

##### Institute of Engineering & Technology

##### GLA University

**Mathura- 281406, INDIA**

**2022-23**

**Mini Project Report**

**MY API**



**Department of computer Engineering**

**Applications GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha, Mathura – 281406**

##### Declaration

We hereby declare that the work which is being presented in this Project

“**MY API”,** done at place where the project is done, has not been in any case duplicatedtosubmittoanyotheruniversityfortheawardofanydegree.Tothebest of my knowledge other than me, no one has submitted to any other university. This is an authentic record of our own work carried by the team members under the supervision of our mentor Gurpreet Kaur

###### Group Members :

###### Ishan Joshi(201500305)

* Harsh Vardhan, (201500279 )

Divyanshu Singh , (201500234)

Course: B.Tech (Computer Science and Engineering)

Supervised By:

Gurpreet KaurTechnical Trainer, GLA University, Department of Computer Engineering & Application

Contents

[Acknowledgement 3](#_Toc120458551)

[Contents 4](#_Toc120458552)

[Introduction 5](#_Toc120458553)

[Javascript 7](#_Toc120458554)

[HTML 8](#_Toc120458555)

CSS 9

Vite.js ………………………………………………………………………

NodeJS…………………………………………………………………….

Open AI …………………………………………………………………..

Snapshots…………………………………………………………………10

[Conclusion 19](#_Toc120458558)

# Acknowledgement

We thank the almighty for giving us the courage and perseverance in completing the project. This project itself is an acknowledgement for all those people who have given us their heartfelt co-operation in making this project a grand success. We extend our sincere thanks to Gurpreet Ma’am, Technical Trainer at

“GLA University, Mathura” for providing his valuable guidance at every stage of this project work. We are profoundly grateful towards the unmatched services rendered by him. And last but not least, we would like to express our deep sense of gratitude and earnest thanks giving to our dear parents for their moral support and heartfelt cooperation in doing the main project.

## **Introduction**

An API, or application programming interface, is a set of defined rules that enable different applications to communicate with each other. It acts as an intermediary layer that processes data transfers between systems, letting companies open their application data and functionality to external third-party developers, business partners, and internal departments within their companies.

The definitions and protocols within an API help businesses connect the many different applications they use in day-to-day operations, which saves employees time and breaks down silos that hinder collaboration and innovation. For developers, API documentation provides the interface for communication between applications, simplifying application integration.

***Objectives***

* To help and solve problems
* To learn logical attitude.
* To develop the skill of answering precisely.
* To make to think and answer quickly.
* To consolidate and rapid revision of logical facts.
* To develop a situation for children to understand and to answer immediately.
* To exercise the practice of brief answering in the limited time.
* To get equipped with useful and important facts.
* To motivate thepeople gather knowledge.
* To faster creative instincts among people.

**Technologies Used**

**Javascript**

avaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as **LiveScript,** but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name **LiveScript**. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

It is used for:

* WEB applications
* Desktop applications
* Web servers and application servers
* Games
* And much, much more!

## **Advantages of JavaScript**

The merits of using JavaScript are −

* **Less server interaction** − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
* **Immediate feedback to the visitors** − They don't have to wait for a page reload to see if they have forgotten to enter something.
* **Increased interactivity** − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
* **Richer interfaces** − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

HTML -

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages. • Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext. • As its name suggests, HTML is a Markup Language which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display. Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.There are three important characteristics of XML that make it useful in a variety of systems and solutions −

**Features of HTML:**

* It is easy to learn and easy to use.
* It is platform-independent.
* Images, videos, and audio can be added to a web page.
* Hypertext can be added to the text.
* It is a markup language.

**Advantages:**

* HTML is used to build websites.
* It is supported by all browsers.
* It can be integrated with other languages like CSS, JavaScript, etc.

Node JS

## **Introduction:** Node.js is an open-source and cross-platform runtime environment for executing [JavaScript](https://www.geeksforgeeks.org/JavaScript-tutorial/) code outside a browser. You need to remember that **NodeJS is not a framework and it’s not a programming language**. Most people are confused and understand it’s a framework or a programming language. We often use Node.js for building back-end services like APIs like Web App or Mobile App. It’s used in production by large companies such as Paypal, Uber, Netflix, Walmart, and so on.

