CodeSnap - Group 7

EVENTIQUE

An AI-Enhanced Event Planning Web Application

Team Members

Renugopal Sivaprakasam (11699653), Sparsha Dandganhalli Prakash (11684265), Soniya Kodipunjula (11708235), Govind Asawa (11641270), Usha Chandrashekar (11722182), Ruthvick Ravula (11653170)

Table of Contents

Cover Page

Product Vision

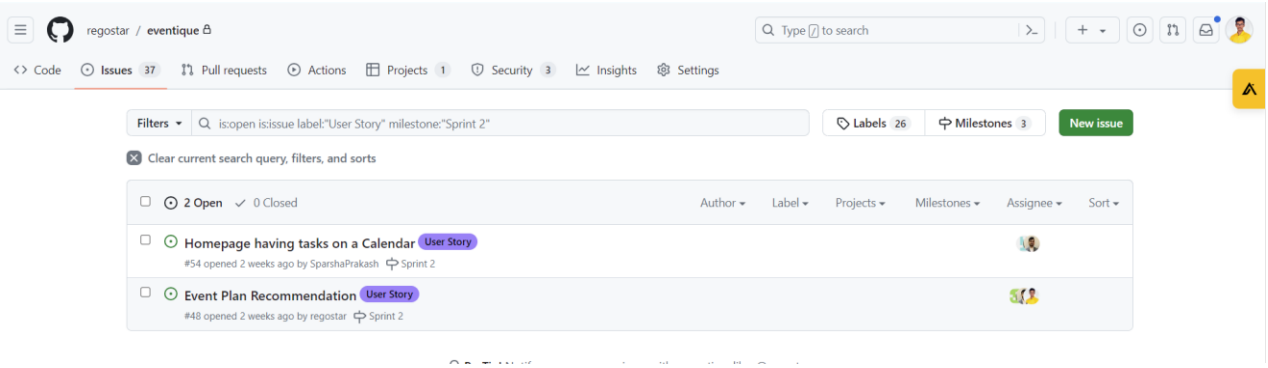
For anyone **who** is planning an event like a birthday party, wedding, or conference, **the** application Eventique is a web app **that** works on a browser on any device. It uses artificial intelligence to supercharge your event planning in minutes with a recommender, scheduler, organizer, and notification system. **Unlike** other event management companies (such as Cvent, EventPro, and Bizzabo) which don't have recommendation systems, are pricey and take a lot of time, **our** product provides a platform to design your event quickly in a way you would like within a few minutes.

Github Repository

https://github.com/regostar/eventique

Active User Stories and Tasks

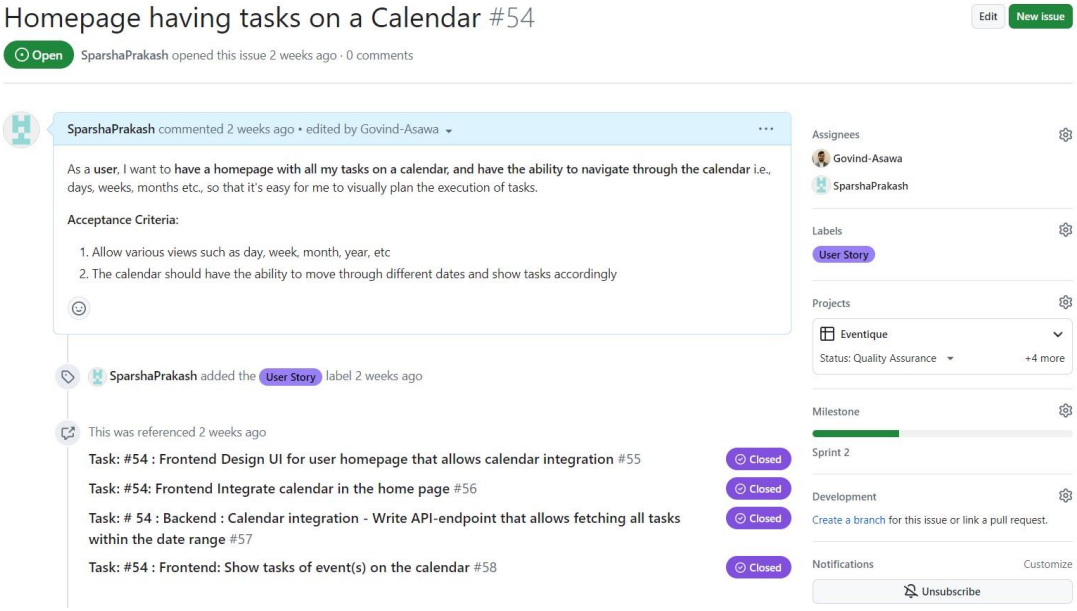
USER Stories:



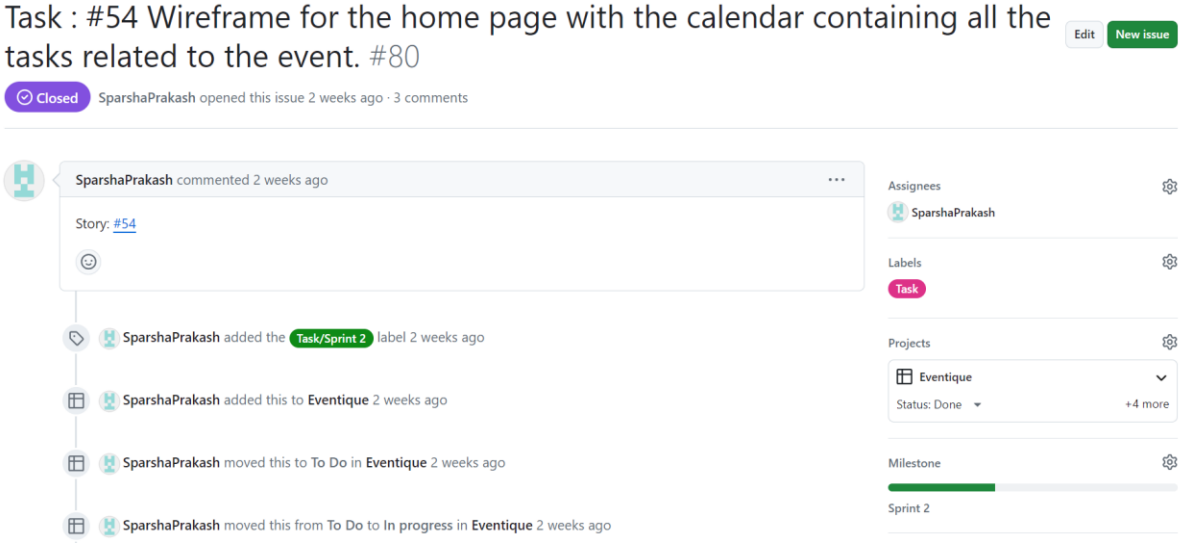
All Tasks, User Stories, and Spikes high level view:

	Title	Assignees	Status
1	Task : #54 Wireframe for the home page with the calendar containing all the tasks related t... #80	SparshaPrakash	Done
2	Homepage having tasks on a Calendar #54	Govind-Asawa and ...	Quality Assurance
3	Task : Dockerize the application #88	regostar	Done
4	Task : Host the application on GCP Cloudrun #98	regostar	Done
5	Task : Logging Setup #95	SparshaPrakash	Done
6	Task : # 48 Backend : API which responds tasks and it's description in a certain specified for... #52	regostar	Done
7	Spike: Event Data Database Schema Design #75	Govind-Asawa, reg...	Done
8	Task : Setup CI Pipeline #89	regostar	Done
9	Task : # 48 Backend : Develop an API end-point at the server side that takes user prompt a... #51	regostar and ruthvi...	Done
10	Task: #54: Frontend Integrate calendar in the home page #56	Govind-Asawa	Done
11	Bug: #46: signup page #77	Govind-Asawa	Done
12	Task : #48 Wireframe for generating tasks given an event description by a user #49	Usha-Chandrashekar	To Do
13	Event Plan Recommendation #48	Govind-Asawa, reg...	Quality Assurance
14	Task: #54 : Frontend: Show tasks of event(s) on the calendar #58	Govind-Asawa	Done
15	Task: # 54 : Backend : Calendar integration - Write API-endpoint that allows fetching all tas... #57	Govind-Asawa	Done
16	Task: #54 : Frontend Design UI for user homepage that allows calendar integration #55	SparshaPrakash	Done
17	Task : # 48 Frontend development for chatbot screen using the provided design #50	Govind-Asawa and ...	In review
18	Task : Logs Visualization #97	regostar	Done
19	Impl req/resp cycle to show tasks on calendar. Added moment library #101		Done

