DANIEL NASHED

- GitHub: https://github.com/danielnashed/ReSearchGraph_1
- Website: https://project2.d269jq01wtg7ba.amplifyapp.com/

AI RESEARCH TRENDS DASHBOARD

Dashboard to Visualize Latest AI Trends

TABLE OF CONTENTS

PROBLEM STATEMENT

AI Research Field is Exploding

- ➔ More than 200 papers are released every single day in the field of GenAI
- ➔ Difficult to keep up with the latest trends in the field

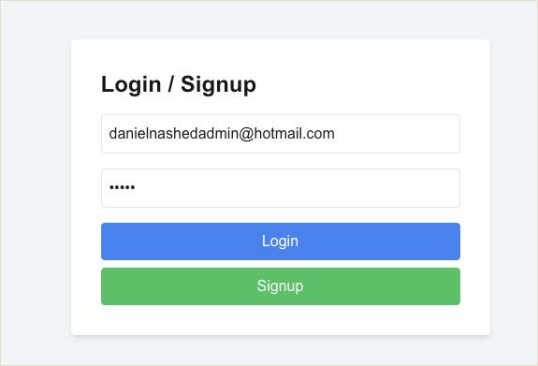
GOALS

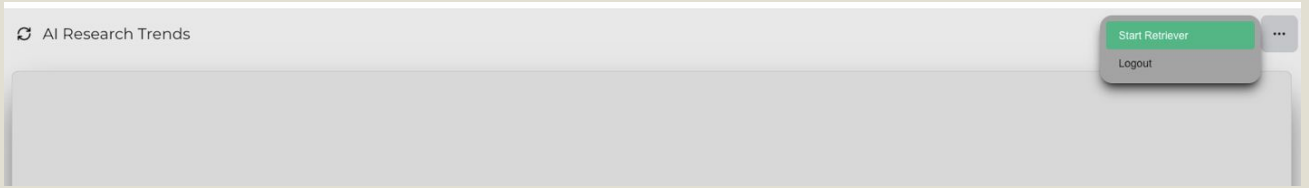
➔ Automate the retrieval and clustering of academic papers in Generative AI

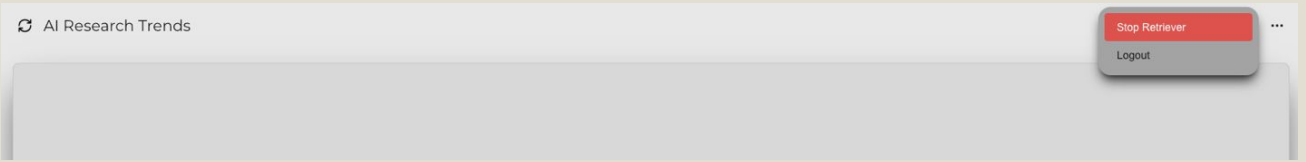
- Fetch & process daily stream of academic papers from arXiv.org
- Group papers into existing clusters or new clusters automatically
- Users can interactively explore research trends through a frontend interface.

