

*End Term Project Report*

*On*

*CodeBuddy*

*Submitted to*

***University of Petroleum and Energy Studies***

*In Partial Fulfilment for the award of the degree of*

Bachelors in Technology

In

COMPUTER SCIENCE AND ENGINEERING (with specialisation in Cloud Computing and Virtualization Technology)

Submitted By

|  |  |  |
| --- | --- | --- |
| **Specialization** | **SAP ID** | **Name** |
| CCVT | 500082997 | Aastha |
| CCVT | 500084925 | Akshat Dhaka |

*Under the guidance of*

Dr. Alok Aggarwal

Designation: Professor

**University of Petroleum and Energy Studies**

**Dehradun-India**

September 2023

Table of Contents

|  |  |  |
| --- | --- | --- |
| **Topic** | | **Page No** |
| Table of Content | |  |
| 1 | Project Title | 3 |
| 2 | Introduction |  |
|  | 2.1 Project Scope | 3-4 |
|  | 2.2 Problem Statement | 4 |
|  | 2.3 Target Beneficiary | 4-5 |
|  | 2.4 Purpose of the Project | 5 |
| 3 | Project Description |  |
|  | 3.1 Reference Algorithm | 5-6 |
|  | 3.2 SWOT Analysis | 6-7 |
|  | 3.3 Project Features | 7-8 |
|  | 3.4 Design diagrams | 8 |
| 4 | System Requirements |  |
|  | 4.1 User Interface | 8 |
|  | 4.2 Software Interface | 8-9 |
| 5 | Non-functional Requirements |  |
|  | 5.1 Performance requirements | 9 |
|  | 5.2 Security requirements | 9 |
|  | 5.3 Software Quality Attributes | 9-10 |
| 6 | Methodology | 10-11 |
| 7 | Results | 11 |
| 8 | References | 11 |

**1. Project Title**

CodeBuddy: A platform for elevated coding efficiency through seamless GitHub Actions

**2. Introduction**

In the world of software development, collaboration has become increasingly important. Gone are the days when developers worked in isolation; today, they thrive in a more connected and cooperative environment. In this era of rapid technological advancement, where new programming languages and frameworks emerge frequently, developers need a supportive platform that not only nurtures their creativity but also upholds the highest standards of code quality. This is where CodeBuddy comes into play, serving as the ultimate ally for developers looking to refine their coding skills, explore new languages, and collaborate seamlessly. CodeBuddy is designed to be a digital haven for developers of all levels, from newcomers taking their initial steps into the coding world to seasoned professionals seeking to streamline their workflow. Its primary goal is to provide a versatile and user-friendly platform where developers can harness the power of collaboration, code improvement, language versatility, and code quality enhancement. Through its intuitive interface and a rich set of features, CodeBuddy aspires to become the go-to destination for all your coding needs.

**2.1. Project Scope**

* The Developer's Dilemma:

Imagine this scenario: You've just written a substantial piece of code for your project, and it's functioning as intended. However, upon reviewing your work, you wonder if there's a more elegant way to express your logic. Maybe you're contemplating rewriting it in a different programming language to explore its potential. Or perhaps you want to invite a fellow developer to review your code and provide valuable feedback. Historically, these tasks have been time-consuming and complex, involving multiple tools and platforms. Recognizing this developer's dilemma, CodeBuddy offers an integrated solution. With CodeBuddy, you can easily refactor your code, switch between programming languages all within a single, unified environment. It simplifies the development process, allowing developers to focus on their core strength: creating exceptional software.

* Effortless Code Refactoring:

Code quality is of utmost importance. Well-structured, readable code not only reduces maintenance efforts but also enhances overall application performance. CodeBuddy simplifies code refactoring, helping you identify and eliminate code issues, redundancy, and inefficiencies. With just a few clicks, you can apply best practices and ensure that your codebase is both elegant and easy to maintain.

* Embrace Multilingual Coding:

The programming world is vast and diverse, with each language offering unique strengths and weaknesses. CodeBuddy empowers you to explore this diversity by

seamlessly switching between programming languages. Whether you want to experiment with a new language, port an existing project, or simply improve your skills, CodeBuddy has you covered. Say goodbye to the limitations of a single language and embrace the freedom to code in your language of choice.

* Enhancing Code Quality:

Code quality is not just about appearance; it's about reliability and maintainability. CodeBuddy equips you with powerful tools to assess and enhance your code's quality. Through automated code analysis, you can identify potential issues, security vulnerabilities, and performance bottlenecks. CodeBuddy offers suggestions and insights to help you write code that meets the highest industry standards.

* The CodeBuddy Experience:

CodeBuddy is designed with simplicity and usability in mind. The platform offers an intuitive interface that caters to developers of all backgrounds. Whether you're a beginner looking to learn, an intermediate developer seeking collaboration, or an expert aiming to streamline your workflow, CodeBuddy adapts to your needs.

* Implementing GitHub Actions:

Through the project the objective is to learn and implement the new and trending technology of DevOps. Through this it will automatically deploy the website to AWS.