User Story 1:-



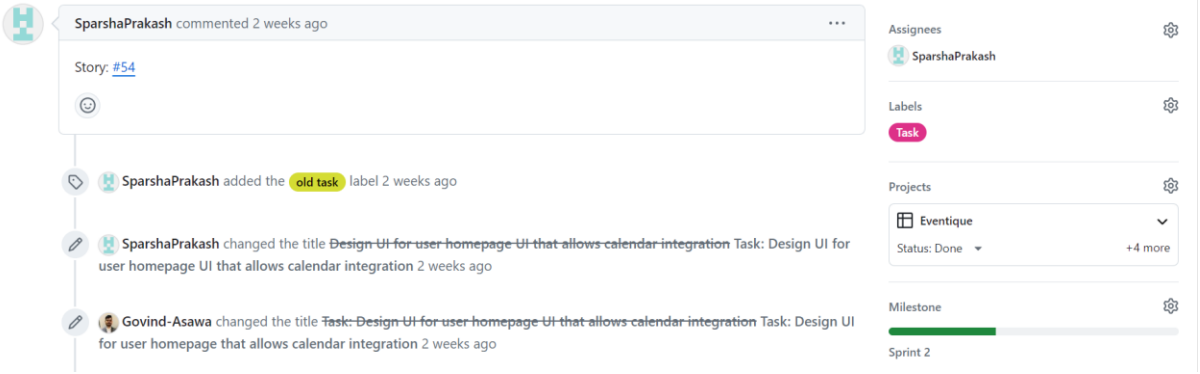
Tasks related to this user story -



Task: #54 : Frontend Design UI for user homepage that allows calendar integration #55

Edit New issue

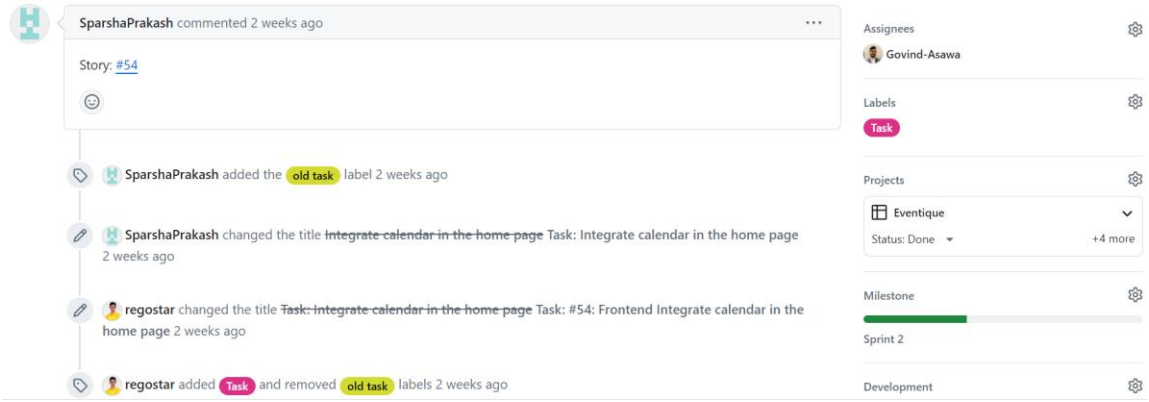
Closed SparshaPrakash opened this issue 2 weeks ago · 1 comment

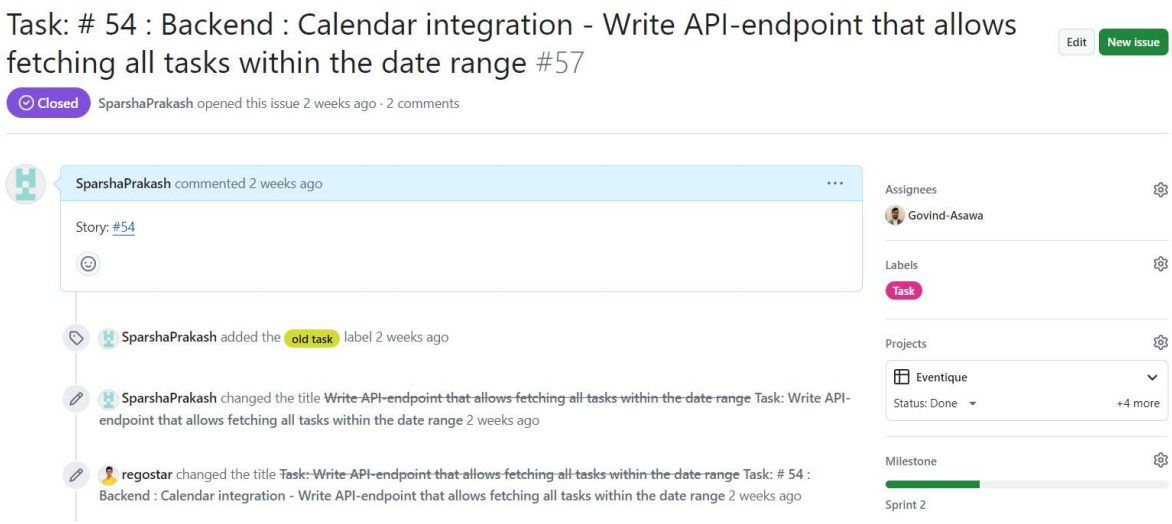


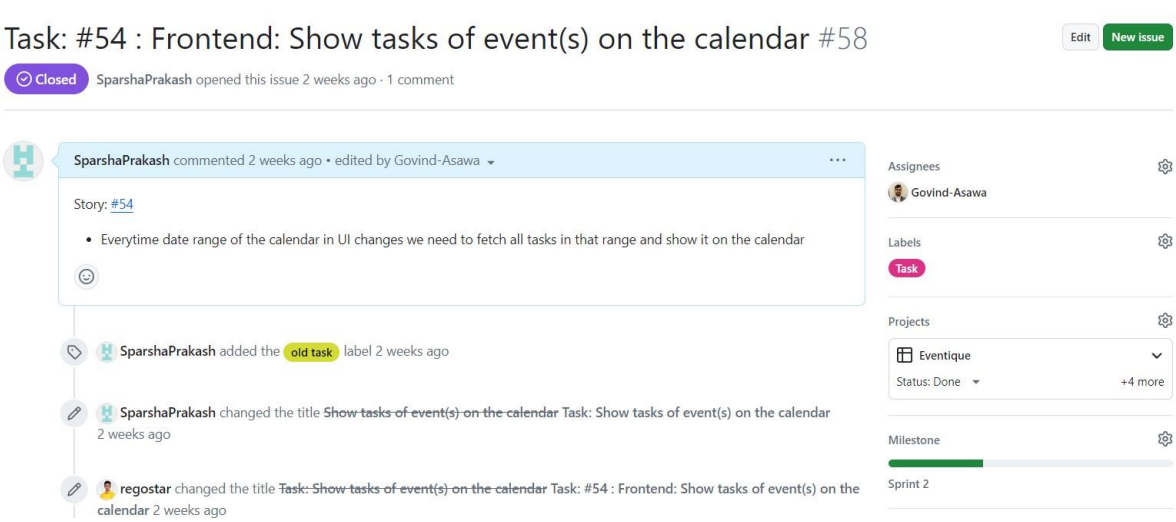
Task: #54: Frontend Integrate calendar in the home page #56

Edit New issue

Closed SparshaPrakash opened this issue 2 weeks ago · 1 comment




Task: # 54 : Backend : Calendar integration - Write API-endpoint that allows fetching all tasks within the date range #57
Edit
New issue
Closed
SparshaPrakash opened this issue 2 weeks ago · 2 comments
SparshaPrakash commented 2 weeks ago
Story: #54
SparshaPrakash added the old task label 2 weeks ago
SparshaPrakash changed the title Write API-endpoint that allows fetching all tasks within the date range Task: Write API-endpoint that allows fetching all tasks within the date range 2 weeks ago
regostar changed the title Task: Write API-endpoint that allows fetching all tasks within the date range Task: # 54 : Backend : Calendar integration - Write API-endpoint that allows fetching all tasks within the date range 2 weeks ago
Assignees
Govind-Asawa
Labels
Task
Projects
Eventique
Status: Done
+4 more
Milestone
Sprint 2


Task: #54 : Frontend: Show tasks of event(s) on the calendar #58
Edit
New issue
Closed
SparshaPrakash opened this issue 2 weeks ago · 1 comment
SparshaPrakash commented 2 weeks ago • edited by Govind-Asawa
Story: #54
Everytime date range of the calendar in UI changes we need to fetch all tasks in that range and show it on the calendar
SparshaPrakash added the old task label 2 weeks ago
SparshaPrakash changed the title Show tasks of event(s) on the calendar Task: Show tasks of event(s) on the calendar 2 weeks ago
regostar changed the title Task: Show tasks of event(s) on the calendar Task: #54 : Frontend: Show tasks of event(s) on the calendar 2 weeks ago
Assignees
Govind-Asawa
Labels
Task
Projects
Eventique
Status: Done
+4 more
Milestone
Sprint 2

User Story 2: -

Event Plan Recommendation #48

Edit | New issue

Open **regostar** opened this issue 2 weeks ago · 0 comments

regostar commented 2 weeks ago

As a **user**, I want to get a **list of tasks to be planned or executed by describing an event** so that I spend the least amount of effort getting confused about the tasks and rather focus on getting the event done.

