Jiphy the Social Media App for Developers

[1] Shivansh Srivastava, [2]Sajal Tyagi, [3]Ishan Dixit
[1][2][3] Galgotias University

Abstract: This research paper talks about a blogging site specially made for developers. It has all the features in social media site like sharing, posting, and interacting with other people. What makes our website different from other is it is specifically targeted to developers. There are only a few other alternative where developers can interact with other fellow developers and work on same code base at real time.

Keywords: React, nodejs, firebase, mongoDB, Heroku, Docker, Tensorflow, AWS

I. INTRODUCTION

Nowadays, there are many platforms where people can ask for help in their code communication to each other but then too there is something missing you upload the question get a solution it doesn't work for you may be a dependency error system error or maybe some package you ask the question again to solve this problem we have designed website where you can ask for help using the online code editor which uses AI for auto-completion get your answers to work together in the same code base and it is also a social website where you could interact with other people share post message others we have designed our website in such a way that it can be maintained easily it is scalable and monitored

Website development process

The web development process goes like first we design the backend with all the API needed for the social media part of the website and then the code editor and messaging feature then comes creating and implementing the machine learning model to our code editor the deploying everything to different micro-services

1. Analysis
2. Requirements
3. Content Writing
4. Coding

Analysis:- The languages required for this project is javascript and python the framework used for the frontend is react js and the backend is based on node js and express server for chat feature socket.io is used for user authentication firebase service is used and data of the user is stored in MongoDB database the package manager used for frontend and backend is npm and for the machine learning part machine learning model is made on development environment and deployed on AWS services to use

Content Writing: Our website contains a blog section where the website new feature and updates are published if a user is new to our website to explain how everything works they can take the help of our blogs and get started these blogs should be easy to understand even for a beginner so content wring is a key aspect of our website.

Requirements: The coding phase started by creating user and post models and routed and then working on the frontend in parallel and implementing these features using Axios to fetch data from MongoDB and authenticate the user using firebase then implementing chat feature by using sockets for individual chat and chat rooms and then creating a basic code editor and compiler after that creating a docker image and pushing it on docker hub after that coms creating ML model by gathering datasets using TensorFlow unsupervised learning then uploading it to AWS service and used it in our website compiler.

Coding: We containerized our application using docker and used Heroku for deployment backend and frontend separately and for caching error we also connected Redis container and used AWS SagaMaker to deploy our ML model.

ISSN (Online) 2394-6849
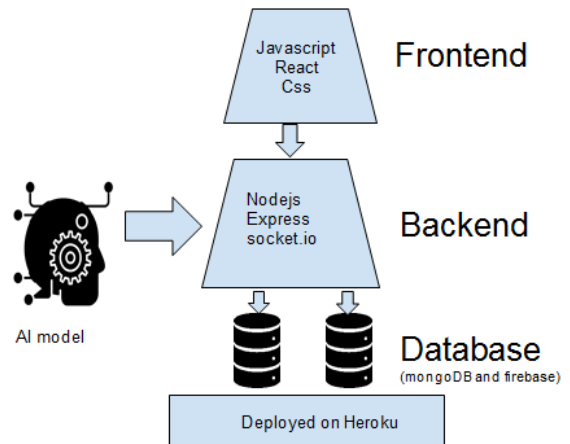
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 8, Issue 7, July 2021

Fig:1 Architecture

II. LITERATURE REVIEW

The problem we are trying to solve is the lack of a platform for developers where they can get all the services like asking about code error, sharing code snippets, working on a code editor online with different developers in real-time, creating new connections, and sharing their experience on a social platform--everything at one place. There are a few alternatives that exist, but the difference is Jiphy provides all the features such as social media platform, code editor and compiler, social forum, blog post, and Q&A section on a single platform. When a user registers to Jiphy, they experience a clean UI, FAQ sections on how the website works, and a few blog posts to get started on our website. People can use it as a platform to increase connections or can also use it to work on the same code base in real-time with multiple developers. We have used MERN stack in web development and firebase for authentication. We have also integrated a machine learning model in our code compiler which helps in auto compilation. Everything is hosted as a separate microservice. The web part is containerized and hosted on Heroku and the machine learning model is hosted on AWS SagaMaker. The future scope of our project would be making it open source and asking other developers for their contribution and new ideas a few of them will be a section for tech news updates and a section for learning chatbots to answer your tech questions.