**2.2. Problem Statement**

In the dynamic realm of software development, developers encounter challenges in refining coding skills, exploring new languages, and collaborating seamlessly. With the continual emergence of programming languages and frameworks, there is a need for a comprehensive platform—CodeBuddy. This platform aims to be a versatile and user-friendly haven, supporting diverse languages, fostering collaboration, enhancing code quality, and providing a streamlined workflow. CodeBuddy seeks to empower developers, from novices taking initial coding steps to seasoned professionals, by offering a unified solution for their coding needs in an ever-evolving technological.

**2.3. Target Beneficiaries**

The target beneficiaries also known as target group or project beneficiaries are those who will benefit from this project. They can be directly or indirectly affected by the project.

CodeBuddy is meant to help a wide range of people who are interested in coding and programming. This includes beginners who are just starting to learn, students studying computer science, people looking to become professional developers, experienced developers, teachers who want to teach coding and small groups or teams working on projects. It is also helpful for coding communities, open-source contributors, freelancers, mentors, recruiters and anyone looking to improve their coding skills or find coding opportunities. It is like a friendly place where all these people can learn and make their process easy and simple.

**2.4. Purpose of the project**

The purpose of the project is to streamline the process of software development and deploy it automatically. Below are some points for the same:

* Enhance Code Quality: CodeBuddy aims to help developers write clean, well-structured, and maintainable code. It provides tools and suggestions to identify and eliminate code issues, ensuring that software projects are of high quality.
* Support Code Refactoring: One of the key objectives is to simplify the code refactoring process. CodeBuddy offers features and automated assistance to refactor code efficiently, reducing technical debt and improving code readability.
* Enable Language Flexibility: CodeBuddy seeks to empower developers by allowing them to experiment with and switch between different programming languages seamlessly. This objective encourages exploration and adaptation to the most suitable language for specific tasks.
* Simplify Development Workflow: The platform aims to streamline the development workflow by providing an integrated environment where developers can perform various tasks without the need for multiple tools or platforms.
* Seamless Deployment Procedure: It will deploy the website using GitHub Actions which is trending nowadays and makes deployment simple and hassle free.

**3. Project Description**

In the world of software development, developers often run into problems that slow down their work and make it hard to write high-quality code. These problems include the need to make code better, the wish to try different programming languages, and the importance of keeping code in good shape. It's tough for developers to find a solution that combines all these things into one easy-to-use platform. CodeBuddy steps in to solve these problems by giving developers a single place to improve their code, experiment with different languages, and make their code better.

**How does GPT API works internally?**

Generative Pre-trained Transformers, commonly known as GPT, are a family of neural network models that uses the transformer architecture and is a key advancement in artificial intelligence (AI) powering generative AI applications such as ChatGPT. GPT models give applications the ability to create human-like text and content (images, music, and more), and answer questions in a conversational manner. Organizations across industries are using GPT models and generative AI for Q&A bots, text summarization, content generation, and search.

The GPT models are neural network-based language prediction models built on the Transformer architecture. They analyse natural language queries, known as prompts, and predict the best possible response based on their understanding of language. To do that, the GPT models rely on the knowledge they gain after they’re trained with hundreds of billions of parameters on massive language datasets. They can take input context into account and dynamically attend to different parts of the input, making them capable of generating long responses, not just the next word in a sequence.

GPT-3 was trained with over 175 billion parameters or weights. Engineers trained it on over 45 terabytes of data from sources like web texts, Common Crawl, books, and Wikipedia. Prior to training, the average quality of the datasets was improved as the model matured from version 1 to version 3.

GPT-3 trained in a semi-supervised mode. First, machine learning engineers fed the deep learning model with the unlabelled training data. GPT-3 would understand the sentences, break them down, and reconstruct them into new sentences. In unsupervised training, GPT-3 attempted to produce accurate and realistic results by itself. Then, machine learning engineers would fine-tune the results in supervised training, a process known as reinforcement learning with human feedback (RLHF).