Acceptance Criteria:

1. The system must allow users to input a description of an event and receive a list of tasks related to planning or executing that event with minimal effort.
2. Each task for the event must have a specific time and date.

regostar assigned **regostar** and **Govind-Asawa** 2 weeks ago

regostar added the User Story label 2 weeks ago

Assignees: regostar, Govind-Asawa, ruthvick-rr

Labels: User Story

Projects: Eventique — Status: Quality Assurance +4 more

Milestone: Sprint 2

Tasks related to this User story:-

Task : # 48 Backend : API which responds tasks and it's description in a certain specified format #52

Edit | New issue

Open **regostar** opened this issue 2 weeks ago · 1 comment

regostar commented 2 weeks ago • edited by Govind-Asawa

Story: #48

Write a prompt that gets a list of tasks from an LLM in the **format needed** based on the event description given by the user.
The format includes -

- Date and time of task
- Task with a description

regostar self-assigned this 2 weeks ago

regostar added the old task label 2 weeks ago

Assignees: regostar

Labels: Task

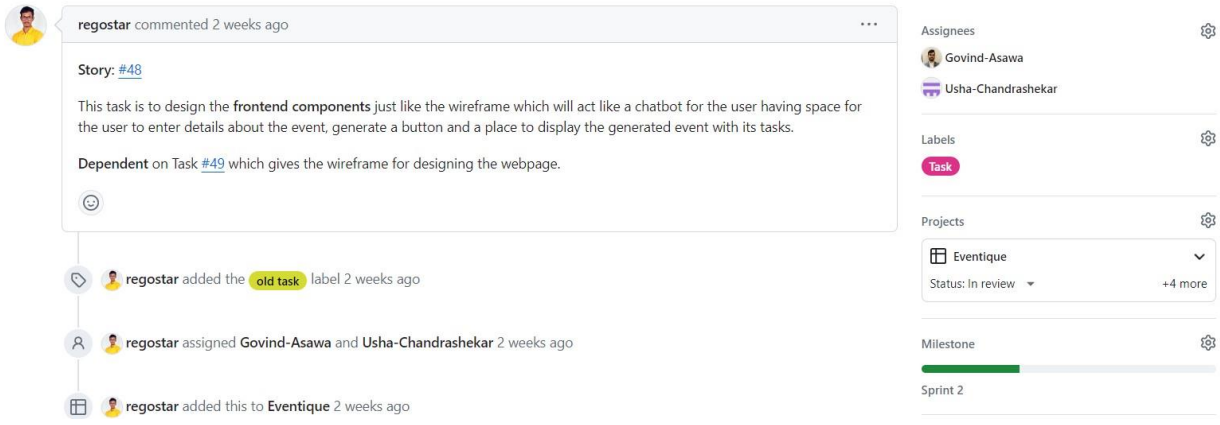
Projects: Eventique — Status: Done +4 more

Milestone: Sprint 2

Task : # 48 Frontend development for chatbot screen using the provided design #50

Edit | New issue

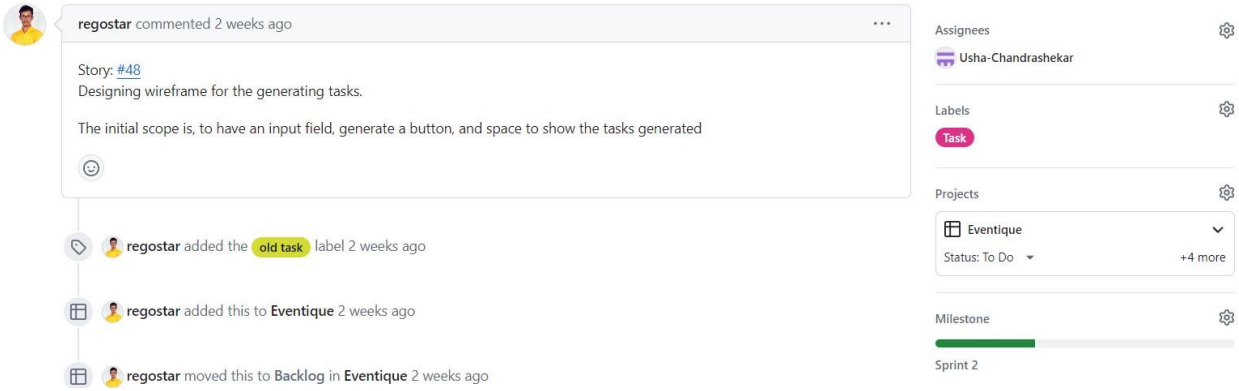
Open regostar opened this issue 2 weeks ago · 1 comment



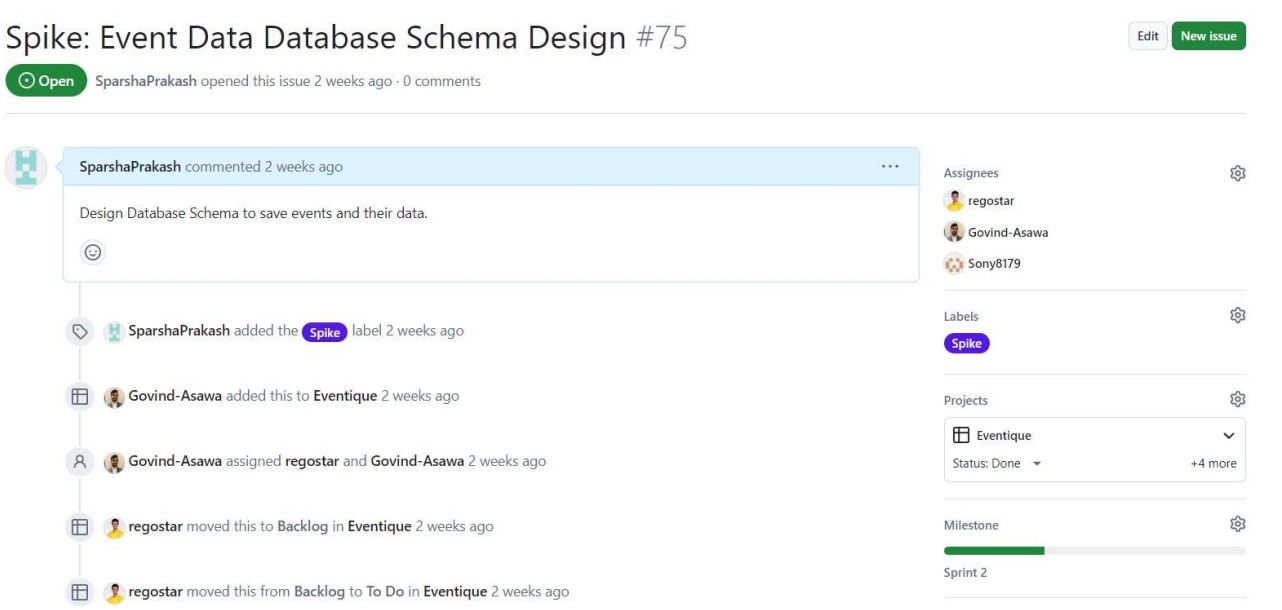
Task : #48 Wireframe for generating tasks given an event description by a user #49