**Features of NodeJS:**There are other programming languages also which we can use to build back-end services so what makes Node.js different I am going to explain.

1. It’s easy to get started and can be used for prototyping and agile development
2. It provides fast and highly scalable services
3. It uses JavaScript everywhere, so it’s easy for a JavaScript programmer to build back-end services using Node.js
4. Source code cleaner and consistent.
5. Large ecosystem for open source library.
6. It has Asynchronous or Non-blocking nature.

**Advantages of NodeJS:**Here are the benefits of using Node.js 

1. **Easy Scalability:** Developers prefer to use Node.js because it is easily scaling the application in both horizontal and vertical directions. We can also add extra resources during the scalability of the application.
2. **Real-time web apps:** If you are building a web app you can also use PHP, and it will take the same amount of time when you use Node.js, But if I am talking about building chat apps or gaming apps Node.js is much more preferable because of faster synchronization. Also, the event loop avoids HTTP overloaded for Node.js development.
3. **Fast Suite:** NodeJs runs on the V8 engine developed by Google. Event loop in NodeJs handles all asynchronous operation so NodeJs acts like a fast suite

and all the operations can be done quickly like reading or writing in the database, network connection, or file system

1. **Easy to learn and code:** NodeJs is easy to learn and code because it uses JavaScript. If you are a front-end developer and have a good grasp of JavaScript you can easily learn and build the application on NodeJS
2. **Advantage of Caching:**It provides the caching of a single module. Whenever there is any request for the first module, it gets cached in the application memory, so you don’t need to re-execute the code.
3. **Data Streaming:** In NodeJs HTTP request and response are considered as two separate events. They are data stream so when you process a file at the time of loading it will reduce the overall time and will make it faster when the data is presented in the form of transmissions. It also allows you to stream audio and video files at lightning speed.
4. **Hosting:** PaaS (Platform as a Service) and Heroku are the hosting platforms for NodeJS application deployment which is easy to use without facing any issue.
5. **Corporate Support:** Most of the well-known companies like Walmart, Paypal, Microsoft, Yahoo are using NodeJS for building the applications. NodeJS uses JavaScript, so most of the companies are combining front-end and backend Teams together into a single unit.

# ­­­­

Vite.js- Vite (French word for "quick", pronounced /vit/, like "veet") is a build tool that aims to provide a faster and leaner development experience for modern web projects. It consists of two major parts:

* A dev server that provides [rich feature enhancements](https://vitejs.dev/guide/features.html) over [native ES modules](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Modules), for example extremely fast [Hot Module Replacement (HMR)](https://vitejs.dev/guide/features.html#hot-module-replacement).
* A build command that bundles your code with [Rollup](https://rollupjs.org/), pre-configured to output highly optimized static assets for production.

Vite is opinionated and comes with sensible defaults out of the box, but is also highly extensible via its [Plugin API](https://vitejs.dev/guide/api-plugin.html) and [JavaScript API](https://vitejs.dev/guide/api-javascript.html) with full typing support.

You can learn more about the rationale behind the project in the [Why Vite](https://vitejs.dev/guide/why.html) section.

## Browser Support[**​**](https://vitejs.dev/guide/#browser-support)

The default build targets browsers that support [native ES Modules](https://caniuse.com/es6-module), [native ESM dynamic import](https://caniuse.com/es6-module-dynamic-import), and [import.meta](https://caniuse.com/mdn-javascript_operators_import_meta" \t "_blank). Legacy browsers can be supported via the official [@vitejs/plugin-legacy](https://github.com/vitejs/vite/tree/main/packages/plugin-legacy) - see the [Building for Production](https://vitejs.dev/guide/build.html) section for more details.

## Trying Vite Online[**​**](https://vitejs.dev/guide/#trying-vite-online)

You can try Vite online on [StackBlitz](https://vite.new/). It runs the Vite-based build setup directly in the browser, so it is almost identical to the local setup but doesn't require installing anything on your machine. You can navigate to vite.new/{template} to select which framework to use.