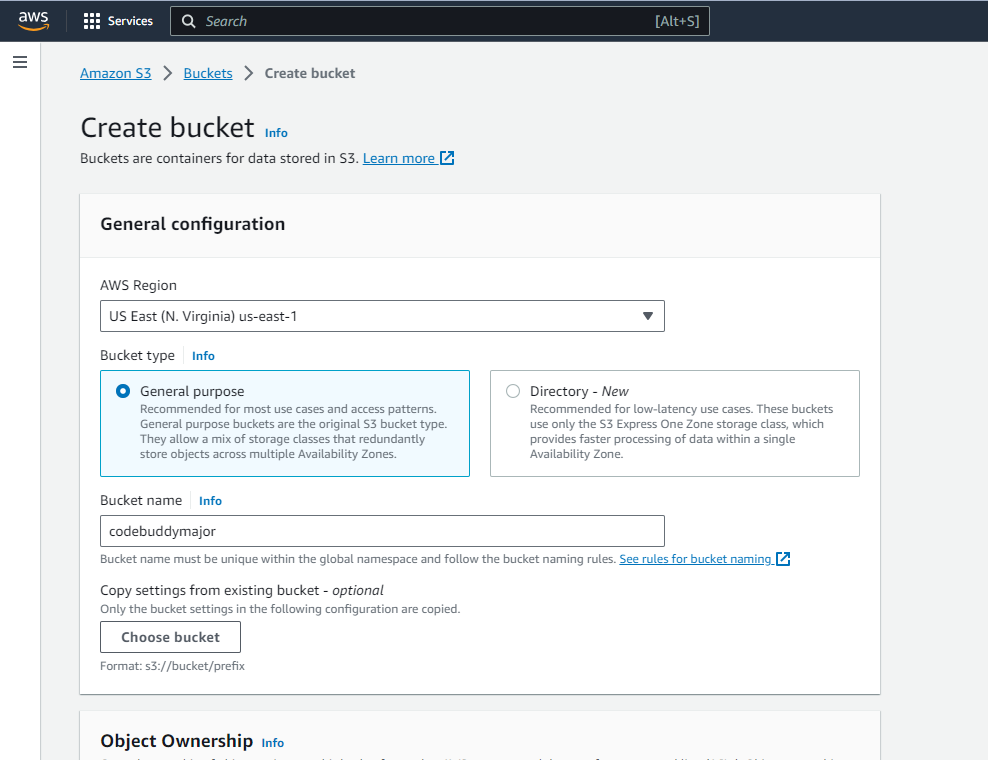
**How does CodeBuddy works?**

Below are the steps for implementing CodeBuddy :

* Initialization and Access: We accessed the GPT API by obtaining appropriate API keys or credentials. This allows to make requests to the GPT model hosted on a server.
* Input Prompt: Then submit a natural language prompt or query to the GPT API. This input can be a question, a request for content generation, or any other text-based task.
* Communication with GPT Model: The API sends the input prompt to the underlying GPT model, which could be GPT-3 or a later version like GPT-4 [in our case its GPT-3]. This model is pre-trained with massive amounts of data and is capable of understanding and generating human-like text.
* Tokenization and Preprocessing: The input prompt undergoes tokenization, breaking it down into smaller units or tokens. Tokens can be words, sub-words, or characters, facilitating the model's understanding of the input.
* Contextual Understanding: Leveraging the transformer architecture which is the base of GPT API, particularly the encoder-decoder framework, the GPT model analyses the input using self-attention mechanisms(captures dependencies and relationships within input sequence). It dynamically attends to different parts of the input, considering the entire context for a more comprehensive understanding.
* Encoder Processing: The input embeddings, which are mathematical representations of words, are processed through the encoder component of the transformer. The encoder separates words into embeddings and assigns weights to indicate their relevance in the context.
* Fixed-Length Vector Representation: The encoder generates a fixed-length vector representation, known as an embedding, capturing the contextual information from the input sequence. This representation is then used by the decoder module.
* Decoding Process: The decoder utilizes the vector representation to predict the requested output. It employs built-in self-attention mechanisms to focus on different parts of the input, making predictions based on the contextual understanding.
* Parallel Processing: Unlike its predecessors, transformers, including GPT models, are highly parallelizable. They process the entire input simultaneously during the learning cycle, improving efficiency and enabling faster computation.
* Response Generation: The GPT model generates a response based on its understanding of the input prompt, incorporating context and providing a human-like text output. This response can be in the form of natural language text, making it suitable for a variety of language-related tasks.
* Decoding and Presentation: The generated output, initially in tokenized form, is decoded back into human-readable text. This final output is then presented as the API response to the user.
* Iterative Interaction and Customization: We can iterate the process by sending additional prompts to the GPT API, creating a conversational or interactive experience. Additionally, developers have the option to customize the model for specific tasks by providing a few examples for fine-tuning.

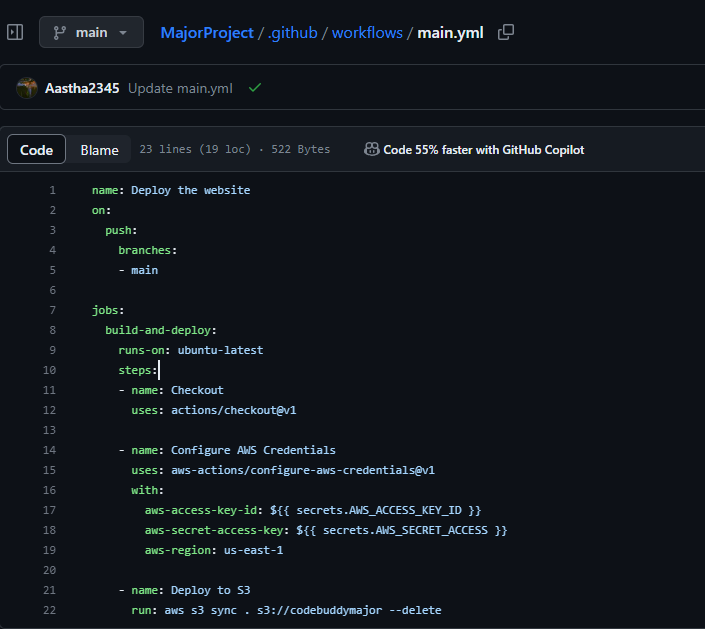
Now the second part of it which is GitHub Actions and deployment.