III. RESULTS

The website was tested by some people, and it worked fine. There were a few bugs which were resolved at the time of production. We received positive reviews from people.
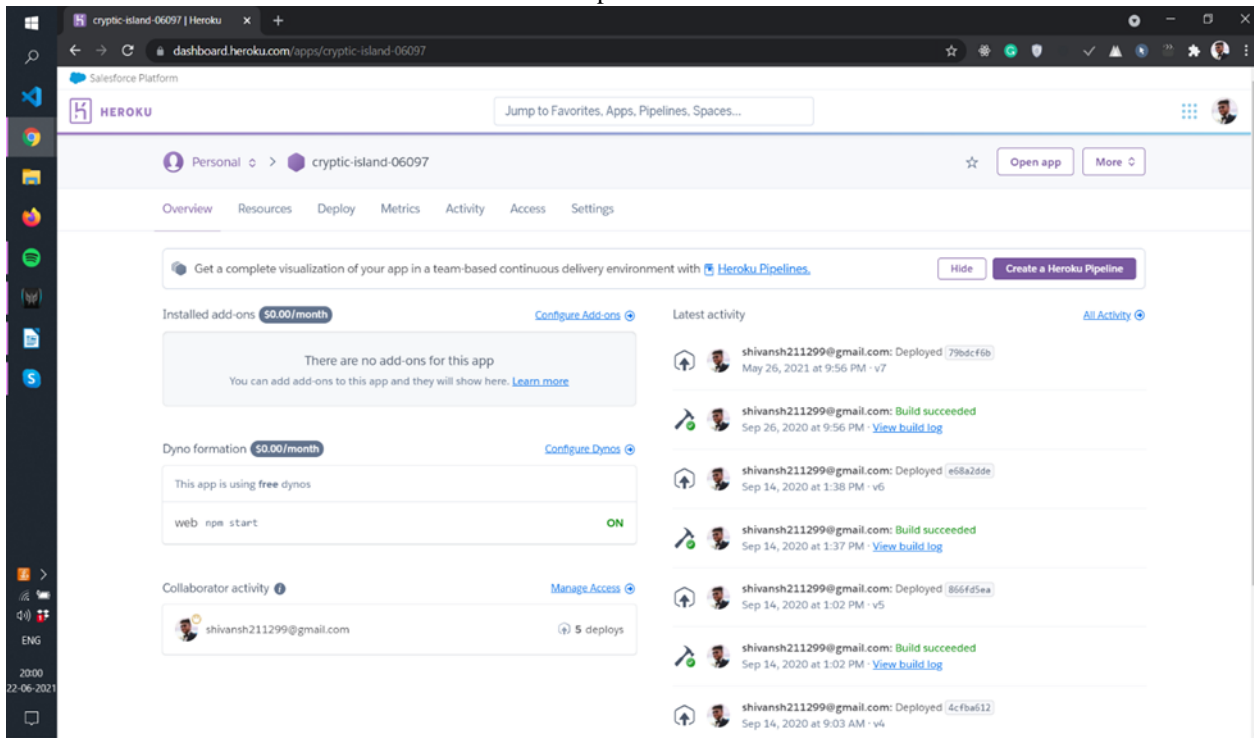
IV.FUTUREWORK

Currently, the UI of our website is not so attractive GSAP animations can be added the website is made to be scalable we can add a monitoring feature and deploy it on Kubernetes so containers can be managed easily we can also include a chatbot that will use NLP behind the scene

V.CONCLUSION

We were able to deploy our website and everything works fine post are being posted code compiler and editor are working properly multiple users are working on the same code base at real-time people are using the social forum feature for asking doubts we will keep improving our

features

VI. REFERENCE

[1] Park, Dae Hoon, and Rikio Chiba. "A Neural Language Model for Query Auto-Completion." Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017. Crossref, doi:10.1145/3077136.3080758.

[2]"Introduction." Socket.IO, 24 June 2021, socket.io/docs/v4/index.html.

[3] "Refs and the DOM –." React, reactjs.org/docs/refs-and-the-dom.html.

[4] "Module: Tf | TensorFlow Core v2.5.0." TensorFlow, www.tensorflow.org/api_docs/python/tf.

[5]Node.Js. "Documentation." Node.Js, nodejs.org/en/docs.

[6]Docker. "Docker Docs." Docker, https://docs.docker.com.