MSc (IT) Summer Internship

ArtifyAI

Name :- **Shivam Javiya**
Student ID :- **202312029**
Company Name :- **Mxicoders Pvt Ltd**
Guide Name :- **Paresh Dalsaniya**

Content Index :

1. Introduction

In our technologically advanced world, AI tools have revolutionized how we gather information, solve problems, and create content. **ArtifyAI** serves as an innovative platform that combines the strengths of various AI models, offering users a consolidated and interactive solution to access AI-driven insights. This project report details the development and implementation of **ArtifyAI**, showcasing its potential to transform the way we interact with AI technologies.

Problem Statement

With the proliferation of AI tools, users often face the challenge of navigating multiple platforms to obtain diverse insights. Each platform may provide different answers or solutions, leading to confusion and inefficiency. **ArtifyAI** aims to solve this problem by integrating multiple AI models into a single platform, offering users a comprehensive and cohesive experience.

Objectives

The primary objectives of **ArtifyAI** are:

1. **Integration of AI Tools:** Combine the capabilities of ChatGPT, Google Gemini, and image generation AI to deliver unified solutions across various domains.
2. **User-Friendly Interface:** Design an intuitive and visually appealing UI/UX that enhances user engagement and accessibility.
3. **Efficiency and Convenience:** Streamline the process of accessing AI insights, reducing the time and effort required by users to find accurate information.
4. **Scalability:** Ensure the platform is scalable to accommodate future expansions and integrations with additional AI tools.

Development and Implementation

Features

- **Multi-Model Integration:** Access the strengths of different AI models in a cohesive manner.
- **Interactive UI/UX:** Enjoy a seamless user experience with easy navigation and responsive design.
- **Real-Time Processing:** Receive instant responses and updates, enhancing user satisfaction.
- **Customization Options:** Tailor the platform to specific needs and preferences, ensuring a personalized experience.

2. Functional Requirements

Requirement	**Description**
User Authentication and Authorization	User Registration : The system shall allow users to create an account using their email and a password. Users can register using social media accounts (e.g., Google). **User Login:** The system shall allow users to log in with their email and password or social media accounts.
Query Processing	• The system shall allow users to select from available AI models (e.g., ChatGPT, Google Gemini) to process their queries. • Users can choose a specific model or allow the system to automatically select the best-suited model based on the query type.
Image Generation	**Text Description:** The system shall allow users to input textual descriptions specifying the desired image features. The input field should include guidelines or examples to help users create effective descriptions. The system shall integrate with an AI model capable of generating images based on textual descriptions (e.g., DALL-E, Stable Diffusion).
Multi-Model Responses	• The system shall support integration with multiple AI models to provide diverse insights and responses. • Users can specify which models to include in the response generation or allow the system to select based on query context.
Response Display:	• The system shall present consolidated responses in a structured and visually appealing manner. • Users should be able to navigate through sections or topics within the response.

Admin:

Requireme nt	**Description**
View User Accounts:	The admin should have access to a dedicated page where they can view all pending orders that are yet to be processed or completed.
Edit User Accounts:	The admin should have the authority to mark an order as completed once it has been successfully prepared and deliveredto the user.
Delete User Accounts:	The admin should be able to insert and manage data related tothe menu items, such as adding new items, updating prices, etc.
Search and Filter Users:	The admin should have a page to view and manage all the menuitems available in the app's database.
Assign User Roles:	Admins shall be able to assign or change roles for individual users, controlling their access to specific features and functionalities.

Non-Functional Requirements

Requirement	**Description**
Performance	- Response Time : • The system should provide AI-generated responses within 3 seconds for most queries. • The system should handle up to 1000 concurrent users without performance degradation.
	Scalability: The system should be scalable to accommodate future growth in user base and data volume.
Reliability	**Availability:** The system should have an uptime of 99.9%. The system should implement failover mechanisms to ensure continuous operation during failures.
	Error Handling The system should provide informative error messages and handle exceptions gracefully.
Security	**Data Protection:** The system should encrypt sensitive user data, such as passwords, both at rest and in transit.
	Access Control: The system should implement role-based access control to protect sensitive features and data.

Requirement	**Description**
Usability	Regular Audits: The system should undergo regular security audits and vulnerability assessments.
Maintainability	Code Quality: The system should be developed using clean and maintainable code practices, with proper documentation.
Modular Design:	The system should have a modular architecture to facilitate easy updates and enhancements.

3. Methodology/Processes

Software development Life Cycle Model

Methodology/Process Model: AGILE PROCESS MODEL

The meaning of Agile is swift or versatile. "Agile process model" refers to a software development approach based on iterative development. Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.

Phases of Software Model

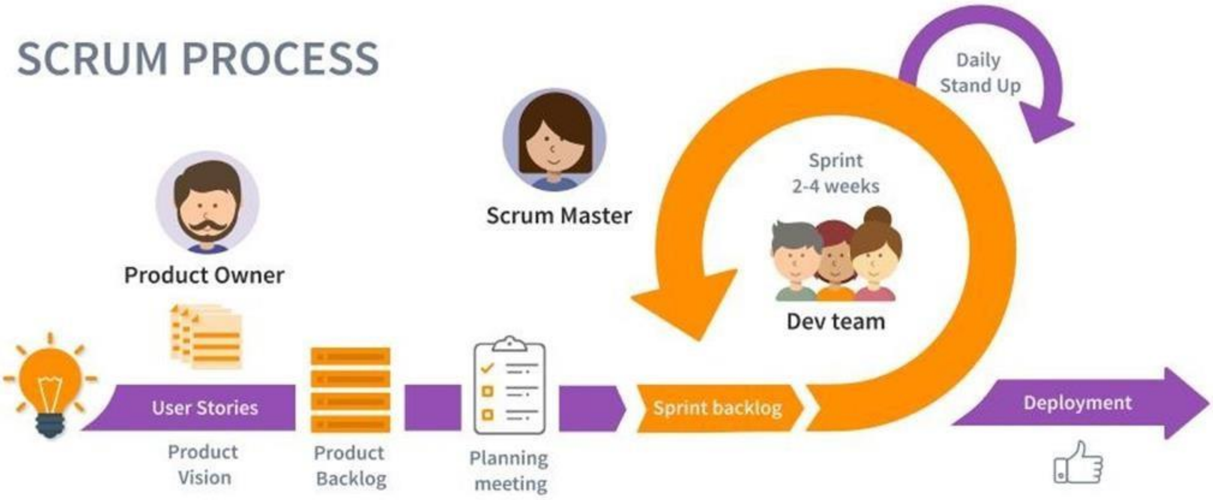
- Requirements gathering
- Design the requirements
- Construction/ iteration
- Testing
- Deployment
- Feedback

Sample User stories

- I want to log in easily : I want to log in using my email and password so that I can access my account and personalize my experience.
- I want to explore AI models : As a user, I want to explore different AI models available on the platform so that I can choose the best one for my needs.
- I want to input queries and receive responses : As a user, I want to input my queries and receive responses from the AI models so that I can get the information or assistance I need.
- I want to generate images from text descriptions : As a user, I want to generate images based on my textual descriptions so that I can visualize my ideas or concepts.