1. Created a S3 bucket where the website will be hosted with the name as codebuddymajor.

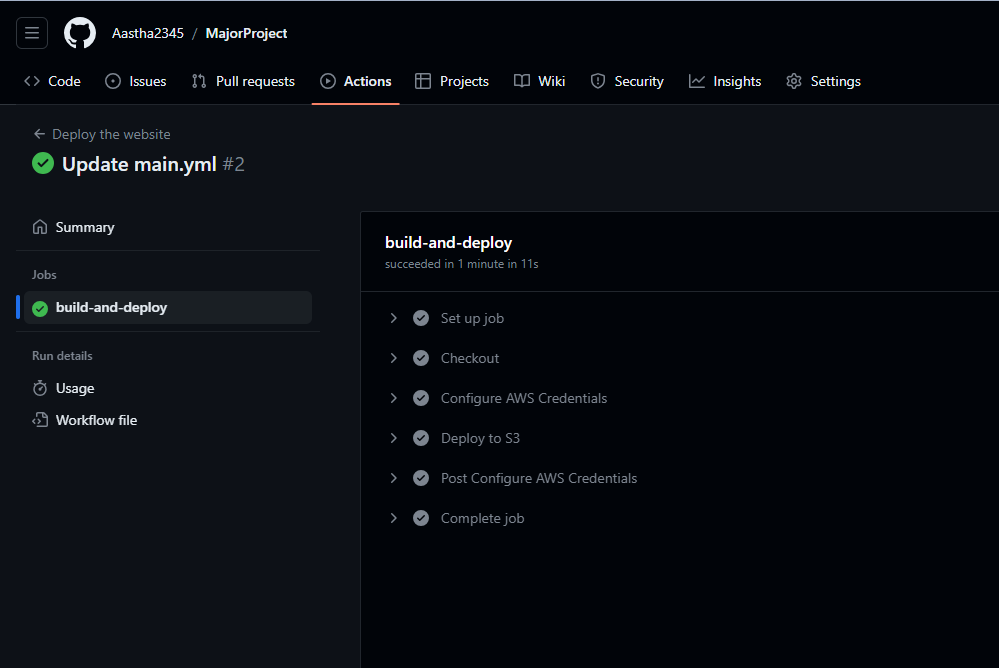


1. Now used GitHub Actions to deploy it to AWS. Wrote a main.yml file in which all the actions are defined. So, whenever there is any change in code it will commit it making seamless continuous integration and continuous deployment.

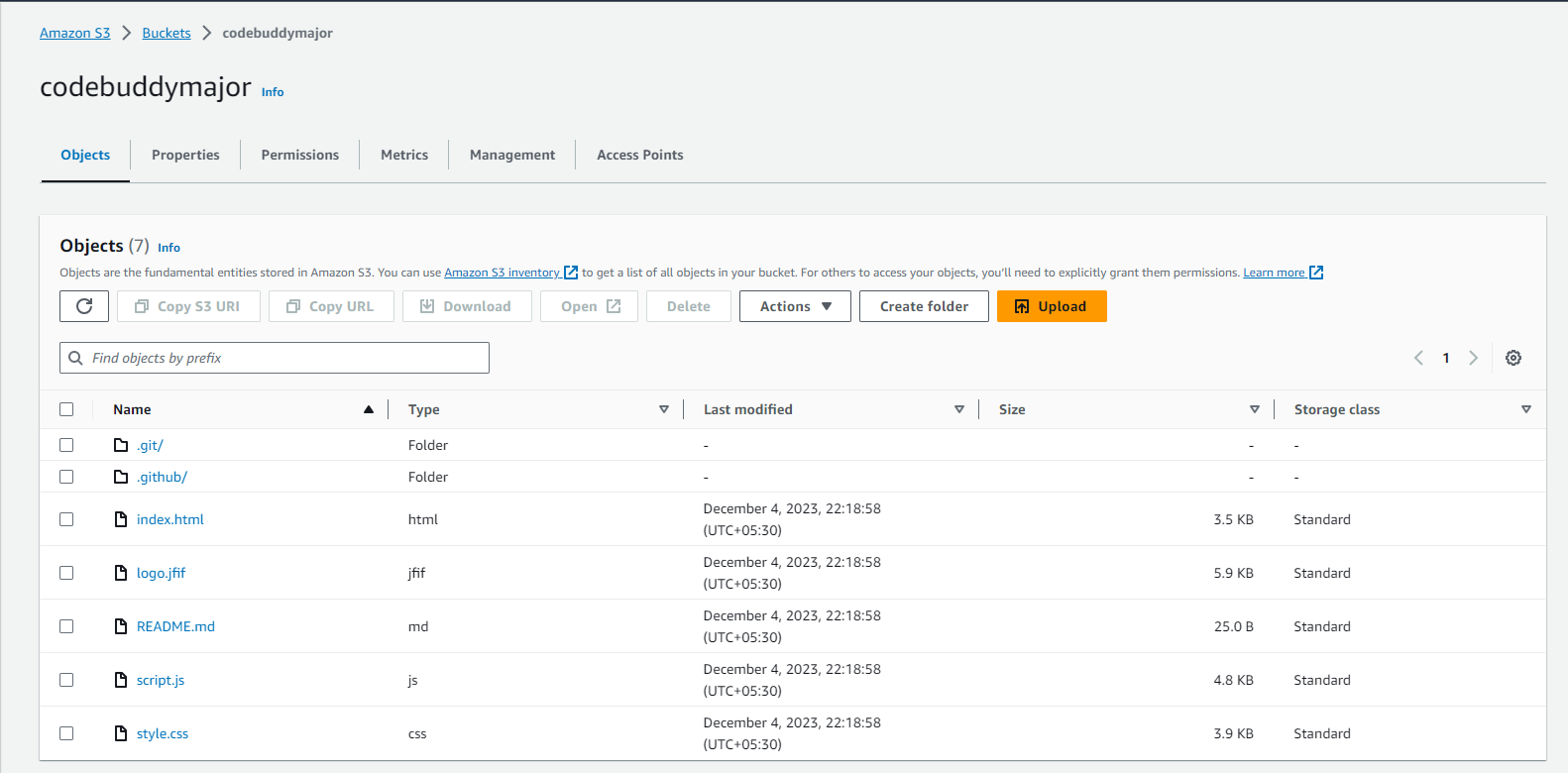
The steps are : name, on: push, jobs and run the steps in that which will configure the AWS credentials to authorize the account and then deploy the website to the S3 bucket which is been created previously.