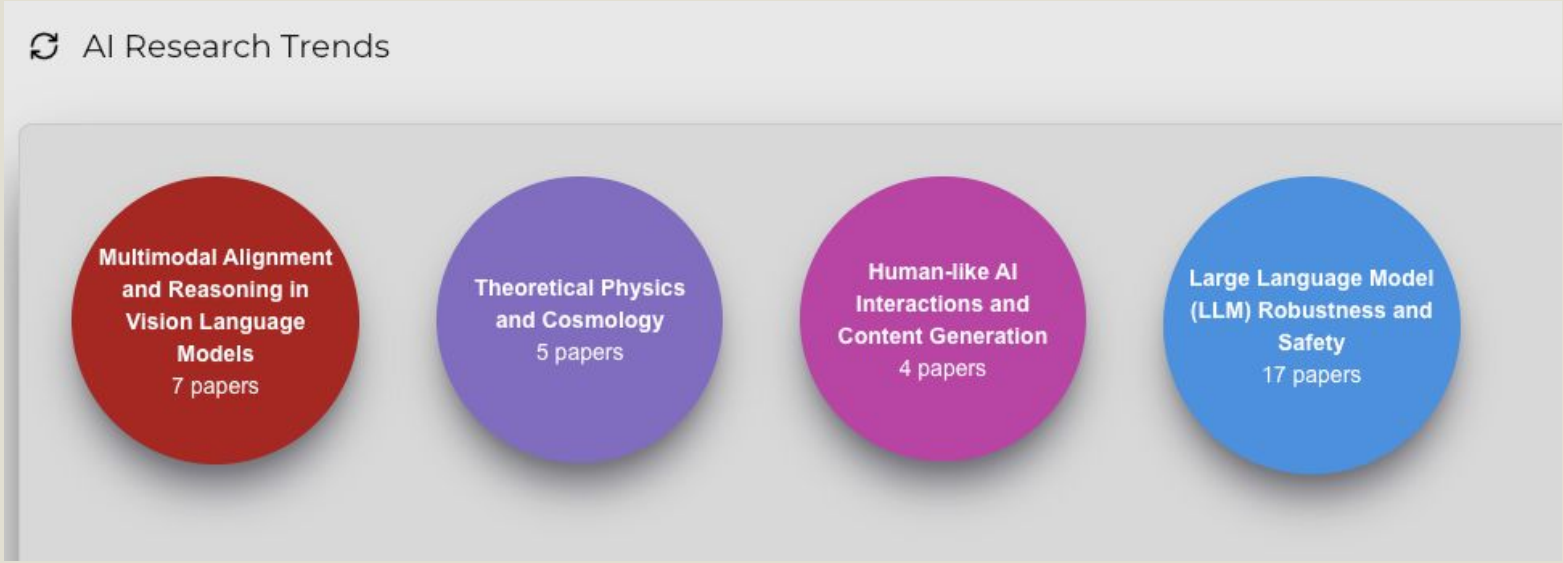
DEMO

Website: https://project2.d269jq01wtg7ba.amplifyapp.com/