Due to the following reasons, we chose scrum as our Software Development Life Cycle Model:

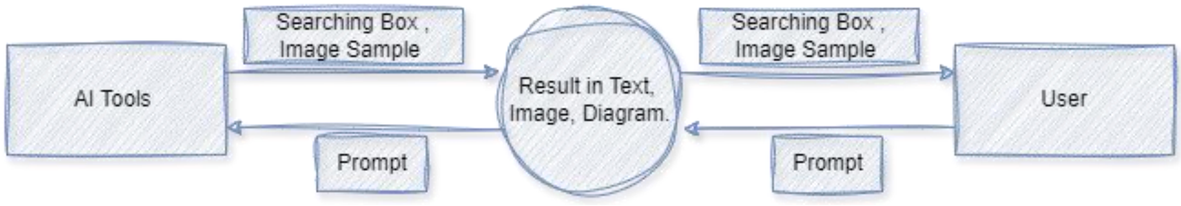
- Meetings with 15 Day Duration (i.e. sprint of 15 Days)
- In the meeting client says, “As an admin, I want a particular functionality” which represents requirement gathering in the form of user stories for which scrum is better suited.
- Meeting then developing then reviewing is a part of the general scrum methodology, which is followed thoroughly for the given project.



4. Design

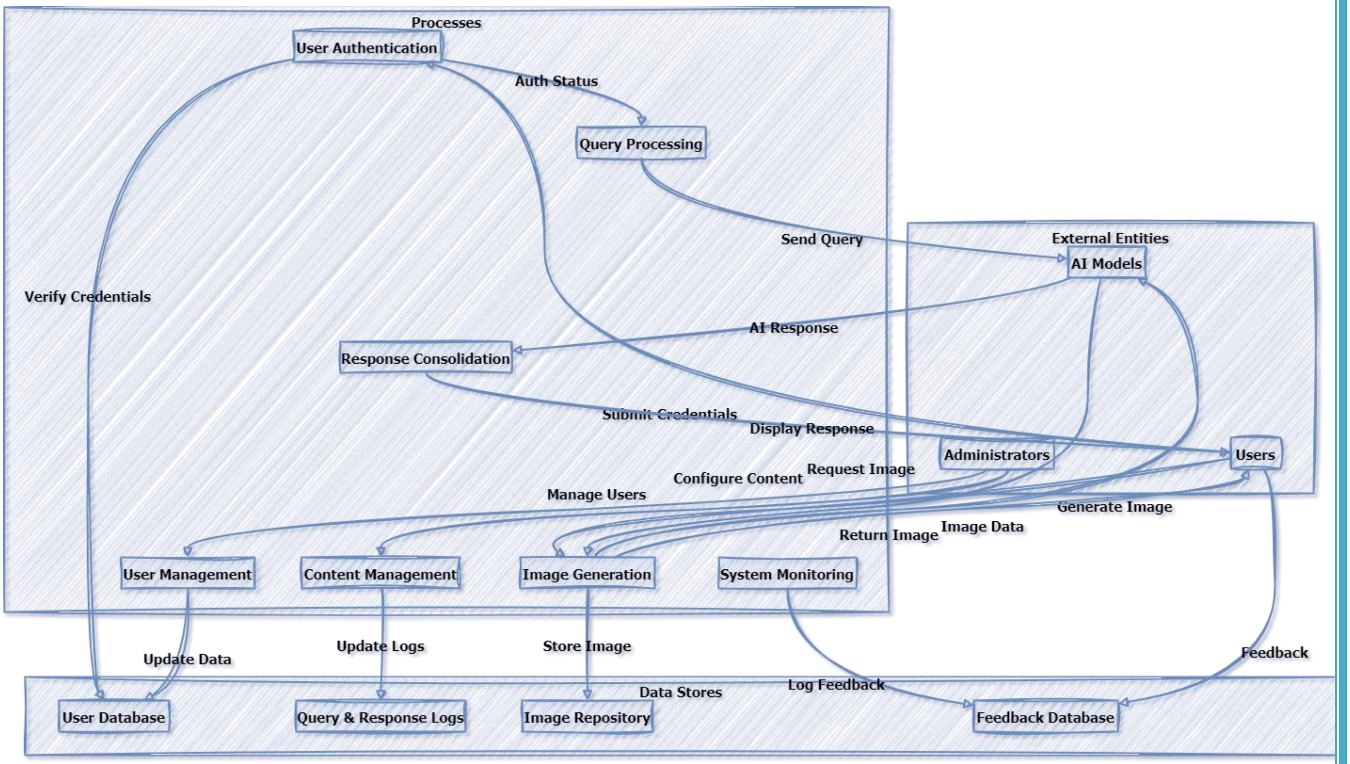
Context level DFD – 0 levels

- The context-level data flow diagram (dfd) describes the whole system. The (0) level dfd describes the all-user module that operates the system. The below data flow diagram of the online shopping site shows the two users who can operate the system Admin and Member user.