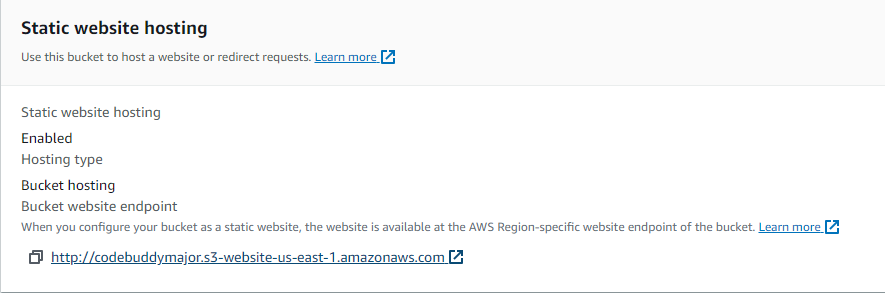
1. As soon as we commit the changes it will start running all the jobs that is defined in the yml file.



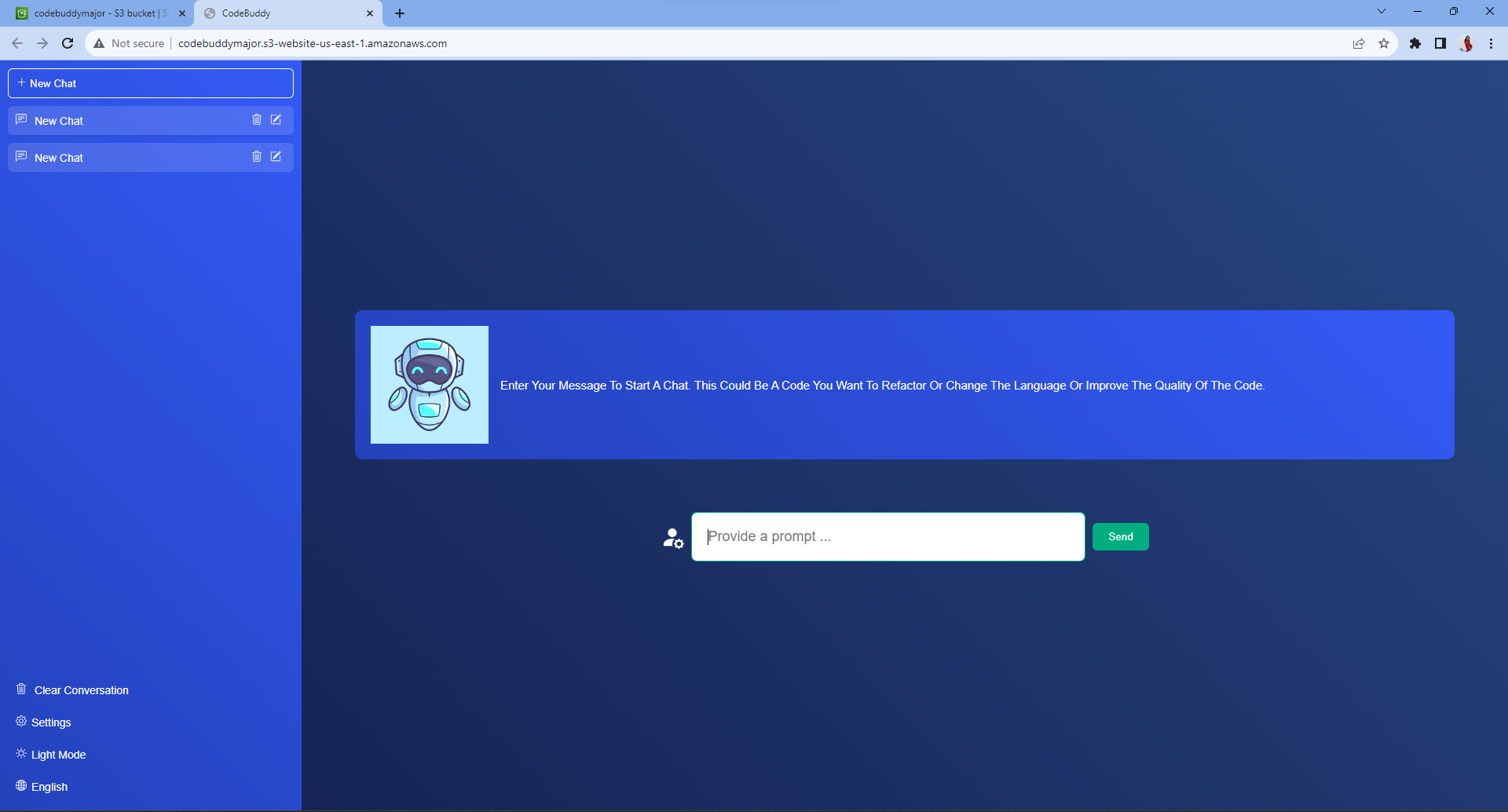
1. Now in the bucket it is clearly visible that the website code is been deployed with the same action without even uploading the files to it. That’s the magic of it.