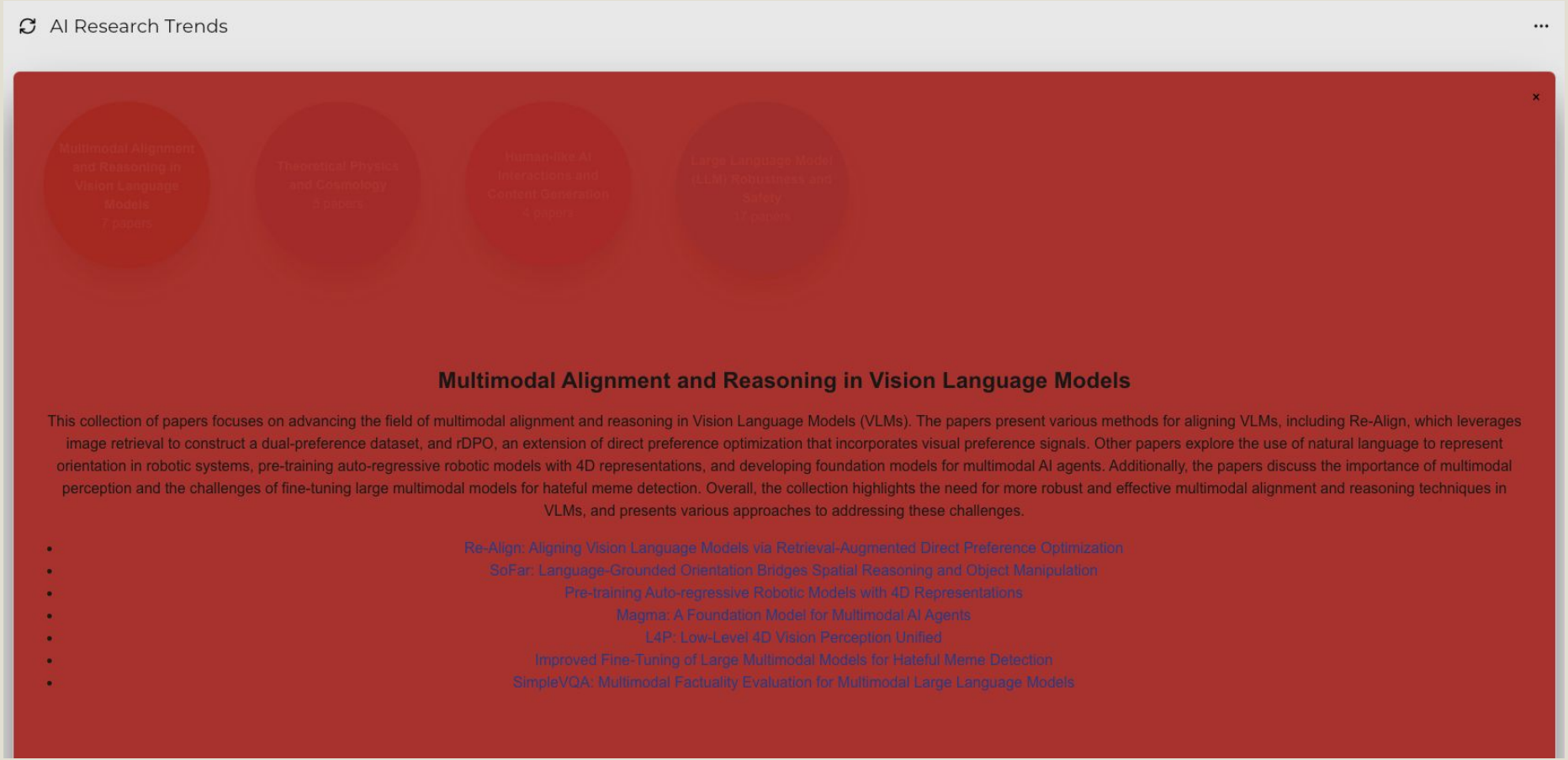




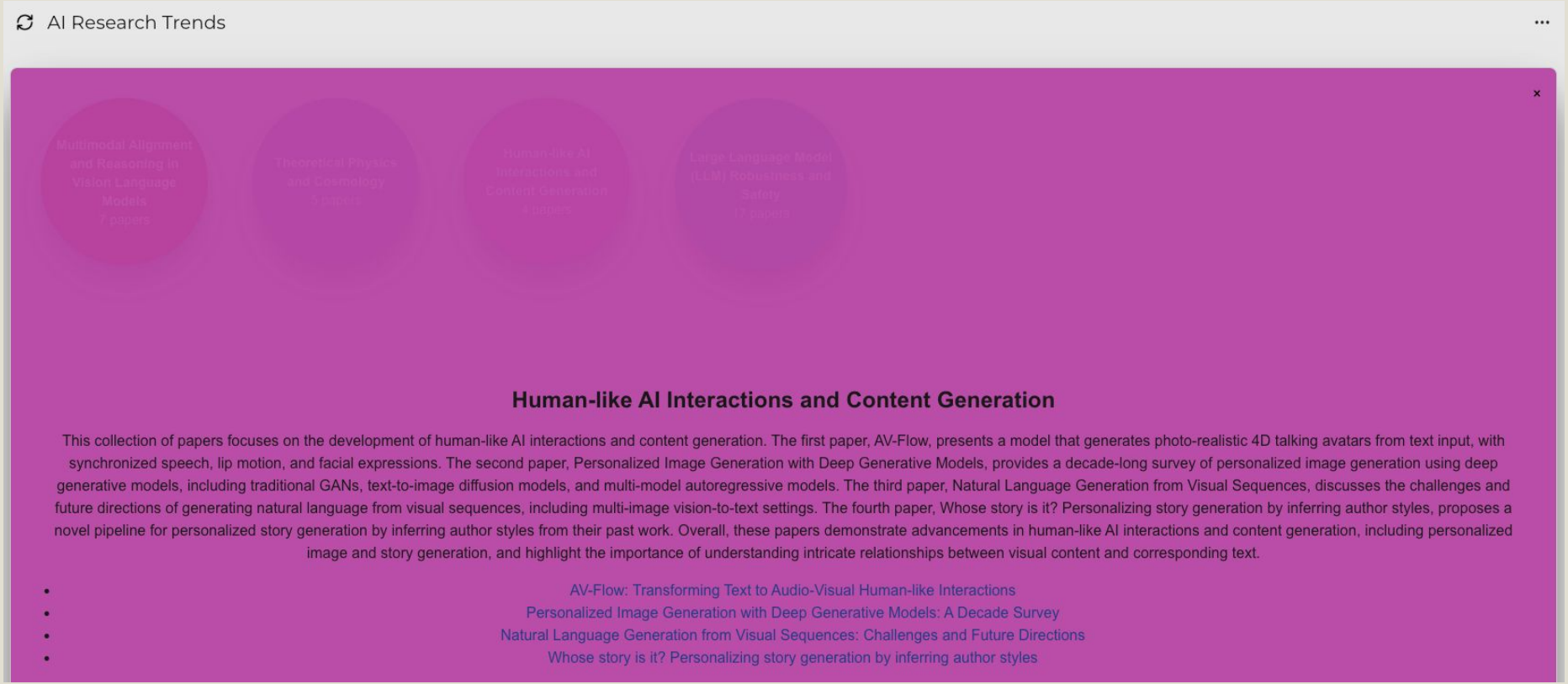




