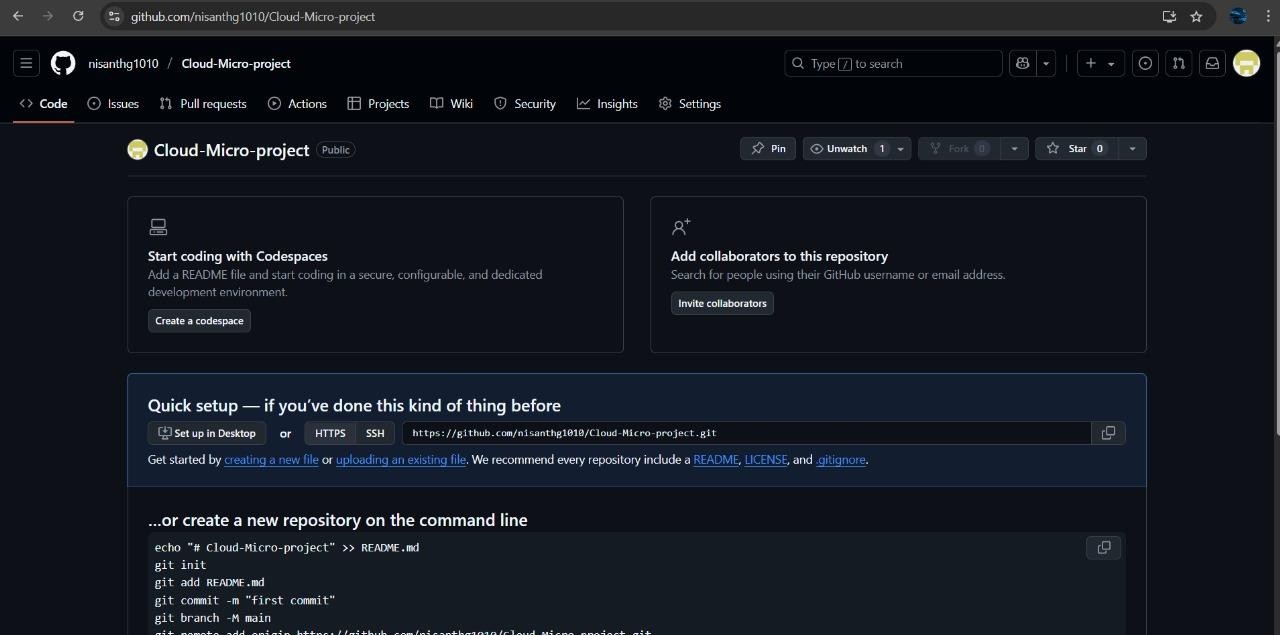


# COMMITTING AND PUSHING CODE

After initializing the repository, the next step is staging and committing code changes. Staging selects specific files to save, while committing records them with a message. Pushing to GitHub backs up the code remotely and supports collaboration.

# REPOSITORY STRUCTURE

Maintaining a clear and organized repository structure is vital for readability, maintainability, and scalability. In this project, the repository is structured to separate content (HTML), styling (CSS), and assets (such as images) into logical directories and files. A README.md file is included to offer documentation and project context, while a .gitignore file prevents unnecessary or sensitive files from being tracked. This structured approach allows developers and collaborators to easily navigate the project, understand its components, and contribute effectively. The hosted repository on GitHub further enables version history access, collaborative coding, and transparent project sharing.



# CONCLUSION

The expense tracker successfully demonstrates the implementation of semantic HTML structure, responsive web design, and cloud hosting to create a static webpage that effectively presents country-specific information in a well-organized and visually appealing format. By leveraging HTML and CSS, the webpage provides a clean and structured layout, incorporating anchor-based navigation for smooth scrolling, image embedding for national flags, and Google Maps integration via for accurate location visualization. The emphasis on responsive design ensures an optimal user experience across various screen sizes, including desktops, tablets, and mobile devices.

To enhance accessibility and reliability, the webpage is hosted using Amazon Web Services (AWS) S3, offering scalable static site hosting. Additionally, the project follows best practices in version control, utilizing Git and GitHub for efficient code management, open collaboration, and future scalability. While the current implementation is static, it establishes a strong foundation for potential future enhancements, such as dynamic content updates using APIs, interactive elements, and real-time data visualization.

Overall, the importance of modern web development technologies in creating structured, accessible, and interactive digital experiences. By integrating essential features, maintaining a user-friendly design, and leveraging cloud-based hosting, the project serves as a practical demonstration of how web technology and cloud computing can be utilized to present country-specific information effectively.

# REFERENCES

1. W3C, "HTML5 Specification," World Wide Web Consortium (W3C), 2014. [Online].

Available: <https://www.w3.org/TR/html5/>

1. W3C, "Cascading Style Sheets (CSS) - The Official Definition," World Wide Web Consortium (W3C), 2023. [Online]. Available: <https://www.w3.org/Style/CSS/>
2. Git SCM, "Git - Distributed Version Control System," Git, 2024. [Online]. Available: <https://git-scm.com/>
3. GitHub Inc., "GitHub: Where the world builds software," GitHub, 2024. [Online].

Available: <https://github.com/>

1. E. Freeman and E. Robson, Head First HTML and CSS, 2nd ed., O'Reilly Media, 2012.
2. B. Beighley, Responsive Web Design with HTML5 and CSS, 2nd ed., Peachpit Press, 2015.
3. Mozilla Developer Network (MDN), "HTML: HyperText Markup Language," MDN Web Docs, 2024. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTML>
4. M. Fowler, "Continuous Integration," martinfowler.com, 2006. [Online]. Available: https://martinfowler.com/articles/continuousIntegration.html
5. Amazon Web Services, "Getting Started with AWS – Hosting a Static Website," AWS Getting Started Resource Center, 2023. [Online]. Available: [https://aws.amazon.com/getting-](https://aws.amazon.com/getting-started/hands-on/host-static-website/) [started/hands-on/host-static-website/](https://aws.amazon.com/getting-started/hands-on/host-static-website/)
6. J. Duckett, HTML and CSS: Design and Build Websites, 1st ed., Indianapolis, IN, USA: Wiley, 2011.