1. Now we can access our website from anywhere through the link <http://codebuddymajor.s3-website-us-east-1.amazonaws.com> .



1. The website i.e., CodeBuddy is now globally accessible to the world you just need the internet connection.



**3.1. Reference Algorithm**

* GitHub Actions:

GitHub Actions is a CI/CD (Continuous Integration/ Continuous Deployment) platform for automating the builds, test, and deployment process[workflow]. Using GitHub actions, we can build and test every pull request in the repository using workflows, or push the merged pull requests to production with workflows.

The role of GitHub Actions comes into the scene after step 2. When you’ve finished coding the project, upload it to GitHub and build a script to test it every time you commit a change or someone else creates a pull request. If any exception occurs, GitHub Actions will fail the build, preventing the modifications from being merged. Otherwise, the build will succeed, and the modified project will be available for download. The actions are built in a YAML file inside .github/workflow folder

It contains few components like workflows, events, jobs, runners and actions which plays and important role and does whole process internally.

* AWS:

AWS (Amazon Web Services) is a comprehensive, evolving cloud computing platform provided by Amazon that includes a mixture of infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS) and packaged-software-as-a-service (SaaS) offerings. AWS services can offer an organization tool such as compute power, database storage and content delivery services.

* Free tier Maximum limit for EC2:

Amazon EC2 is a web service that provides sizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.

12 months free – 750 hours per month of Linux for t2.micro or t3.micro instance dependent on region

750 hours per month of windows for t2.micro or t3.micro instance dependent on region.

* Free tier Maximum limit for S3:

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance.

12 months free 5 GB in the S3 Standard storage class.

t2.micro instances typically provide 1 GB of RAM.

20,000 Get Requests and 2,000 Put, Copy, Post, or List Requests

**3.2. SWOT Analysis**

Strengths:

* Code Quality: CodeBuddy serves as a valuable platform for learning and enhancing coding quality, making it a resource for users to improve their codes.
* GitHub Integration: Seamless integration with GitHub Actions streamlines deployment, which is a valuable feature for users who want to automate their development processes.
* Diverse User Base: CodeBuddy attracts a diverse range of users, including beginners, students, professionals, and coding enthusiasts, creating a broad and supportive community.
* Seamless Deployment: Deployment on AWS and is scalable and available to everyone.

Weaknesses:

* Content Quality: Ensuring the quality and accuracy of code shared on the platform can be a challenge because it is based on the user’s input.
* Technical Challenges: Technical complexities may arise from the integration with GitHub Actions and the need to manage user-generated content and code.

Opportunities:

* Education and E-Learning: The growing demand for online coding education and e-learning platforms provides an opportunity for CodeBuddy to expand its user base.
* Monetization: Introducing premium features or subscriptions for advanced users or additional services can be a revenue generation opportunity.