The supported template presets are:

| **JavaScript** | **TypeScript** |
| --- | --- |
| [vanilla](https://vite.new/vanilla) | [vanilla-ts](https://vite.new/vanilla-ts) |
| [vue](https://vite.new/vue) | [vue-ts](https://vite.new/vue-ts) |
| [react](https://vite.new/react) | [react-ts](https://vite.new/react-ts) |
| [preact](https://vite.new/preact) | [preact-ts](https://vite.new/preact-ts) |
| [lit](https://vite.new/lit) | [lit-ts](https://vite.new/lit-ts) |
| [svelte](https://vite.new/svelte) | [svelte-ts](https://vite.new/svelte-ts) |

One thing you may have noticed is that in a Vite project, index.html is front-and-central instead of being tucked away inside public. This is intentional: during development Vite is a server, and index.html is the entry point to your application.

Vite treats index.html as source code and part of the module graph. It resolves <script type="module" src="..."> that references your JavaScript source code. Even inline <script type="module"> and CSS referenced via <link href> also enjoy Vite-specific features. In addition, URLs inside index.html are automatically rebased so there's no need for special %PUBLIC\_URL% placeholders.

Similar to static http servers, Vite has the concept of a "root directory" which your files are served from. You will see it referenced as <root> throughout the rest of the docs. Absolute URLs in your source code will be resolved using the project root as base, so you can write code as if you are working with a normal static file server (except way more powerful!). Vite is also capable of handling dependencies that resolve to out-of-root file system locations, which makes it usable even in a monorepo-based setup.

Vite also supports [multi-page apps](https://vitejs.dev/guide/build.html#multi-page-app) with multiple .html entry points.

#### Specifying Alternative Root[**​**](https://vitejs.dev/guide/#specifying-alternative-root)

Running vite starts the dev server using the current working directory as root. You can specify an alternative root with vite serve some/sub/dir.

Open AI - OpenAI is a private research laboratory that aims to develop and direct artificial intelligence ([AI](https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence)) in ways that benefit humanity as a whole. The company was founded by Elon Musk, Sam Altman and others in 2015 and is headquartered in San Francisco.

OpenAI was created in part because of its founders' existential concerns about the potential for catastrophe resulting from carelessness and misuse of general-purpose AI. The company has a long-term focus on fundamental advances in AI and its capabilities. The founders of the company and other investors started the company with a $1 billion endowment. In February 2018, Elon Musk left the company due to a potential conflict of interest with his work at Tesla, the automotive and clean energy company inspired by Nikola Tesla.

The stated intent of the company -- to work toward safe [artificial general intelligence (AGI)](https://www.techtarget.com/searchenterpriseai/definition/artificial-general-intelligence-AGI) for the benefit of humanity -- is reflected in its objective to freely collaborate with other research organizations and individuals. Research and patents made by the company are intended to remain open to the public except in cases where they could negatively affect safety.

### Timeline and history of OpenAI

OpenAI was originally focused on developing AI and machine learning tools for video games and other recreational purposes. Less than a year after its official founding on December 11, 2015, it released its first AI offering, an open source toolkit for developing [reinforcement learning (RI)](https://www.techtarget.com/searchenterpriseai/definition/reinforcement-learning) algorithms called OpenAI Gym. Over the next two years, OpenAI focused on more general AI development and AI research.

**THIS ARTICLE IS PART OF**

### [What is generative AI? Everything you need to know](https://www.techtarget.com/searchenterpriseai/definition/generative-AI)

* Which also includes:
* [**Bard vs. ChatGPT: What's the difference?**](https://www.techtarget.com/whatis/feature/Bard-vs-ChatGPT-Whats-the-difference)
* [**Pros and cons of AI-generated content**](https://www.techtarget.com/whatis/feature/Pros-and-cons-of-AI-generated-content)
* [**36 AI content generators to explore in 2023**](https://www.techtarget.com/whatis/feature/AI-content-generators-to-explore)

In 2018, OpenAI published a [report](https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf) to explain to the world what a Generative Pre-trained Transformer (GPT) is. A GPT is a neural network, or a machine learning model, created to function like a human brain, and it's trained on input (large data sets) to produce outputs (i.e., answers to users' questions).