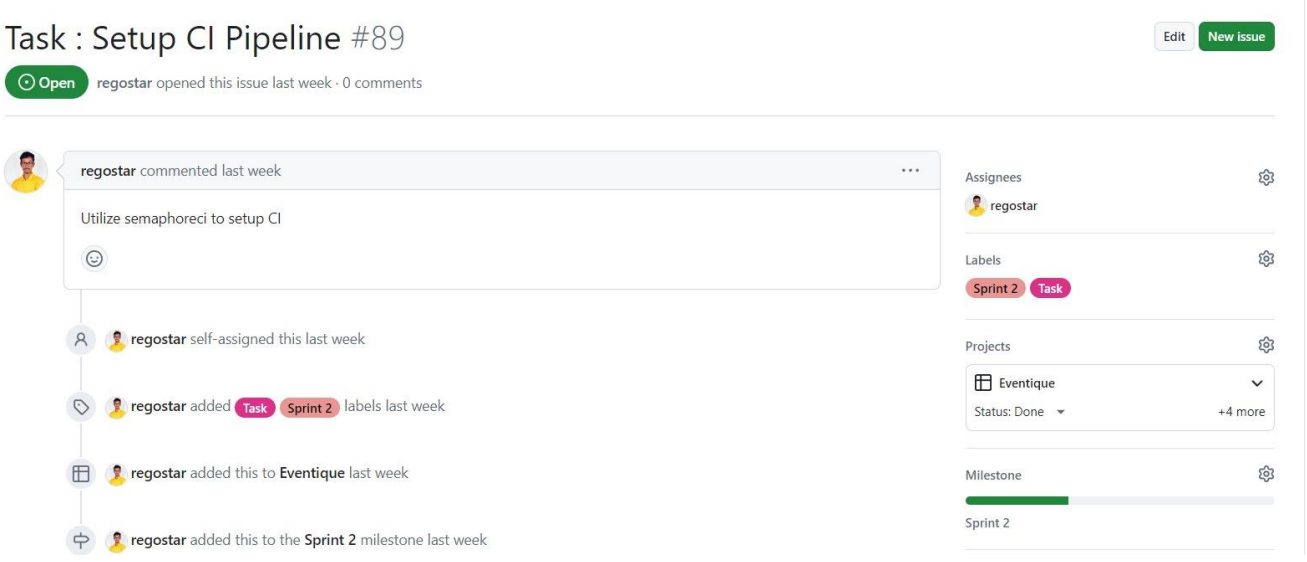
Edit | New issue

Open regostar opened this issue 2 weeks ago · 2 comments



Independent Spikes and Tasks

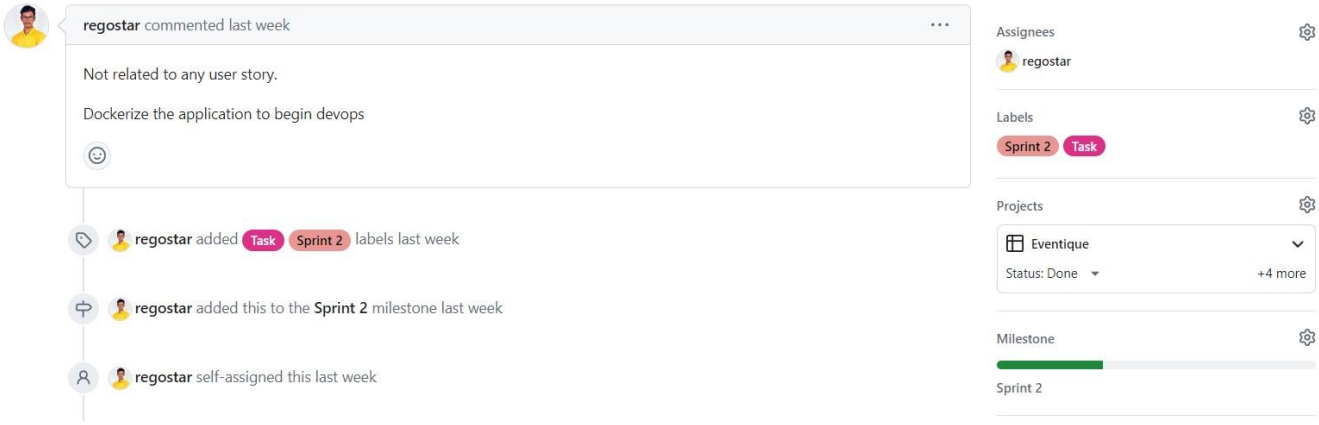




Task : Dockerize the application #88

Edit New issue

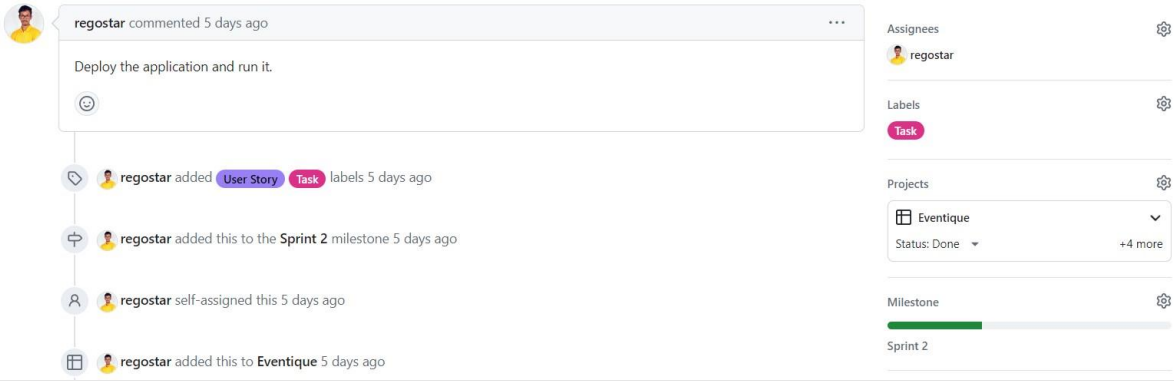
Open regostar opened this issue last week · 0 comments

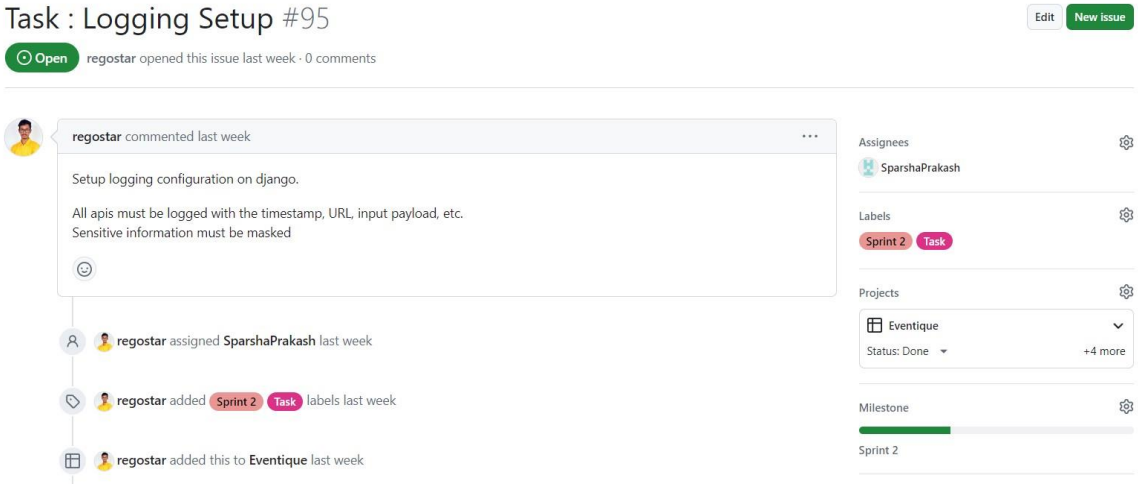


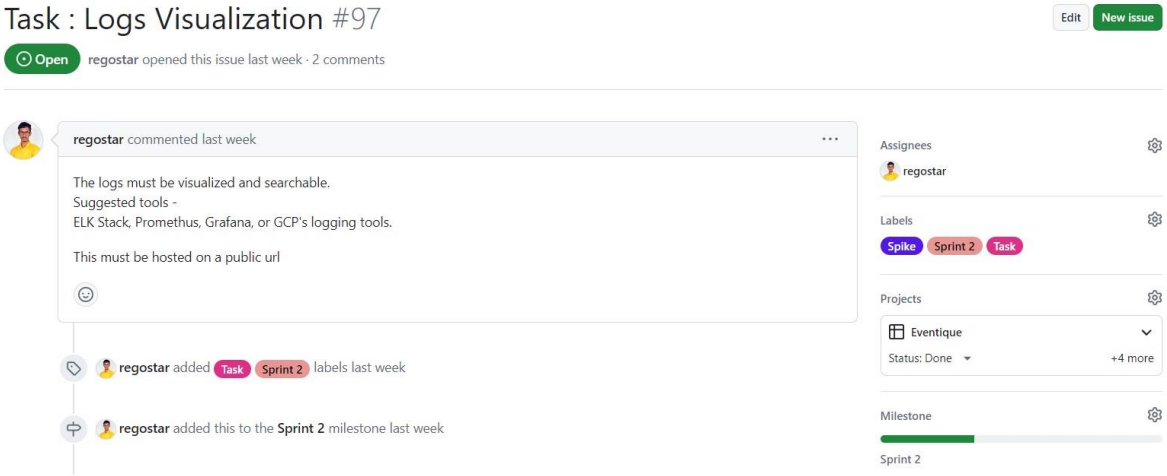
Task : Host the application on GCP Cloudrun #98

Edit New issue

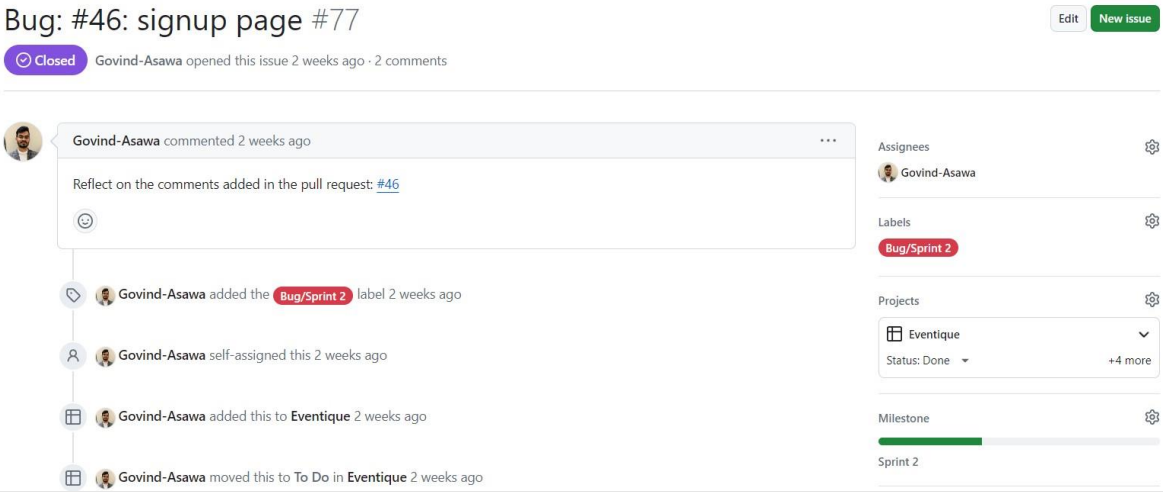
Open regostar opened this issue 5 days ago · 2 comments







Tasks Rolled Over from the Previous Sprint



Summary of Scrum Meetings

25th March, Monday@ 10.05PM (Zoom) | None Absent

- Was the first scrum call after the sprint planning call that was carried out March 22nd in-person.
- Few bugs in the signup page were identified from backlog testing and a task was created and assigned.
- Use of Gemini as LLM was finalized from the research spike.
- Simpler frontend wire-frame designs were being worked upon

27th March, Wednesday @ 10.05PM (Zoom) | None Absent

- Progress on bug fix & wireframe creation.
- Started designing Database schema, as part of spike.
- Started working on task related to implementing api-endpoint for event plan creation.