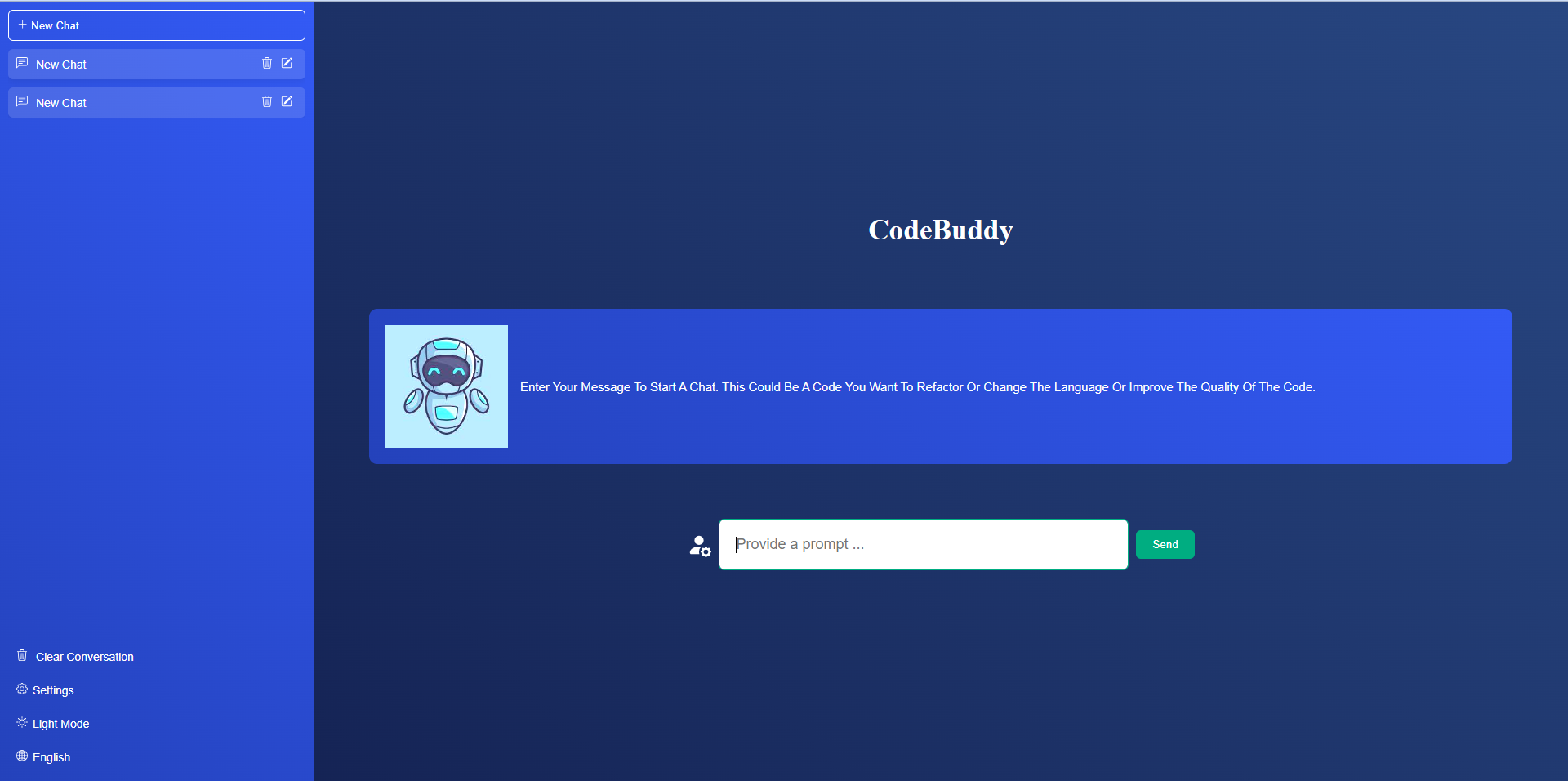
Threats:

* User Data Privacy: Concerns about data privacy and security can lead to regulatory challenges and affect user trust.
* User Churn: If users do not find value or engagement on the platform, they may leave, causing churn.

**3.3. Project Features**

* Seamless GitHub Actions Integration: CodeBuddy offers a cutting-edge feature with its seamless integration with GitHub Actions. This integration allows developers to automate various aspects of their software development workflows, including building, testing, and deploying code. By tapping into the power of GitHub Actions, developers can significantly improve efficiency and reduce manual tasks, ensuring that the code development and deployment processes are faster and more reliable.
* Optimized Testing Processes: CodeBuddy is equipped with tools and capabilities that enable developers to optimize their testing processes. With the platform's assistance, developers can easily configure and manage automated testing suites, ensuring that code is thoroughly tested for quality and reliability. This leads to improved code quality and a higher level of confidence in the software being developed.
* Code Quality Enhancement: One of the cores focuses of CodeBuddy is enhancing code quality. The platform provides developers with guidance and best practices for writing clean, maintainable code. It offers code analysis and quality checks as part of the workflow, enabling developers to identify and address code issues early in the development process. This results in higher-quality code and more efficient development cycles.
* User-Friendly Interface: CodeBuddy's user-friendly interface simplifies the configuration and management of GitHub Actions workflows. This accessibility is a key feature that ensures both beginners and experienced developers can make the most of the platform. The intuitive design and clear documentation make it easy for developers to get started with GitHub Actions, even if they are new to the tool.
* Streamlined Deployment: Efficient and error-free deployment is essential in modern software development. CodeBuddy assists developers in automating the deployment pipeline, reducing the need for manual intervention. This results in quicker and more reliable deployments, helping developers keep up with the rapid pace of software development.
* Accessibility for All Skill Levels: CodeBuddy is designed to cater to developers of all skill levels. Whether you are a beginner just starting out in coding or an experienced professional, the platform's features and resources are accessible and valuable. This inclusivity promotes learning and collaboration among developers with varying levels of expertise.