In March 2019, OpenAI shifted from nonprofit to for-profit status and became formally known as OpenAI LP, controlled by a parent company, OpenAI Inc. Almost two years later, in January 2021, OpenAI introduced Dall-E, a generative AI model that analyzes natural language text from human users and then generates images based on what is described in the text.

Perhaps the company's best-known product is ChatGPT, released in November 2022 and heralded as the world's most advanced chatbot for its ability to provide answers to users on a seemingly unlimited range of topics. Its benefits and drawbacks, as well as its uses in various industries, are still being debated.

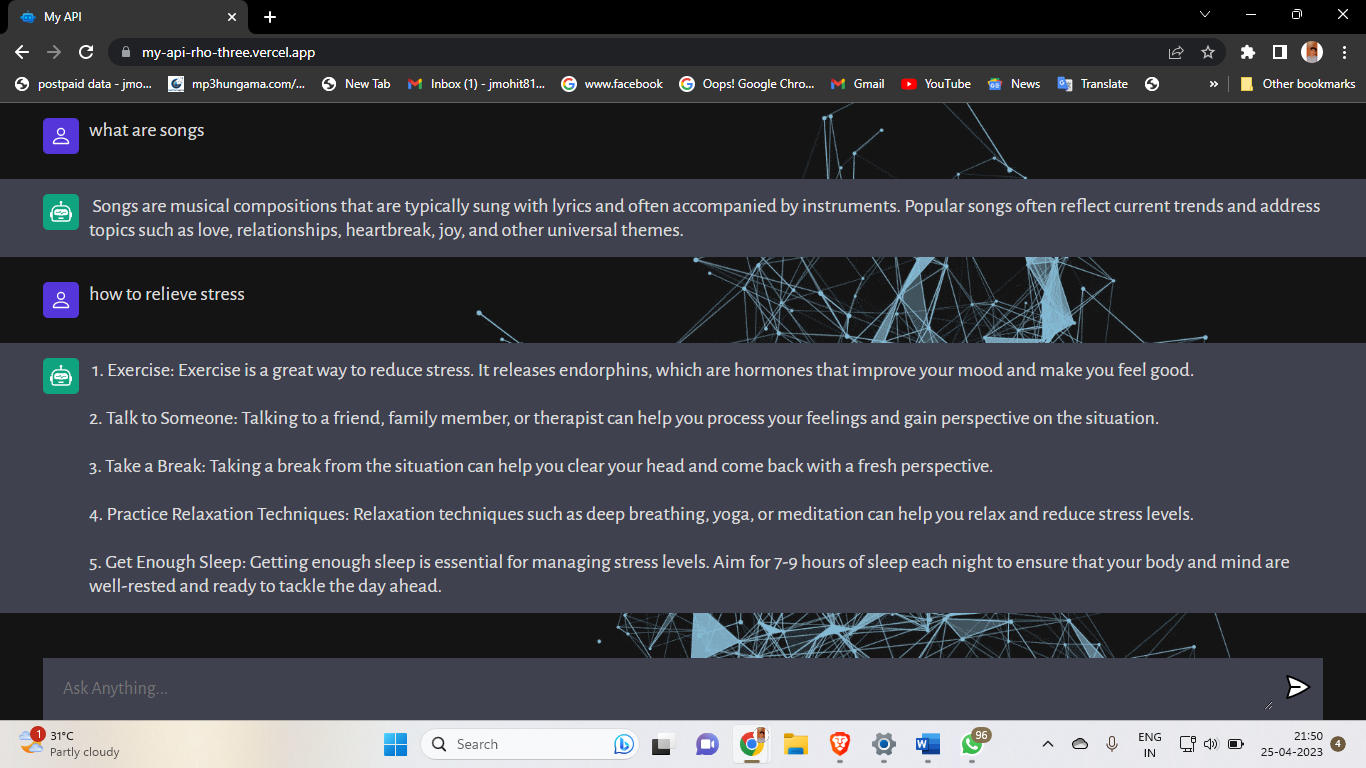
Although Elon Musk no longer serves on the board of the company, co-founder Sam Altman still serves as the company's CEO. Its current president and chairman is Greg Brockman, formerly the CTO of financial services and SaaS company Stripe, and its chief scientist is Ilya Sutskever, formerly of Google.