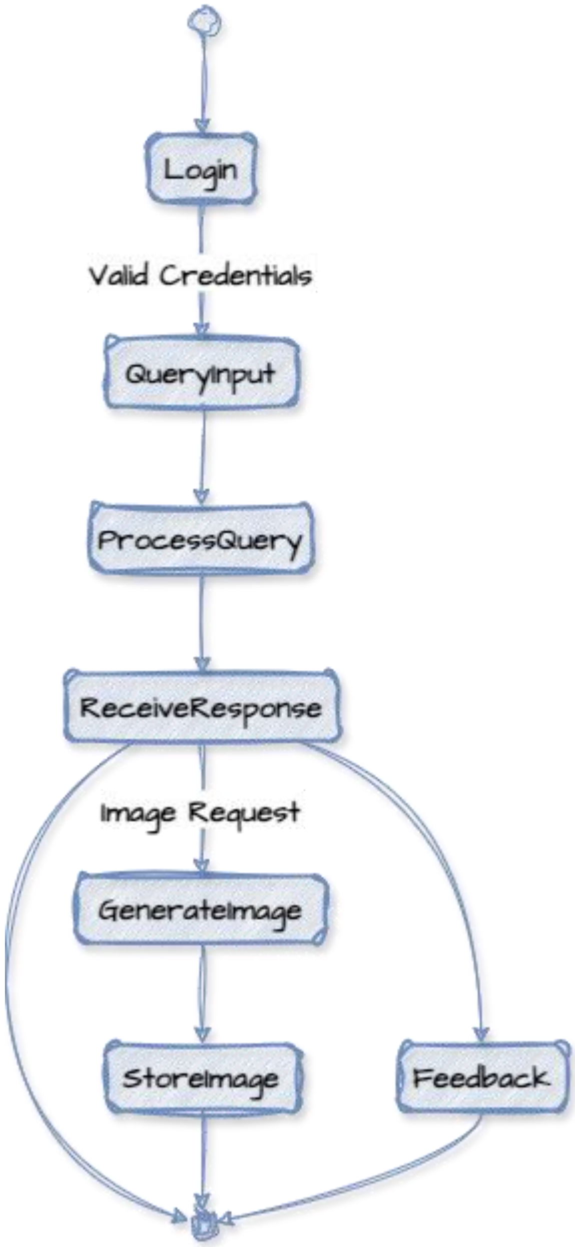

Searching Box , Image Sample
AI Tools
Prompt
Result in Text, Image, Diagram.
Searching Box , Image Sample
User
Prompt

1st Level Admin Side DFD

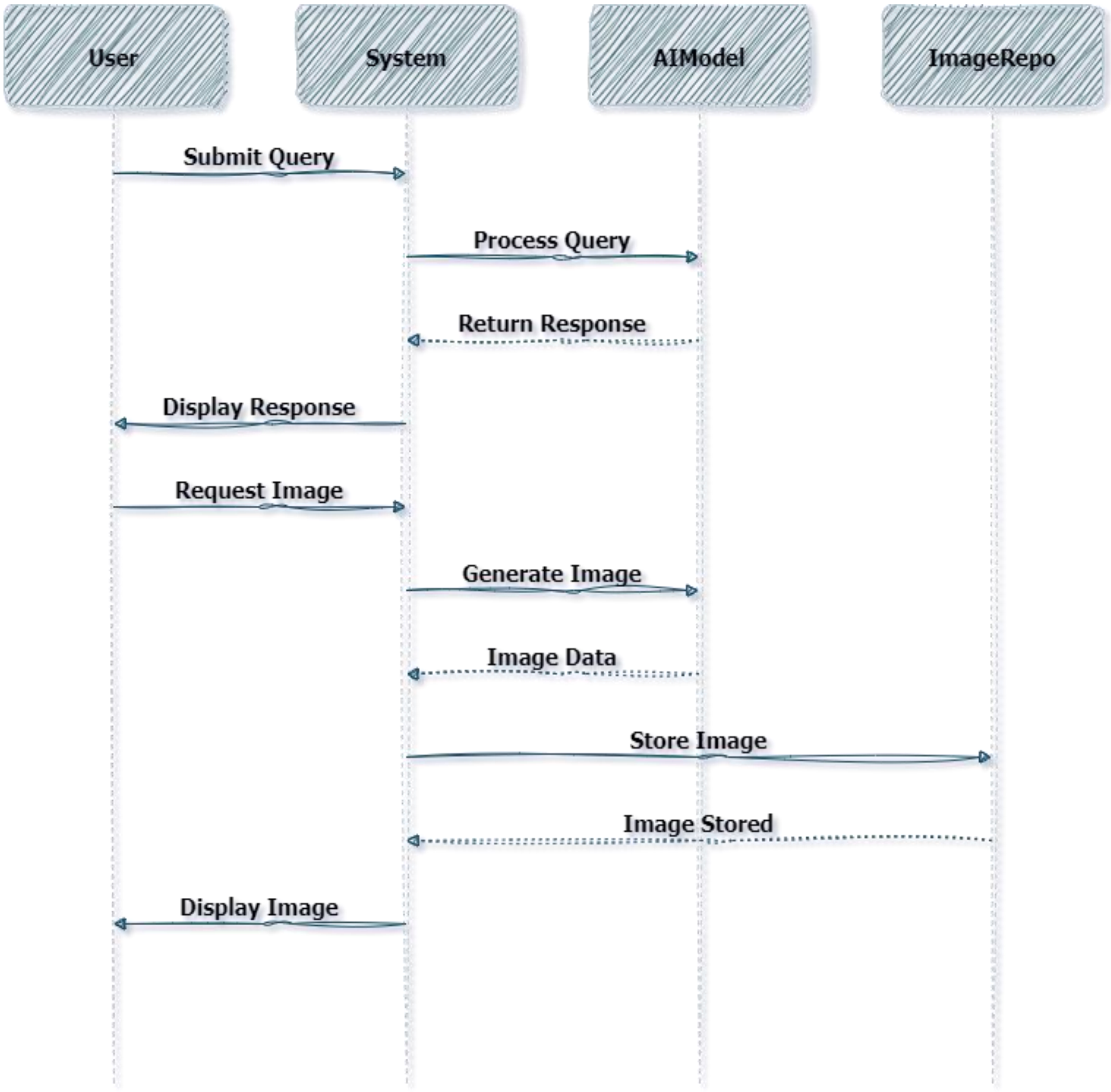
- The Admin side DFD describes the functionality of Admin, Admin is the owner of the website. Admin can first add a category of items and then add items by category wise. and admin can manage order and payment detail.



Activity Diagram :



Sequence Diagram :



5. Software Testing

Introduction

Testing is crucial for identifying and fixing errors in software to ensure its quality and reliability. It plays a significant role in maintaining software performance and is integral to the maintenance phase of the software lifecycle.

Objectives

- **Detection of Errors:** The primary goal of testing is to uncover defects and errors.
- **Effective Test Cases:** A good test case has a high probability of identifying undiscovered.

Levels of Testing:

API Key Management :

- **Secure Storage:** Store API keys securely using environment variables or secure vaults to prevent unauthorized access.
- **Access Control:** Ensure that only authorized components or personnel have access to the API keys.
- **Rotation and Expiry:** Regularly rotate API keys and handle key expiry to maintain security.

Integration Testing with AI Tools :

- **Setup:** Configure your testing environment with the necessary API keys and ensure that they are correctly loaded into your application.
- **Mocking and Stubbing:** Use mocking or stubbing to simulate API responses during testing to avoid hitting the actual API and to control test conditions.
- **Error Handling:** Test how your application handles errors from the AI tools, such as invalid API keys or rate limiting.