30th March, Saturday @10.05PM (Zoom) | None Absent

- Bug fix has been reviewed.
- Certain wireframe & UI blockers were asynchronously discussed and resolved.
- Started working on the homepage UI without calendar integration.
- Started working on the event generation screen.
- Basic structure of api-endpoint for event generation has been created.
- Database schema has been finalized. Found that MongoDB is not required for now, so going ahead with Postgres.

1st April, Monday @10.05 (Zoom) | None Absent

- Started working on calendar integration.
- Basic Database schema implemented.
- Event generation UI work in progress
- Started working on CI Pipeline

3rd April, Wednesday @10.05 (Zoom) | None Absent

- Calendar has been integrated successfully with multiple UI logic changes for Homepage.
- CI Pipeline completed
- Started working on creating api-endpoints to fetch tasks in a date range that needs to be displayed on the calendar.
- Started working on adding logging functionality in the backend.

5th April, Friday @ 1pm (Willis, in-person) | Adhoc unplanned | Absent: Sony, Ruthvick

- Restructured the backend code organization and api-endpoints as more clarity was added when working on fetching tasks.

- Completed working on api-endpoints to fetch tasks.
- Discussed blocker in Event generation(chatbot) UI creation and listed clear action items for the present scope of the work.
- Completed the basic logging setup.
- Started working on the issue to display user tasks on calendar.

6th April, Saturday @ 10.05pm (Zoom) | Absent: None

- Completed the homepage user-story.
- Started to work on Dockerization
- Started to deploy on Google Cloud Run

8th April, Monday @ 10.05pm (Zoom) | Absent: None

- Resolved the bug in the event recommendation system.
- Accomplished a basic version of CI/CD
- Deployed and hosted with a url.
- QA Testing started.

Summary of Scrum Review Meeting

Strengths: -

- Completing the committed user stories, better planning and execution of the tasks. Good split of tasks, spikes and user stories.
- Setup a Basic version of CI/CD Pipeline which helped us streamline development efforts.
- Logging pipeline setup enables us to detect bugs even before customers experience it.
- Improving collaboration by dockerizing the project so that the project runs on everyone’s systems irrespective of the underlying OS.
- Team members learnt to use GIT and started to get comfortable with creating Pull Requests and requesting for review before merging anything.

Weakness: -

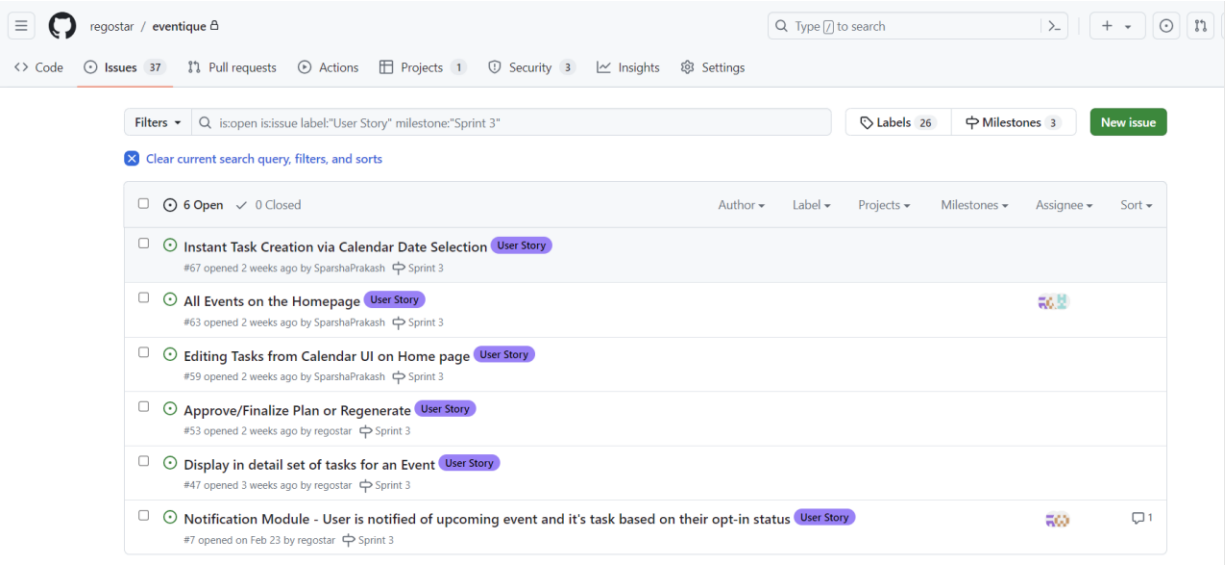
Better presentation required during the sprint review call with TA.
Need to improve integration of frontend and backend tasks seamlessly.

Steps to improve: -

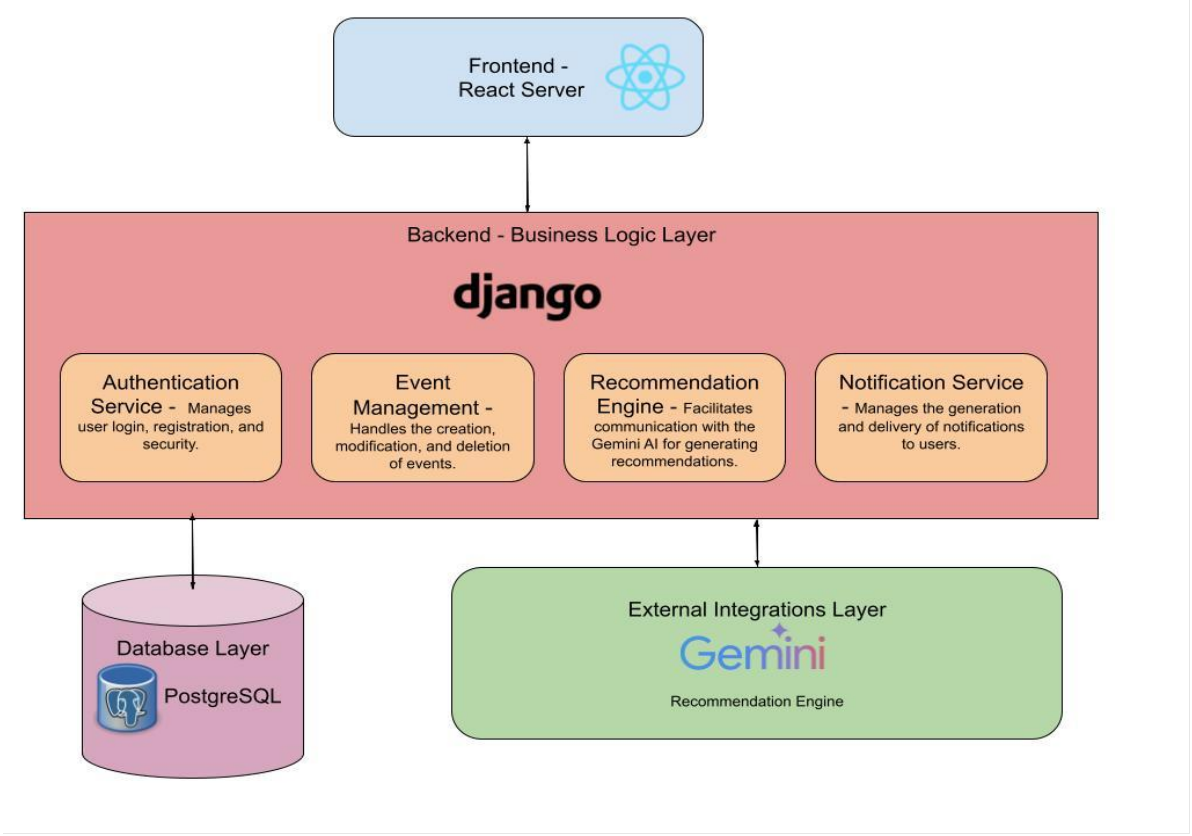
Proper rehearsal of what exactly we are going to discuss. Going from top to bottom from user stories. Giving a business overview instead of technical explanations. Help everyone in the team present during demos.