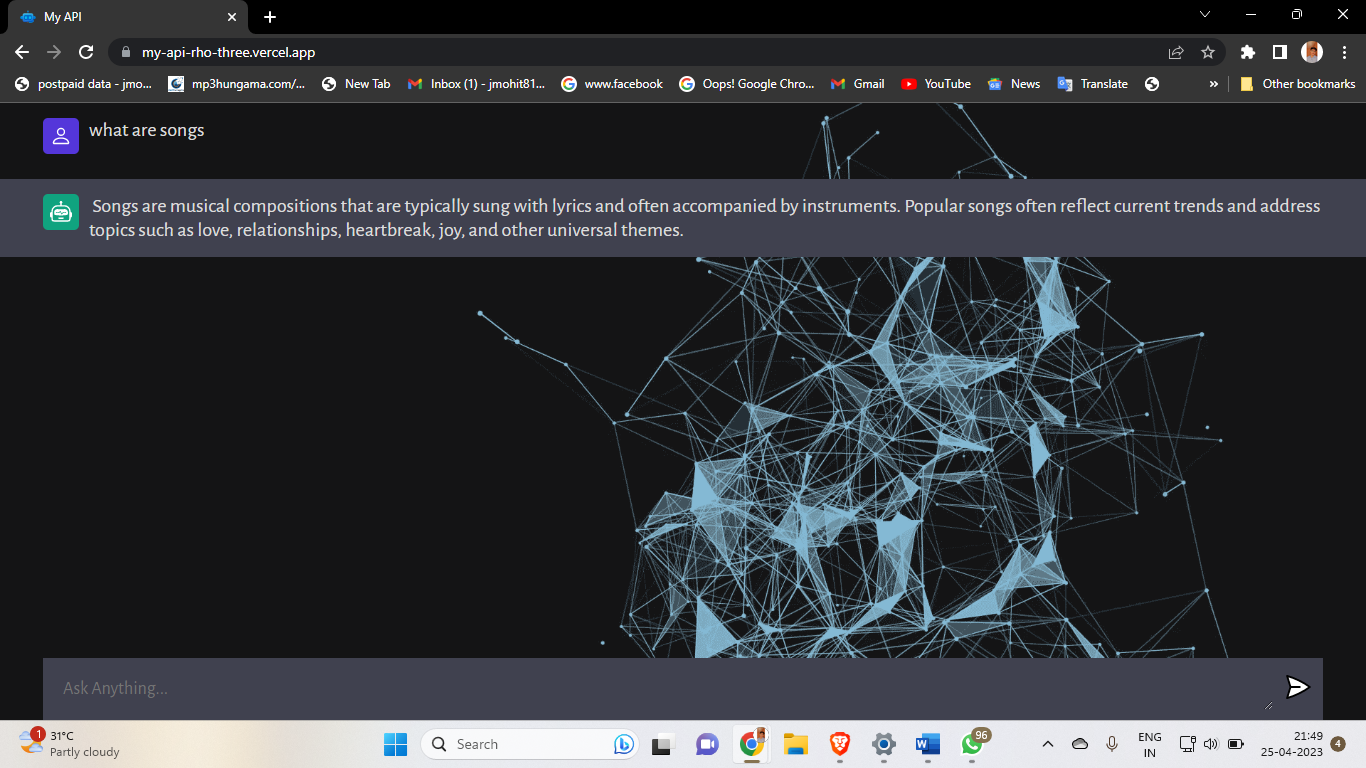
### The future of OpenAI

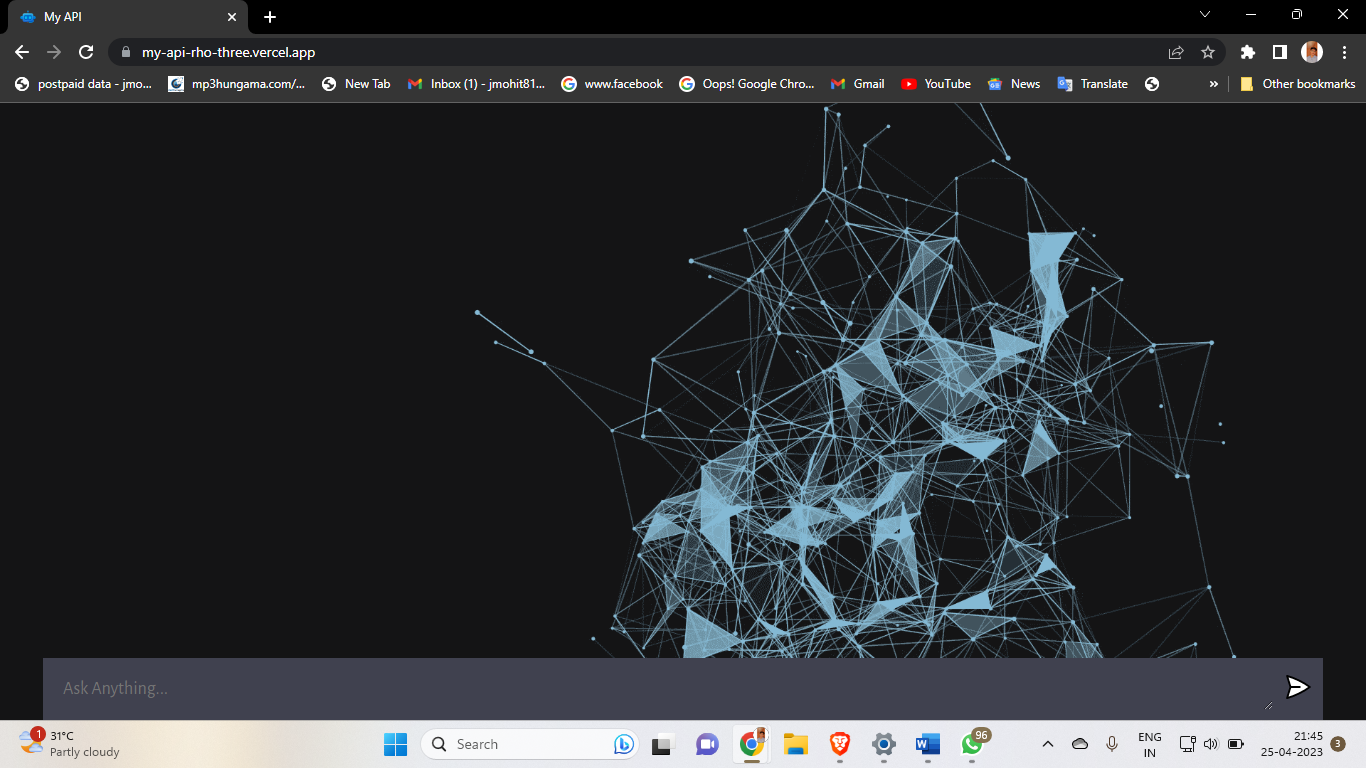
OpenAI has not provided extensive public commentary on future plans, but based on recent investments, democratization of AI is a clear goal of the Microsoft-OpenAI partnership, as nontechnical professionals should soon have more AI tools at their disposal that do not require AI expertise.

In March 2023, OpenAI released the company's newest upgrade in language model technology since GPT-3.5 (the foundation for ChatGPT): [GPT-4](https://www.techtarget.com/searchenterpriseai/news/365532735/OpenAI-releases-latest-version-of-its-star-LLM-GPT-4). GPT-4 has been labeled superior to its predecessors because it delivers multimodal AI functionality, where it can analyze not just text, but also images. Given OpenAI's most recent releases, it's clear the company is giving no indication of slowing down. OpenAI projects it will surpass $1 billion in revenue by 2024.

Snapshots







Conclusion

We have completed our project within time limit with the coordination of our team members under the supervision of our mentor Gurpreet Kaur Ma’am .