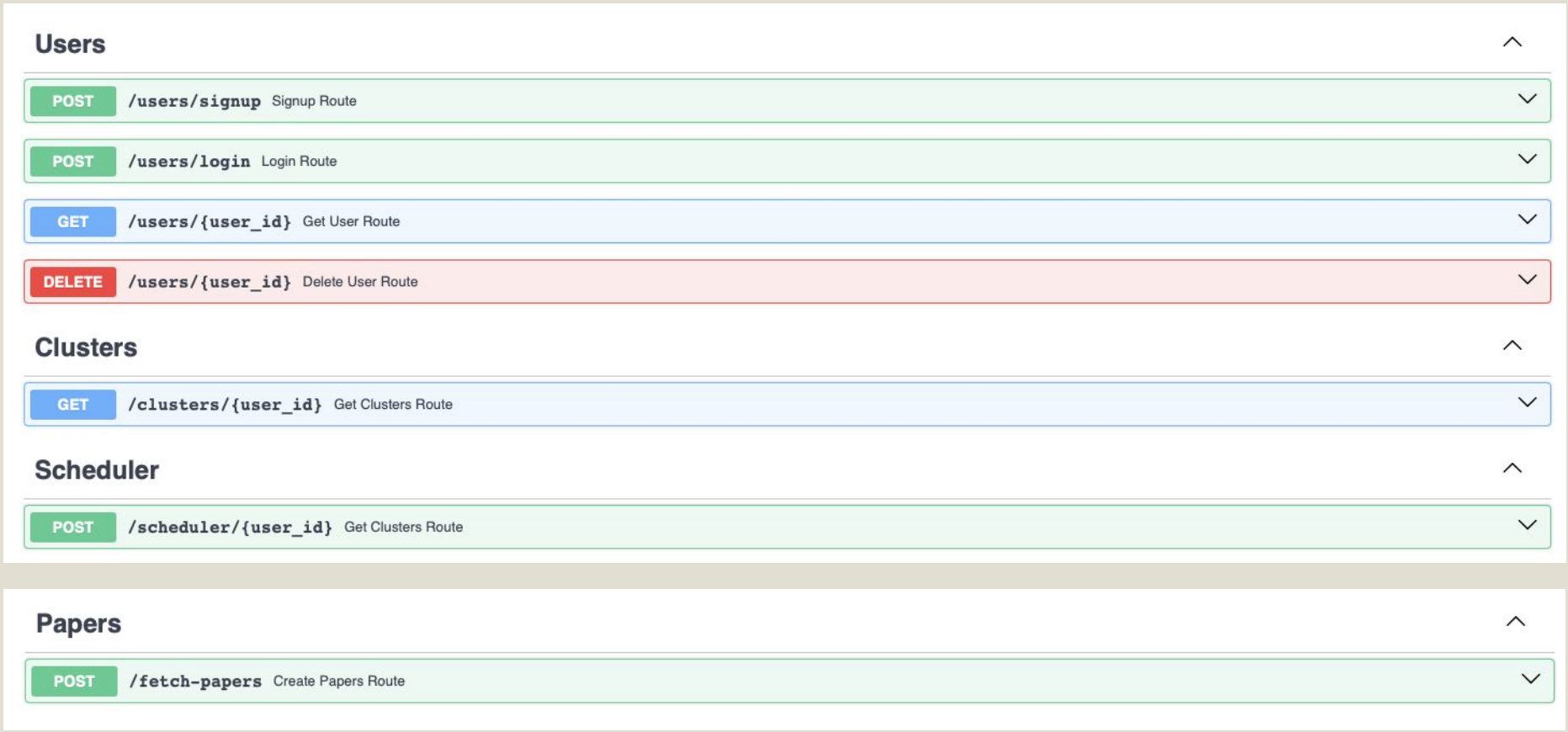
Website: https://project2.d269jq01wtg7ba.amplifyapp.com/