Tasks Rolled over to next sprint: - None

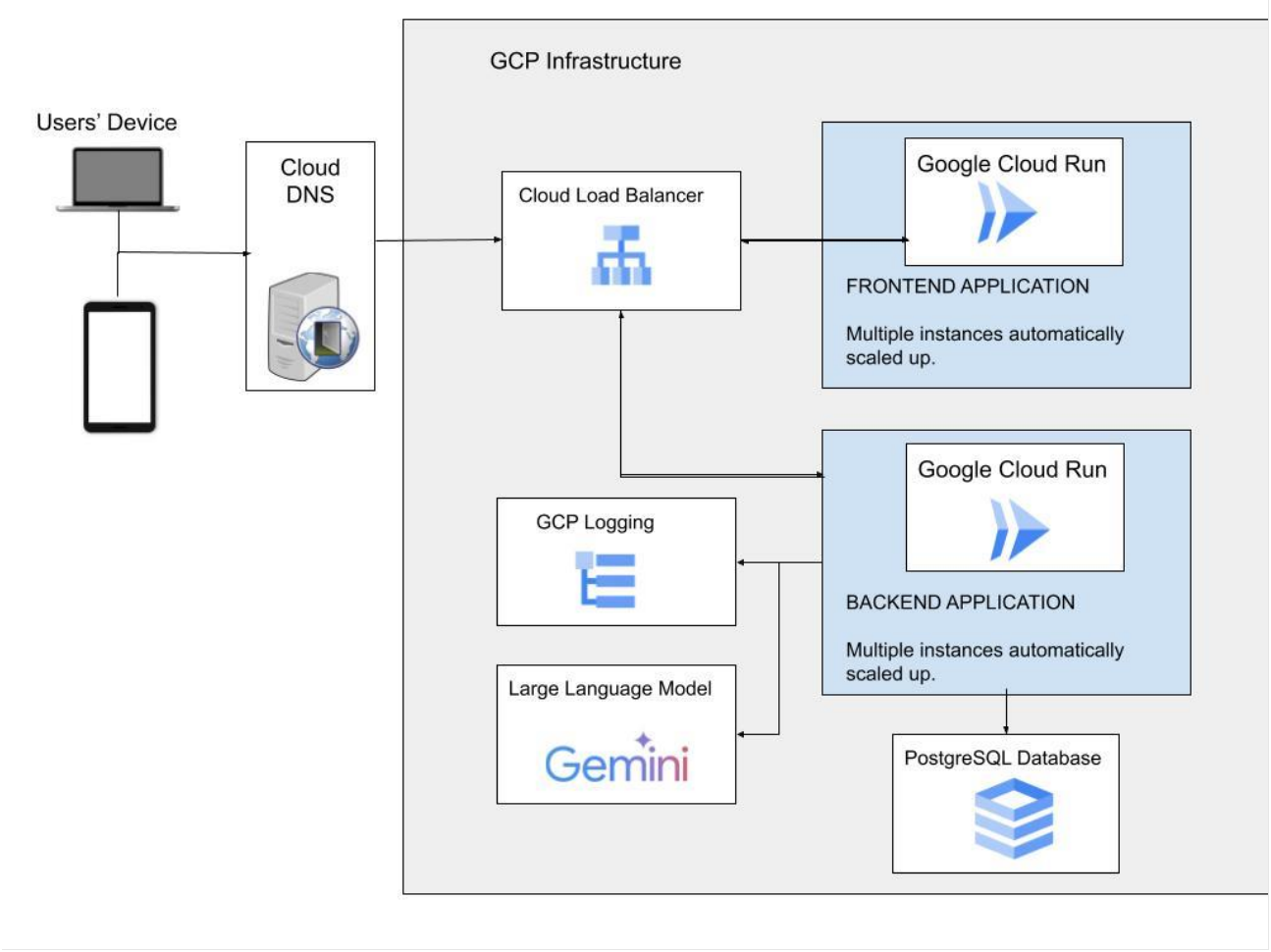
Sprint 3 User stories: -



Software Architecture – Layered Architecture



Cloud Infrastructure



Backend communicates with Frontend via Load Balancer (Nginx)

High-Level Design (HLD)

1. **Frontend (React.js)**

- Calendar View: Display tasks and events, with options to create, edit, and view in detail.
- Event and Task Management: Interfaces to manage the specifics of each event and task.
- Recommendation Engine Interface: Integrates with Gemini AI to provide event plan suggestions.
- Notification Panel: Displays upcoming events and task notifications. (sprint 3)

2. **Backend (Django)**

API Endpoints:

- Authentication (login, logout, register)
- Event CRUD operations
- Task CRUD operations
- Notification management
- Recommendation system integration

Business Logic:

- User authentication and authorization
- Event scheduling and management logic
- Task tracking and updates
- Notification triggering based on user preferences and event/task timings
- Recommendation engine interaction for generating event plans

3. **Database (PostgreSQL)**

Tables: User, Event, Task, Notification

Relationships and indices designed for optimal query performance

4. **Integration with Gemini AI**

- Communicates with the backend to receive event data and user preferences
- Processes the data to provide intelligent recommendations for event planning

5**. Deployment and Operations**

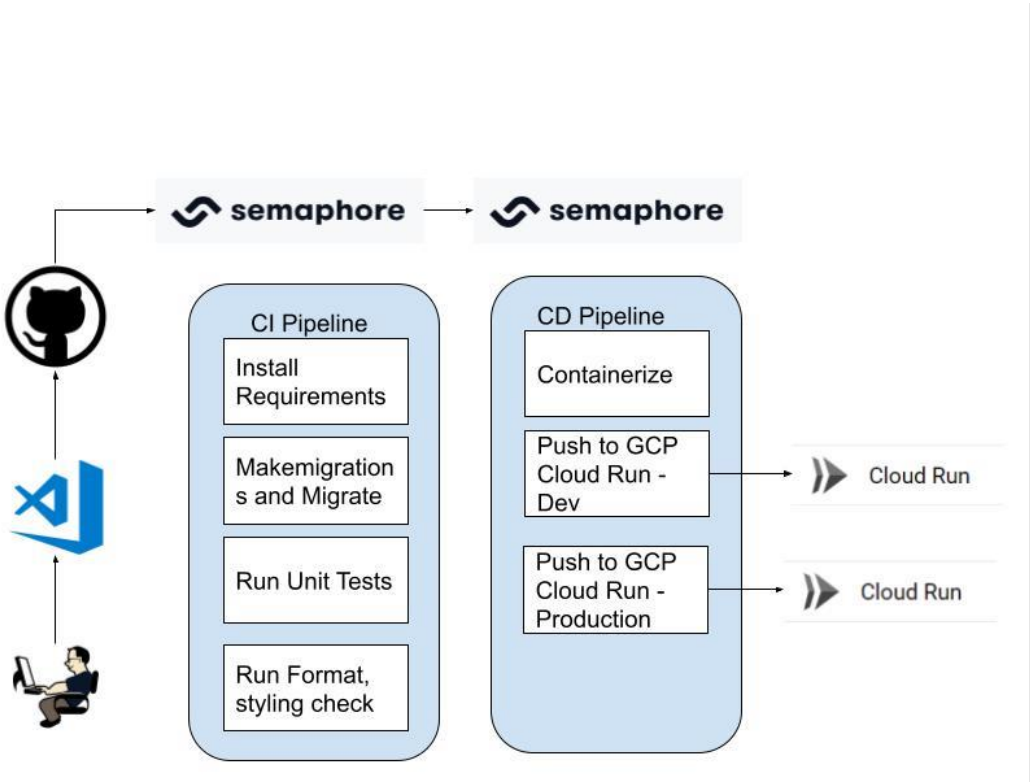
- Containerization: Application components (frontend, backend, database) are containerized using Docker.
- Continuous Integration/Continuous Deployment (CI/CD):
- CI with Semaphore to automate testing and build processes.
- CD for automated deployment of Docker images to Google Cloud Run, ensuring seamless updates and scalability.

6. Security and Compliance

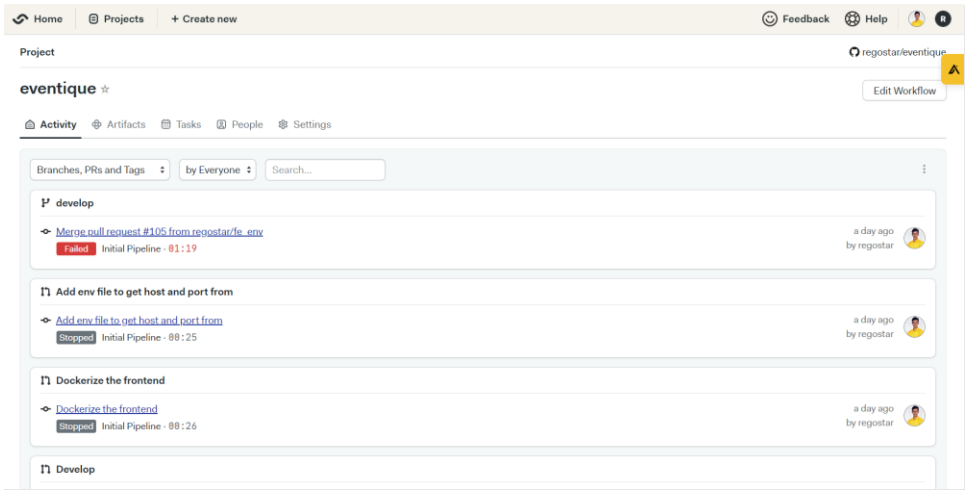
- Implement standard security practices (HTTPS, data encryption, user authentication and authorization, regular security updates).