**3.4. Design Diagrams**



**4. System Requirements**

**4.1 User Interface:**

* Supported Browsers: The user interface would be accessible and fully functional on modern web browsers such as Google Chrome, Mozilla Firefox, Apple Safari, and Microsoft Edge. Responsive Design: The user interface should be responsive to different screen sizes and devices, ensuring a consistent and user-friendly experience on desktops, tablets, and mobile phones.
* Accessibility: The user interface should adhere to accessibility standards (e.g., WCAG) to ensure that it is usable by individuals with disabilities. This includes support for screen readers, keyboard navigation, and other accessibility features.
* Interactive Elements: The UI should incorporate interactive elements, including forms for user registration, code editing, and communication features like messaging or discussion boards.

**4.2 Software Interface:**

* Programming Languages: Used HTML, CSS, JavaScript as the programming language to build the website.
* Third-Party Integrations: Used Open AI Apis to bring data for the backend.
* GitHub Actions: Deployed the website using github actions making continuous integration and deployment process seamless and automatic.

**5. Non-Functional Requirements**

**5.1. Performance Requirements:**

* Responsiveness: CodeBuddy should provide a responsive user interface with low-latency interactions, ensuring that users experience minimal delays when interacting with the platform.
* Scalability: The system must be designed to scale horizontally to accommodate increased user loads without significant degradation in performance. It should support more concurrent users as the user base grows.
* Throughput: The platform should be capable of handling a high volume of concurrent requests, such as code uploads, downloads, and real-time collaboration.
* Resource Utilization: CodeBuddy should optimize resource utilization, ensuring efficient use of server resources, including CPU, memory, and storage, to maintain fast response times.

**5.2 Security Requirements:**

* Data Encryption: All user data, including code snippets and personal information, should be encrypted in transit and at rest to ensure data privacy and security.
* Authentication and Authorization: Users must authenticate securely. Role-based access control (RBAC) should be implemented to manage user access and permissions effectively.
* Secure Communication: All communication between clients and servers should use HTTPS to protect data during transmission.
* Protection Against Common Threats: CodeBuddy must be designed to protect against common web security threats, including SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF) attacks.
* Data Backup and Recovery: Regular data backups and a robust disaster recovery plan should be in place to prevent data loss.

**5.3 Software Quality Attributes:**

* Reliability: CodeBuddy should be highly reliable, with a low rate of system failures or errors. It should be available to users consistently.
* Usability: The platform should be user-friendly and intuitive, with clear navigation and documentation. It should provide a positive user experience, accommodating users of all skill levels.
* Maintainability: CodeBuddy's codebase should be well-structured and maintainable, facilitating future enhancements, bug fixes, and updates.
* Performance Efficiency: The system should perform efficiently, with minimal resource usage, to ensure that users experience fast response times and a smooth interaction with the platform.
* Portability: The platform should be designed for ease of deployment on various hosting environments and infrastructure, ensuring compatibility and flexibility in the choice of servers.
* Scalability: CodeBuddy should be capable of scaling horizontally to handle increased loads without significant degradation in performance.

**6. Methodology**

In the pursuit of creating CodeBuddy, a versatile and innovative platform for code improvement and language flexibility, we have devised a structured methodology to guide our development process. This presentation outlines the step-by-step approach we will follow to bring CodeBuddy to life. Our methodology encompasses key stages, from initial conceptualization to deployment, highlighting the technologies and tools we will employ for the creation of this invaluable resource.

**1. Conceptualization and Objectives:**

* Objective Definition: We begin by clearly defining the objectives of CodeBuddy. Our primary aim is to provide a comprehensive platform for code improvement, code quality enhancement, and language flexibility.
* User-Centric Approach: We place a strong emphasis on understanding the needs and pain points of developers, ensuring that CodeBuddy addresses their challenges effectively.

**2. Technology Selection: Website:** We have chosen Html, CSS, Java script as the core programming language, for its robustness, scalability, and ease of development. This selection forms the foundation of our web application.

**3. Design and Architecture:** A user-friendly and intuitive UI is essential, so we created a design for our website. We also made a few use cases for our website which will be the core and main aim of our project.

**4. Development:**

* Front-End Development: We will develop the front-end of CodeBuddy using HTML, CSS, Java Script languages.
* Integration of OpenAI APIs: To empower CodeBuddy with advanced code analysis capabilities, we integrate OpenAI APIs, leveraging their language models and processing capabilities.

**5. Deployment and Continuous Integration:**

* Deployment Environment: CodeBuddy will be hosted on the AWS platform, providing scalability and reliability.
* GitHub Actions: We will implement Continuous Integration/Continuous Deployment (CI/CD) pipelines using GitHub Actions to automate testing and deployment processes.

**7. Results**

CodeBuddy is a user-friendly website designed for developers to collaborate and enhance their code. Developers can easily input their code and then choose from a range of helpful options. They can opt to refactor their code, seamlessly switch the programming language to another of their choice, and even add comments to improve code quality. CodeBuddy is built with convenience in mind, making it a valuable resource for developers seeking to enhance their code. The platform is also integrated with GitHub Actions for smooth deployment, allowing developers to effortlessly showcase their work. Whether you're a seasoned coder looking to improve your codebase or a novice seeking guidance, CodeBuddy is your go-to tool for code enhancement and collaboration.

**8. References**

<https://platform.openai.com/overview>

<https://docs.github.com/en/actions>