Website: https://project2.d269jq01wtg7ba.amplifyapp.com/

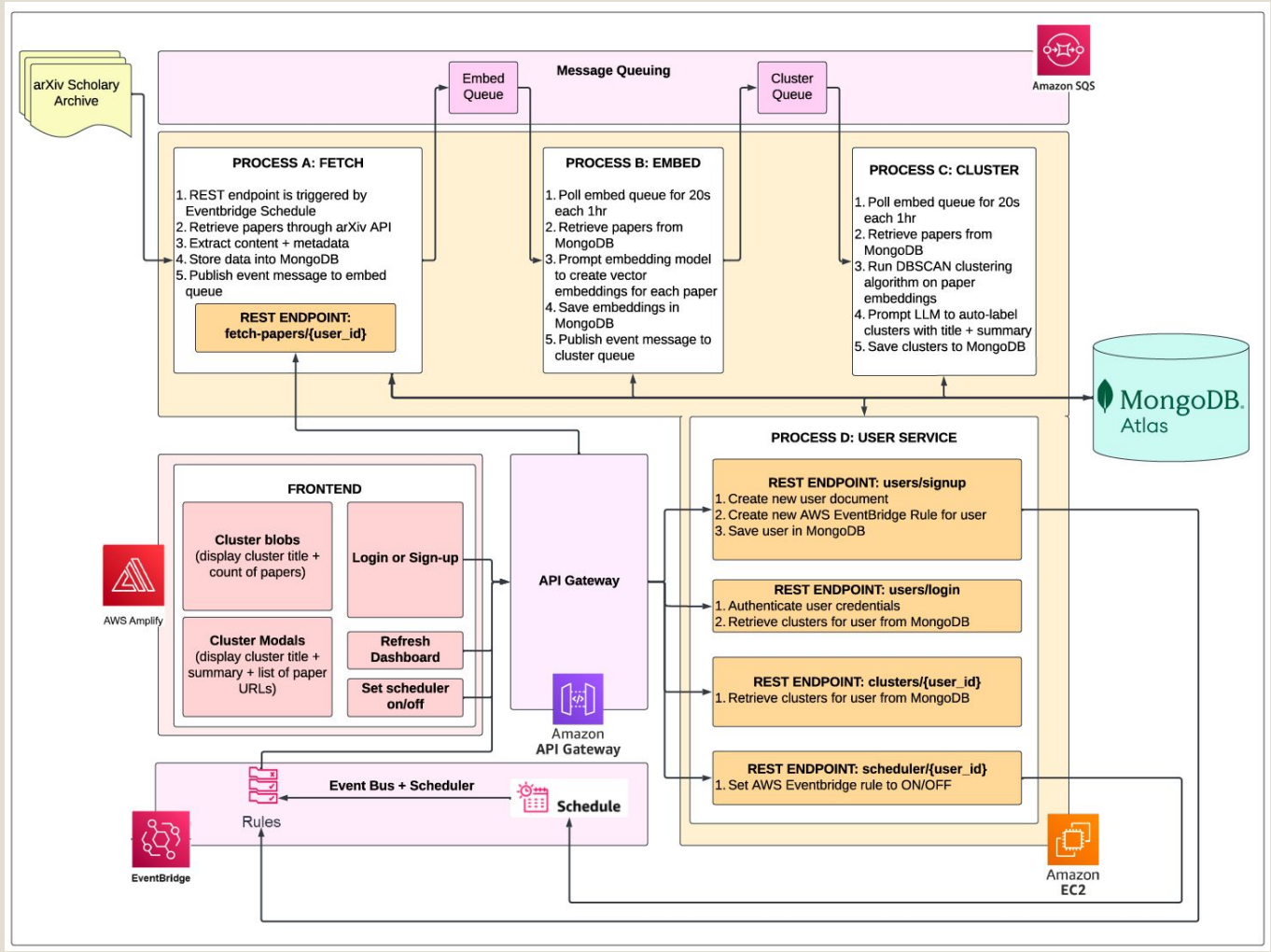


DEMO


Users
POST /users/signup Signup Route
POST /users/login Login Route
GET /users/{user_id} Get User Route
DELETE /users/{user_id} Delete User Route
Clusters
GET /clusters/{user_id} Get Clusters Route
Scheduler
POST /scheduler/{user_id} Get Clusters Route
Papers
POST /fetch-papers Create Papers Route

SYSTEM OVERVIEW

SYSTEM DIAGRAM



TECH STACK

➔ Backend

- FastAPI for web framework (python web framework)
- MongoDB for NoSQL database to store paper embeddings and metadata
- AWS CloudFormation as infra-as-code to provision cloud resources
- AWS API Gateway to manage API requests efficiently
- AWS CloudWatch for monitoring and logging system
- AWS EventBridge to schedule daily retrieval tasks
- AWS SQS to manage processing queues to handle high-load data streams
- AWS ECS for container orchestration
- Docker to build container images & AWS ECR to store images

➔ Frontend

- Next.js: React framework optimized for SEO and fast rendering
- TailwindCSS: Lightweight CSS framework for styling.
- daisyUI: Prebuilt UI components for a modern user experience
- AWS Amplify for hosting frontend

➔ AI/ML

- AWS Bedrock: Generates vector embeddings for research papers
- Incremental-DBSCAN: Dynamic clustering algorithm for research papers

KEY COMPONENTS

- Fetching papers
  - Retrieve research papers from the arXiv API
  - Extract text and metadata & save in MongoDB
- Embeddings
  - Convert title + abstract of each paper into vector embeddings using Cohere Embed model
  - Store embeddings in Paper Document inside MongoDB
- Clustering
  - Apply dynamic + incremental clustering algorithm INCREMENTAL-DBSCAN to group embeddings based on topic similarity.
    - Incremental aspect enables batch processing of papers. New batches either clustered into existing clusters or inserted into new clusters based on similarity threshold.
  - Use Llama 3.3 Instruct (70B) to auto-label clusters + generate summary for each cluster
- User Management
  - User sign-up and login/authentication
  - Creation of unique EventBridge rule + schedule for each user
  - Toggling scheduler on/off for each user

FUTURE IMPROVEMENTS

FUTURE IMPROVEMENT

- ➔ **Expand Data Sources**: fetch papers from Semantic Scholar and Google Scholar
- ➔ **Hybrid Clustering**: allow users to control initial clusters and labels for more fine-grained control over clustering
- ➔ **Context Management**: implement memory model to track important research trends over time
- ➔ **Graph Database:** build a graph using Neo4j to capture semantic relationships between papers (authors names, institution names, journal names, publication year, # of citations, # of references)