CI/CD Pipeline

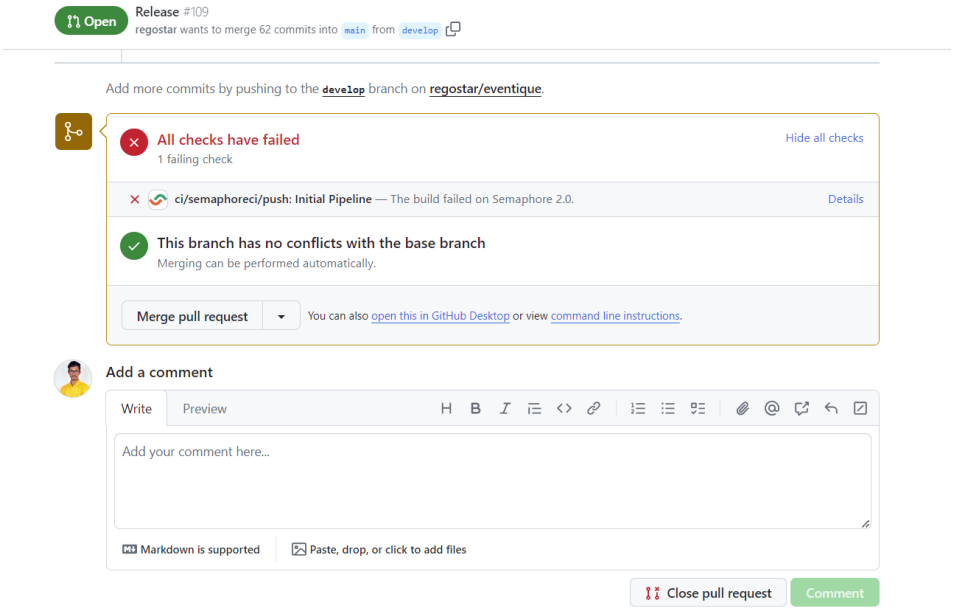
Semaphore Tool is integrated with GitHub

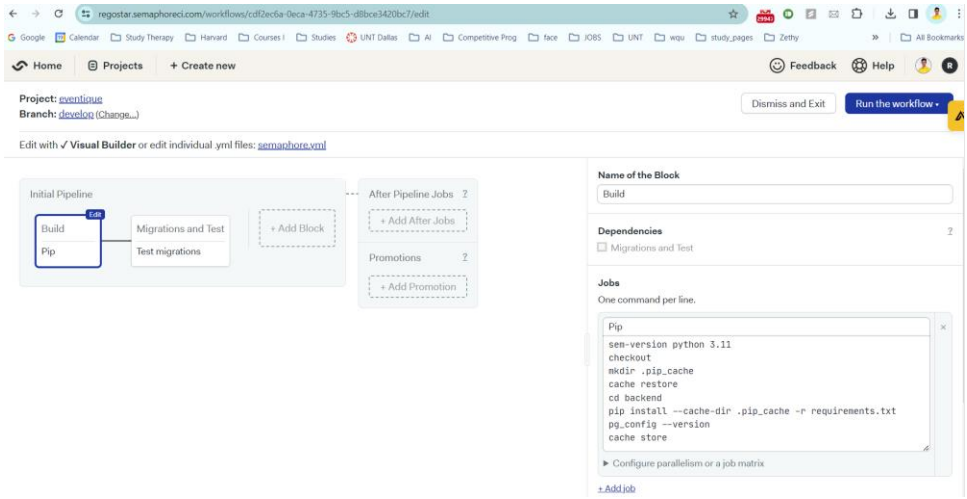


Semaphore - A CI/CD solution to streamline developer workflows.



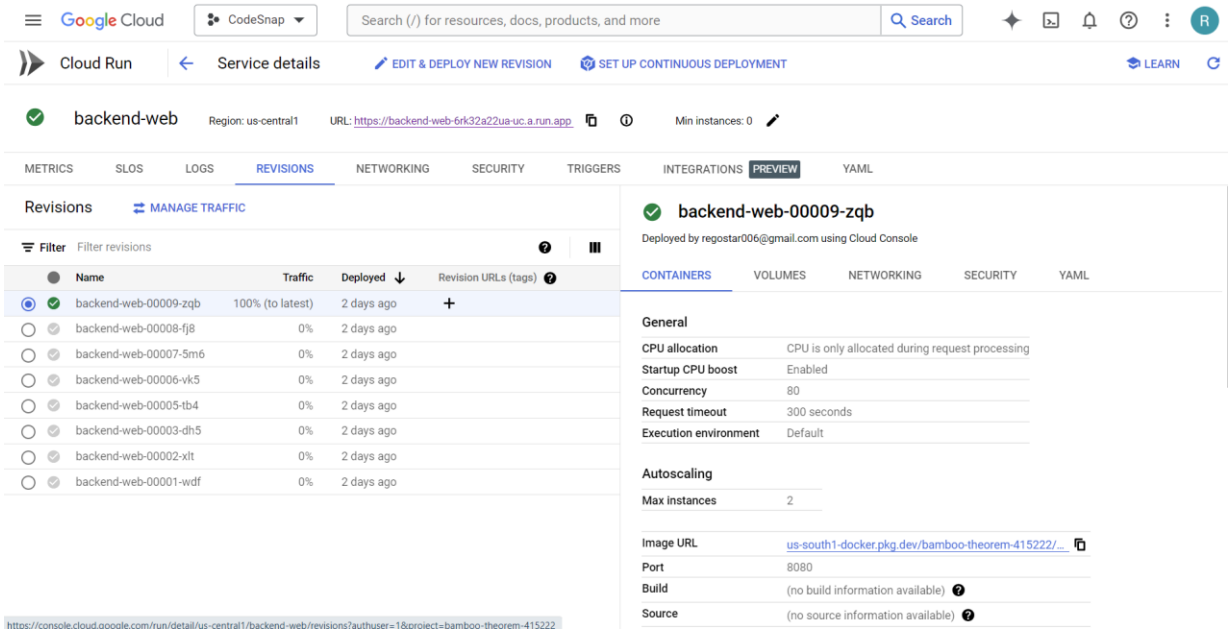
During every pull request, the CI pipeline is automatically run. The unit tests are run and it outputs if it failed or succeeded. Test Build is also done.

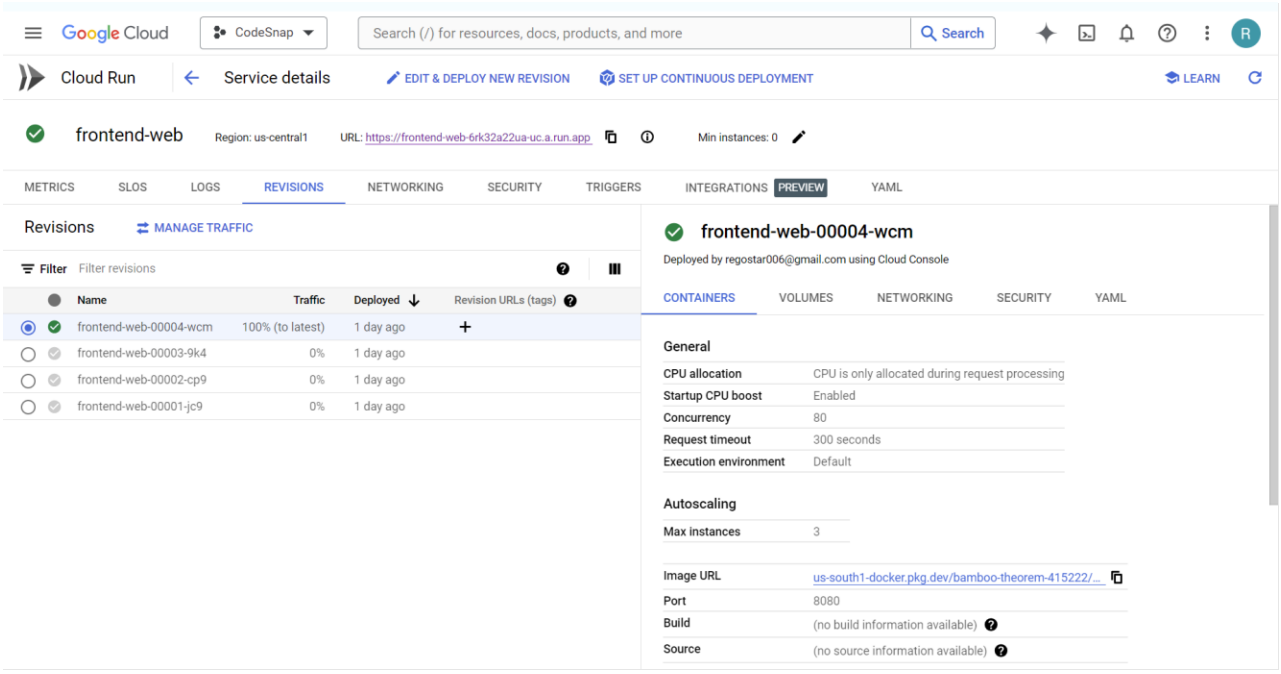




Various New tests, builds, containerization, etc can be added on to the existing CI/CD pipeline