Performance Testing

- **API Rate Limits:** Test how your system performs under the constraints of API rate limits and handle any rate limit errors appropriately.
- **Load Testing:** Simulate multiple concurrent requests to the AI APIs to assess how the system handles high volumes of API calls.

Security Testing

- **API Key Exposure:** Ensure that API keys are not exposed in logs or error messages.
- **Authorization:** Verify that API keys are used correctly and that unauthorized requests are appropriately handled.

Stress Testing

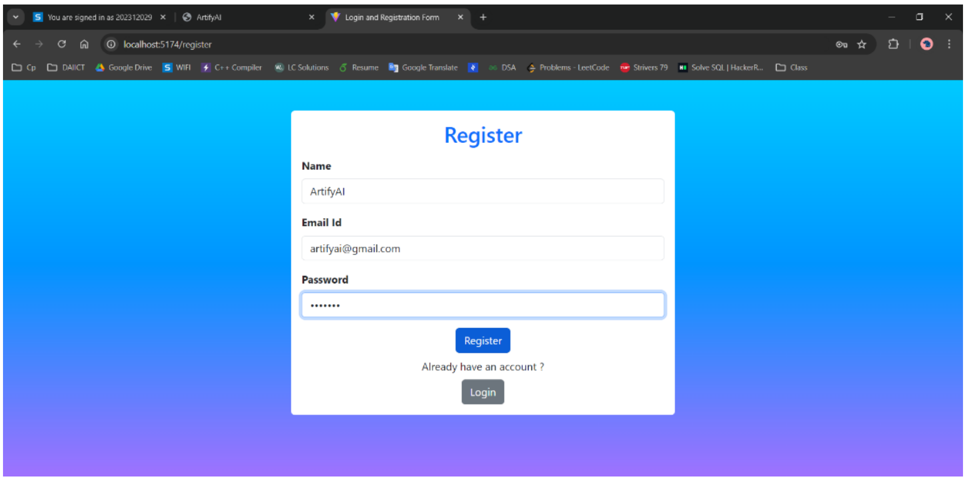
- **High Load:** Stress test your application by simulating high loads and observing how it performs under extreme conditions with the AI APIs.
- **Resilience:** Assess the resilience of your system in handling API failures or slow responses.

Example Integration Testing Procedure

1. **Setup Environment:**
   - Load API keys into environment variables.
2. **Mock API Responses:**
   - Use a library like `nock` or `sinon` to mock API responses during tests.
3. **Test Scenarios:**
   - Test successful integration by sending valid requests and verifying responses.
   - Test error handling by simulating API errors or invalid API keys.
4. **Review Logs:**
   - Ensure no sensitive information (e.g., API keys) is exposed in logs.
5. **Run Performance Tests:**
   - Simulate high request volumes to test API rate limits and system performance.

Snapshot:

Registration Page :



Register

Name

ArtifyAI

Email Id

artifyai@gmail.com

Password

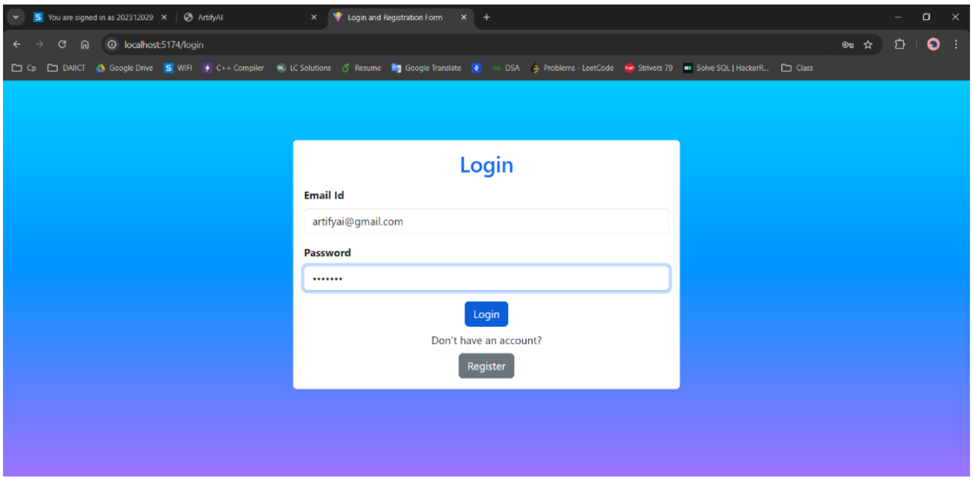
•••••••

Register

Already have an account ?

Login

Login Page :



Login

Email Id

artifyai@gmail.com

Password

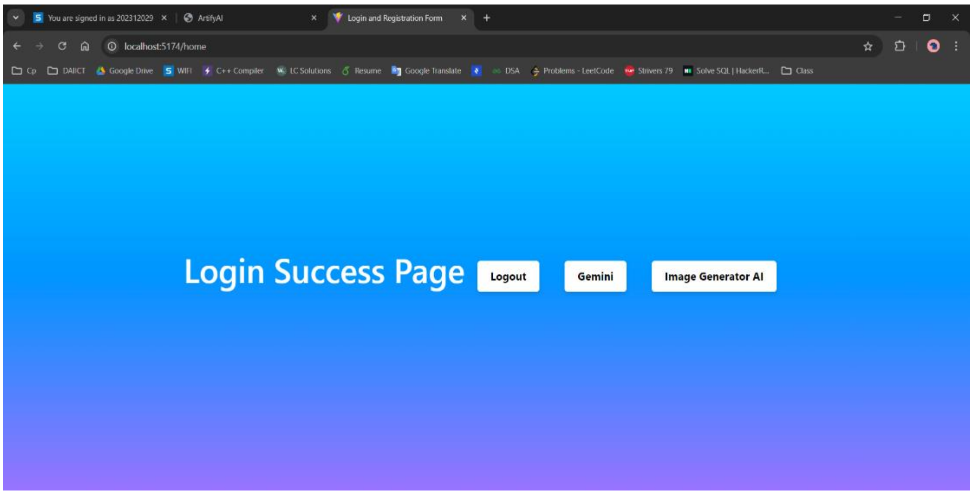
•••••••

Login

Don't have an account?

Register

Home Page :



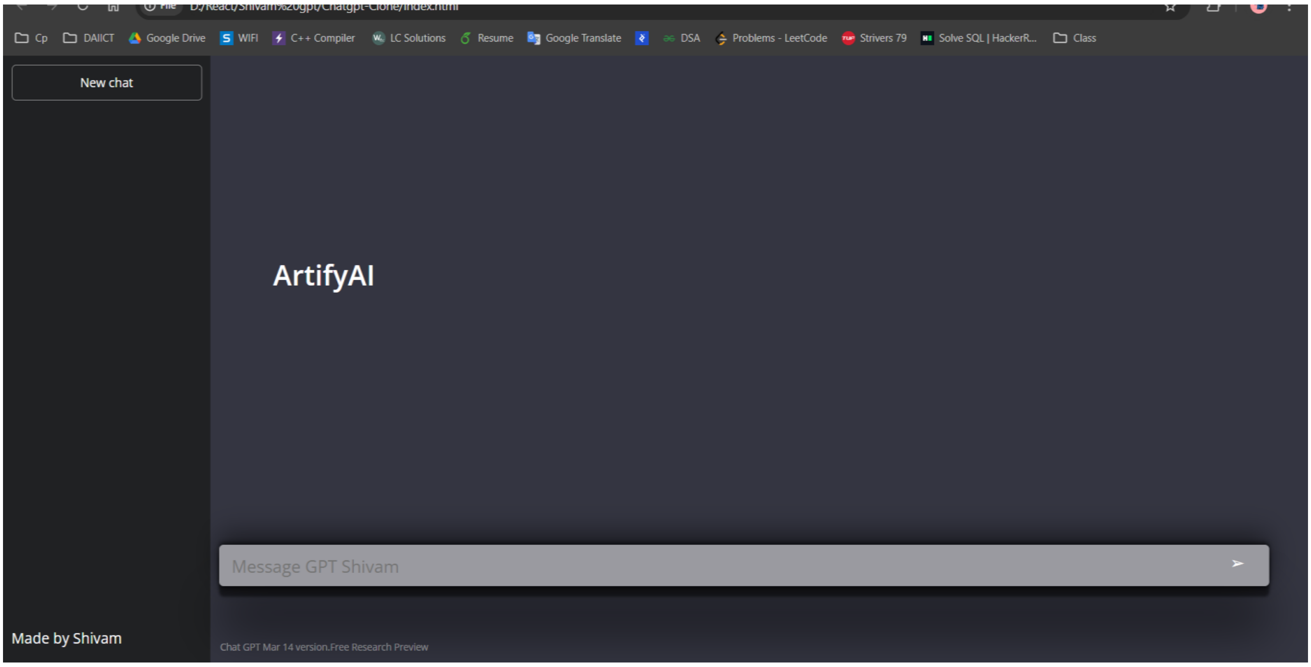
Login Success Page

Logout

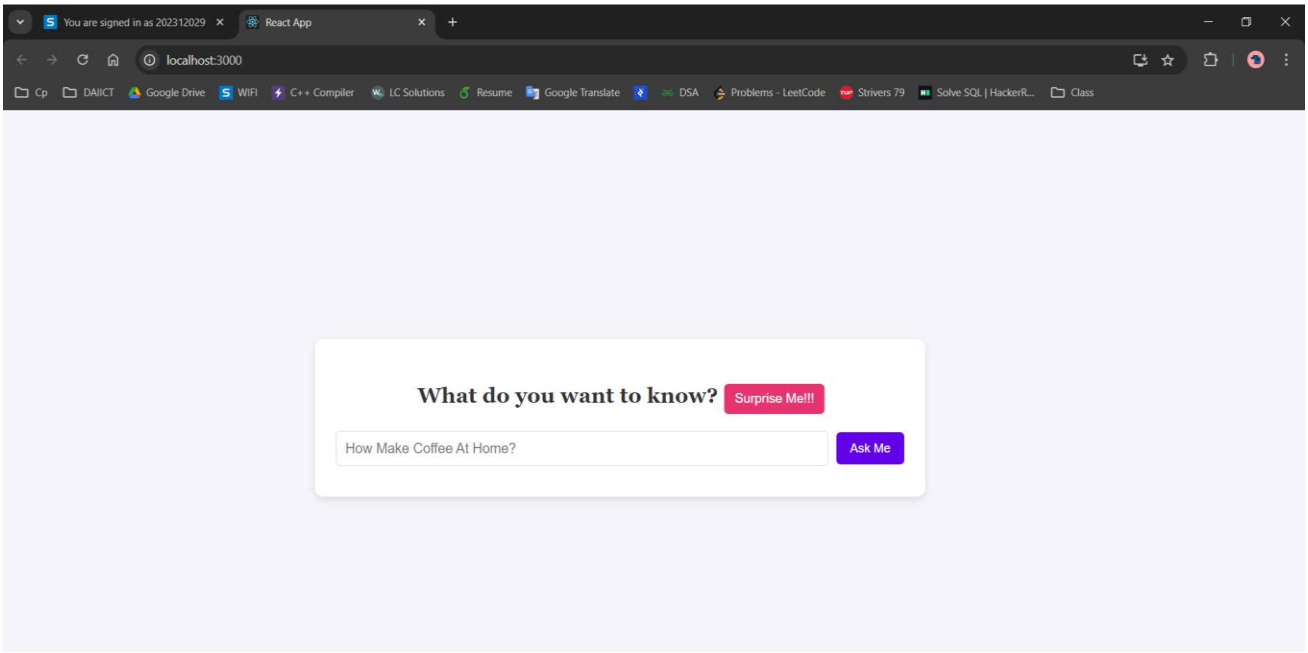
Gemini

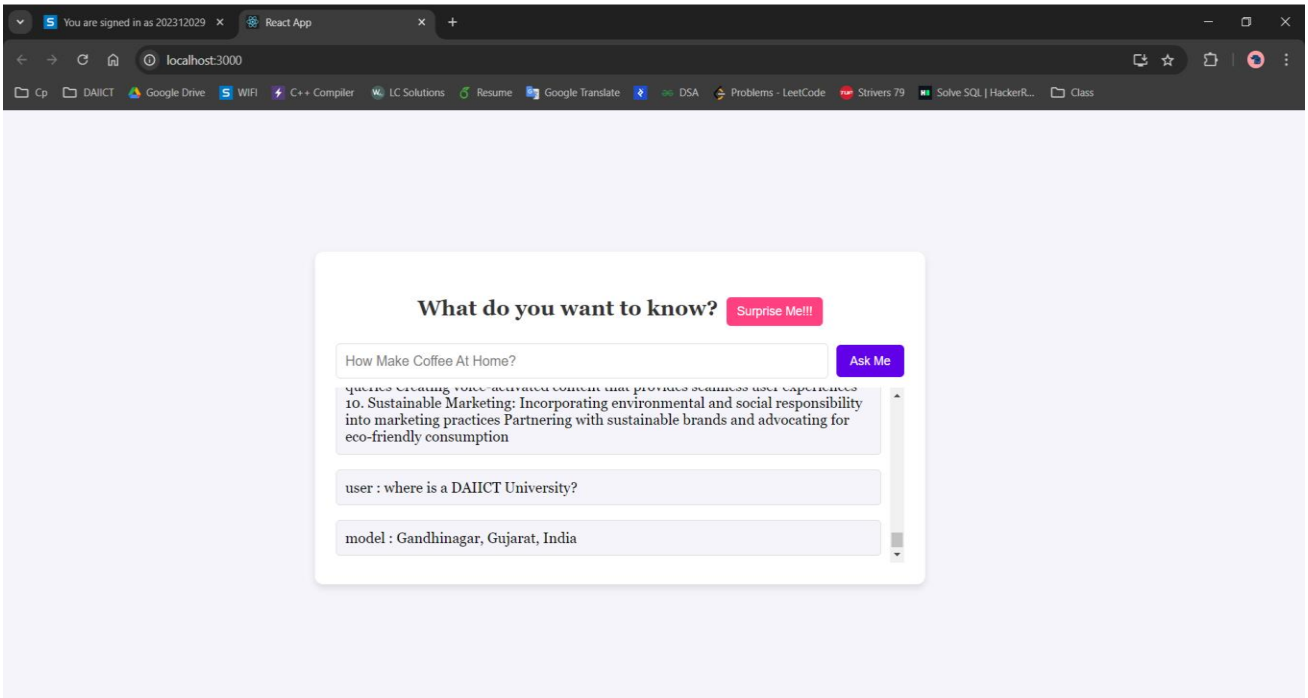
Image Generator AI

Chatgpt page :



Gemini Page :





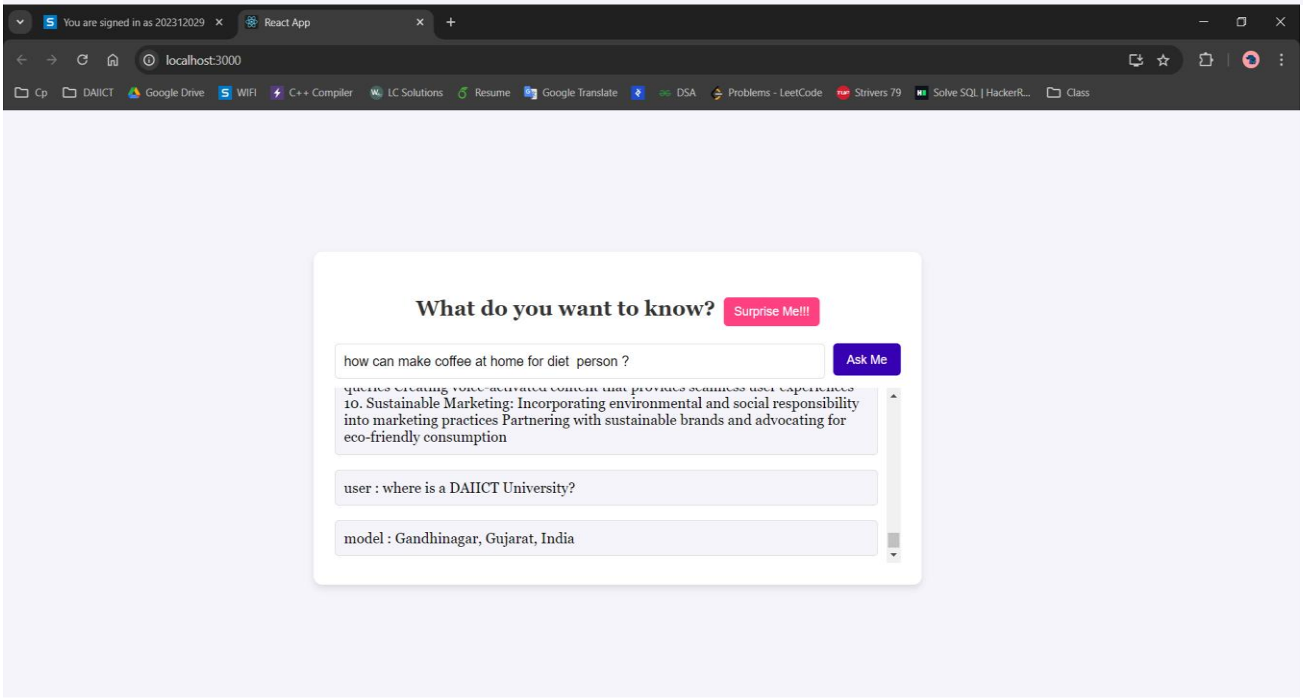
What do you want to know?

How Make Coffee At Home?

queries Creating voice-activated content that provides seamless user experiences
10. Sustainable Marketing: Incorporating environmental and social responsibility into marketing practices Partnering with sustainable brands and advocating for eco-friendly consumption

user : where is a DAIICT University?

model : Gandhinagar, Gujarat, India



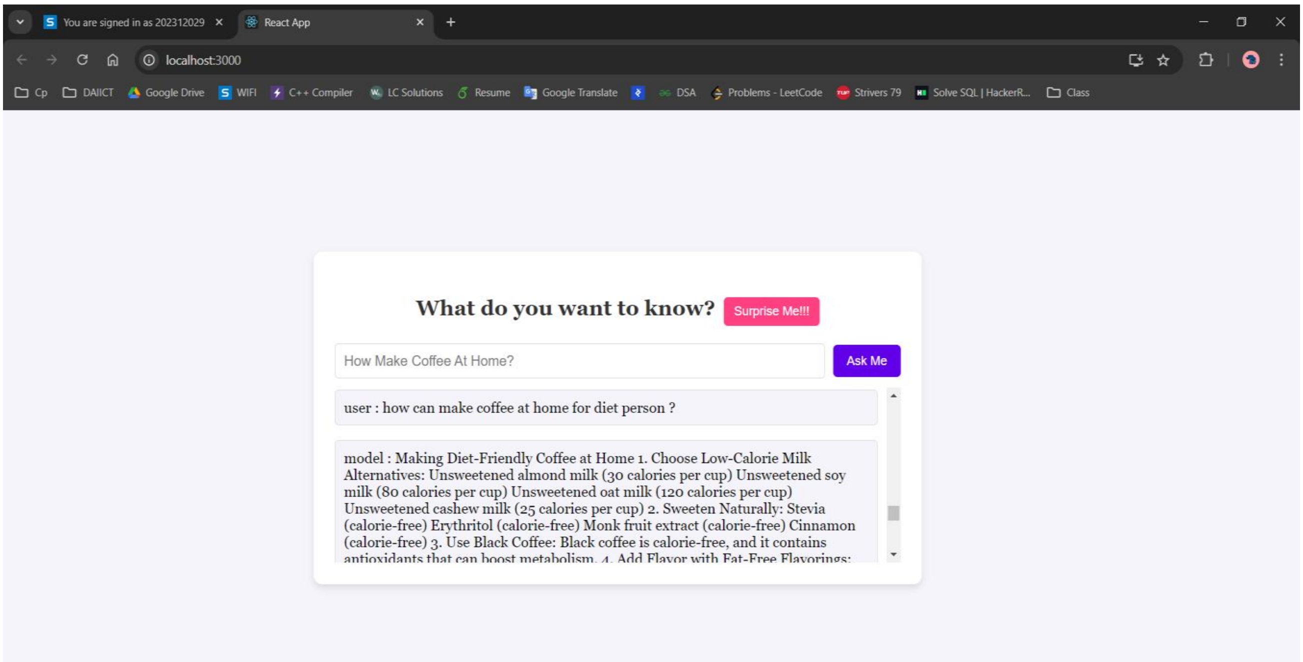
What do you want to know?

how can make coffee at home for diet person ?

queries Creating voice-activated content that provides seamless user experiences
10. Sustainable Marketing: Incorporating environmental and social responsibility into marketing practices Partnering with sustainable brands and advocating for eco-friendly consumption

user : where is a DAIICT University?

model : Gandhinagar, Gujarat, India



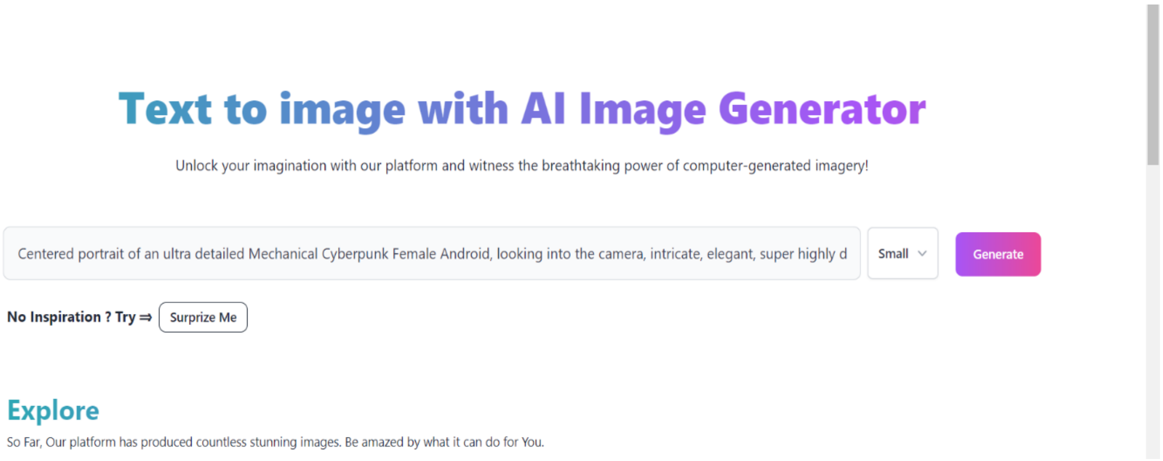
What do you want to know?

How Make Coffee At Home?

user : how can make coffee at home for diet person ?

model : Making Diet-Friendly Coffee at Home 1. Choose Low-Calorie Milk Alternatives: Unsweetened almond milk (30 calories per cup) Unsweetened soy milk (80 calories per cup) Unsweetened oat milk (120 calories per cup) Unsweetened cashew milk (25 calories per cup) 2. Sweeten Naturally: Stevia (calorie-free) Erythritol (calorie-free) Monk fruit extract (calorie-free) Cinnamon (calorie-free) 3. Use Black Coffee: Black coffee is calorie-free, and it contains antioxidants that can boost metabolism. 4. Add Flavor with Fat-Free Flavorings:

Images Generate page :



Text to image with AI Image Generator

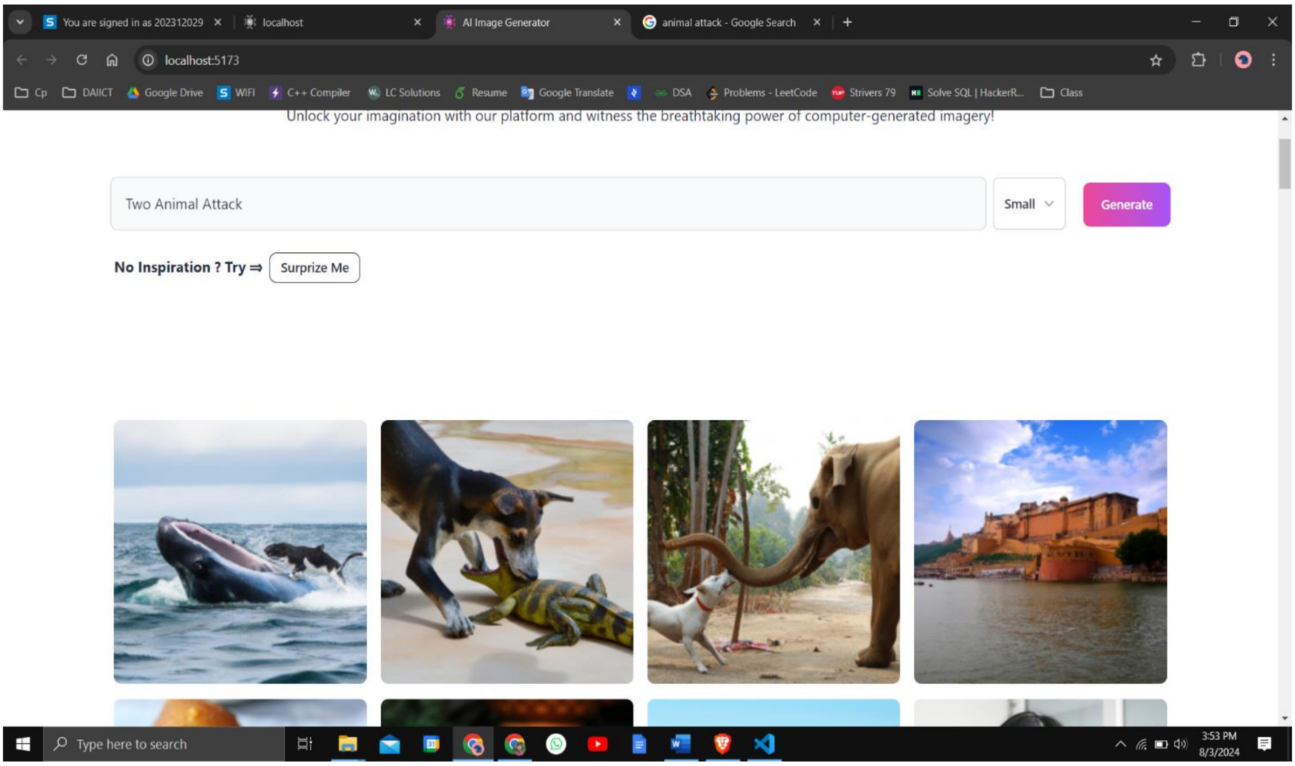
Unlock your imagination with our platform and witness the breathtaking power of computer-generated imagery!

Centered portrait of an ultra detailed Mechanical Cyberpunk Female Android, looking into the camera, intricate, elegant, super highly d

No Inspiration ? Try ⇒

Explore

So Far, Our platform has produced countless stunning images. Be amazed by what it can do for You.

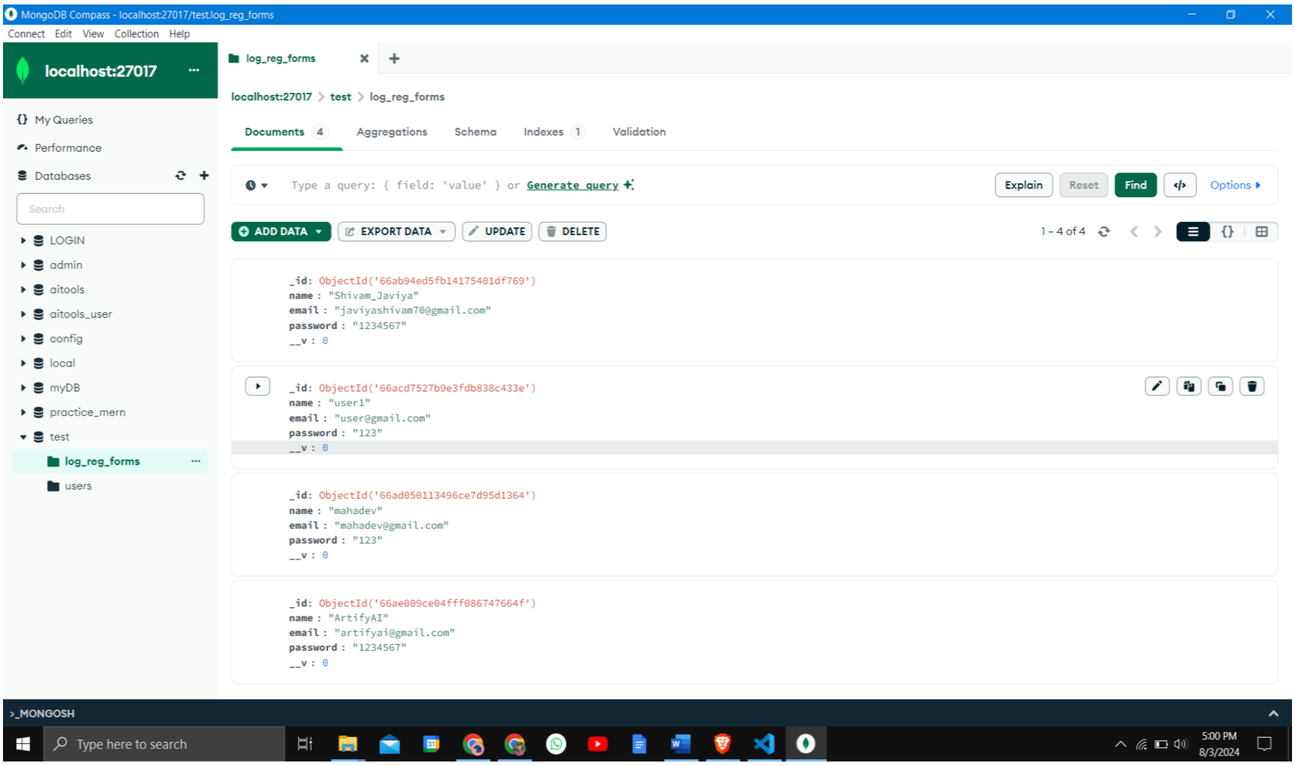


Unlock your imagination with our platform and witness the breathtaking power of computer-generated imagery!

Two Animal Attack

Small

Mango DB database :



```
_id: ObjectId('66ab94ed5fb14175401df769')
name : "Shivam_Javiya"
email : "javiyashivam70@gmail.com"
password : "1234567"
__v : 0

_id: ObjectId('66acd7527b9e3fdb838c433e')
name : "user1"
email : "user@gmail.com"
password : "123"
__v : 0

_id: ObjectId('66ad050113496ce7d95d1364')
name : "mahadev"
email : "mahadev@gmail.com"
password : "123"
__v : 0

_id: ObjectId('66ae009ce04fff086747664f')
name : "ArtifyAI"
email : "artifyai@gmail.com"
password : "1234567"
__v : 0
```

Summary and Lessons Learned :

Summary

The project report on **ArtifyAI**, an AI-driven platform, outlines the development and deployment of a modern web application using the MERN (MongoDB, Express.js, React, Node.js) stack. The platform leverages cutting-edge AI models such as ChatGPT and Google Gemini to provide users with powerful tools for query processing and image generation. Key features include user authentication, multi-model response consolidation, and personalized user experiences.

Lessons Learned

- **User Experience (UX) is Key:** Prioritizing intuitive design and easy navigation significantly enhances user satisfaction. A user-friendly interface is crucial for retaining users and encouraging engagement.
- **Strong Security Measures:** Implement robust encryption and secure authentication mechanisms to protect user data and ensure secure API interactions. Regularly update security protocols to address emerging threats.
- **Scalability and Performance:** Design the system to efficiently handle growing user demands and traffic spikes. Optimize code, use caching, and consider server-side rendering to improve loading times and responsiveness.
- **Real-Time Updates:** Integrate real-time features such as order tracking and notifications to enhance user engagement and provide timely information.
- **Continuous Integration and Deployment:** Establish a robust CI/CD pipeline to facilitate efficient updates, bug fixes, and feature rollouts, ensuring a stable and continuously improving application.
- **User Feedback and Iterative Improvement:** Actively seek user feedback and incorporate it into development cycles for ongoing enhancements. User insights are invaluable for identifying areas of improvement and ensuring the platform meets user needs.

Reference: OpenAi, google gemini,Youtube ,Github.