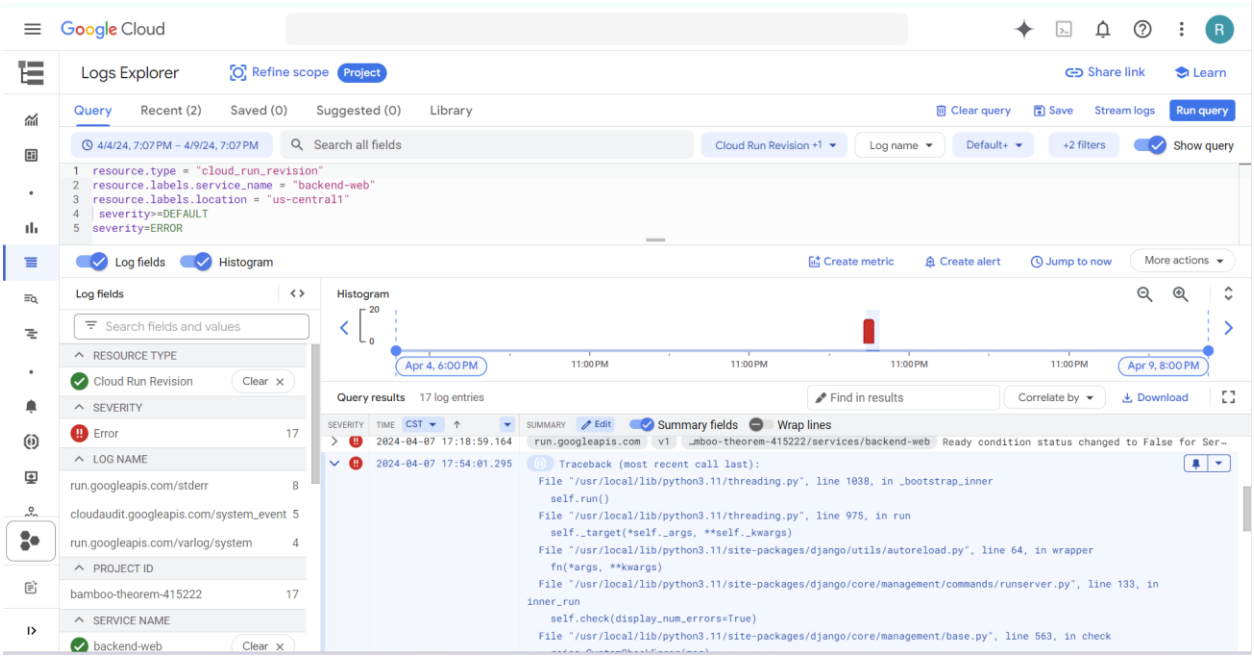
Semaphore tool will dockerize the project and ship it to GCP Cloudrun as indicated in the architecture diagram.



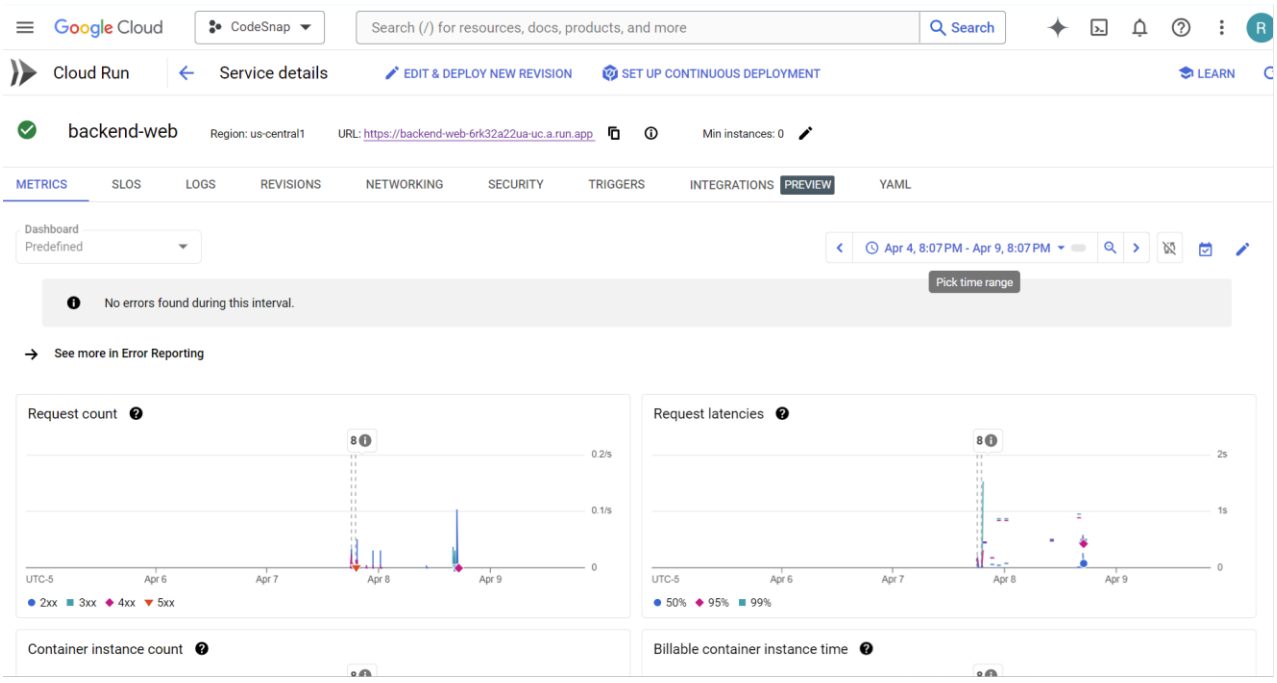


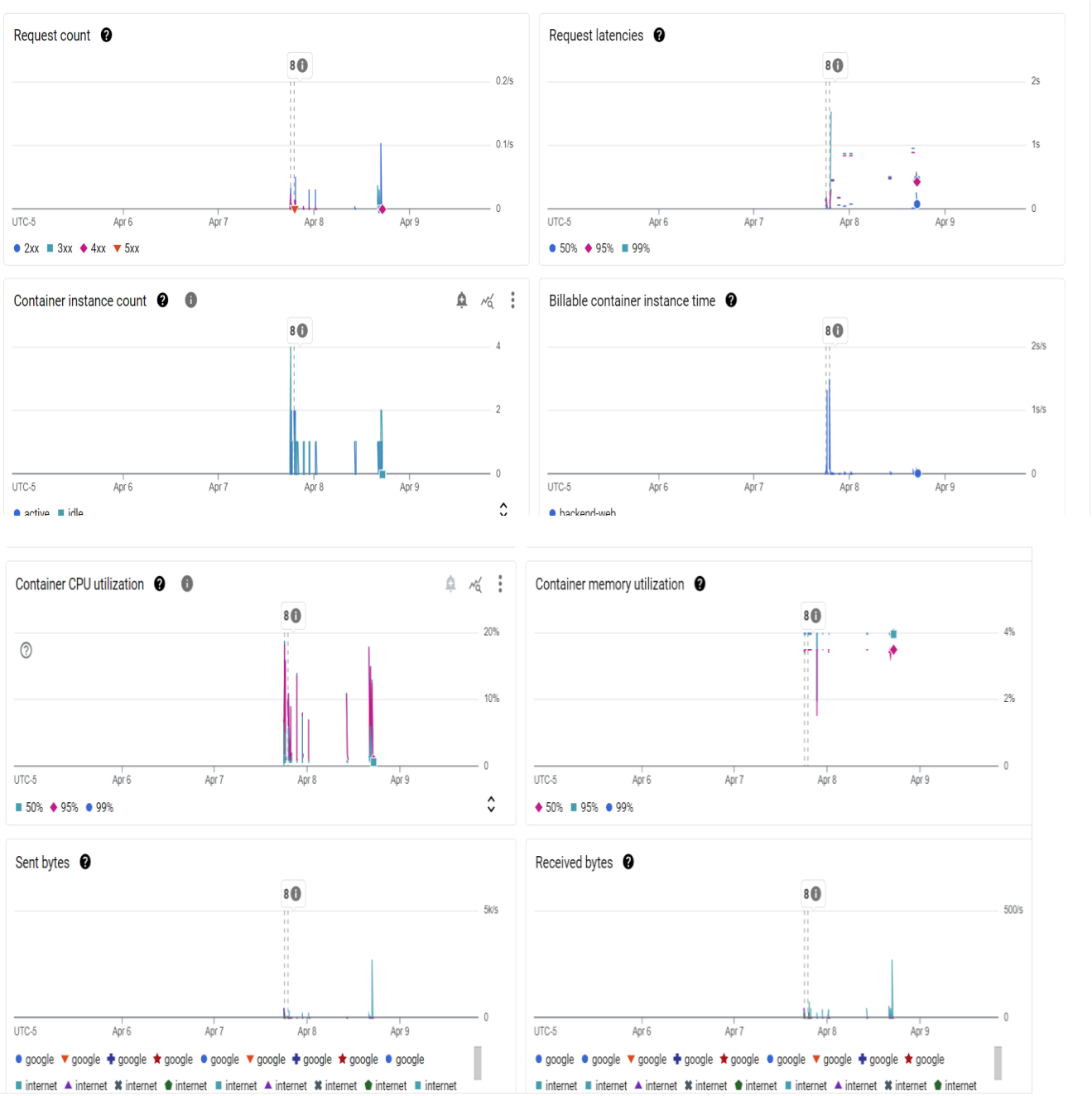
Logging :-

Utilized Google Logs Explorer to view App Logs



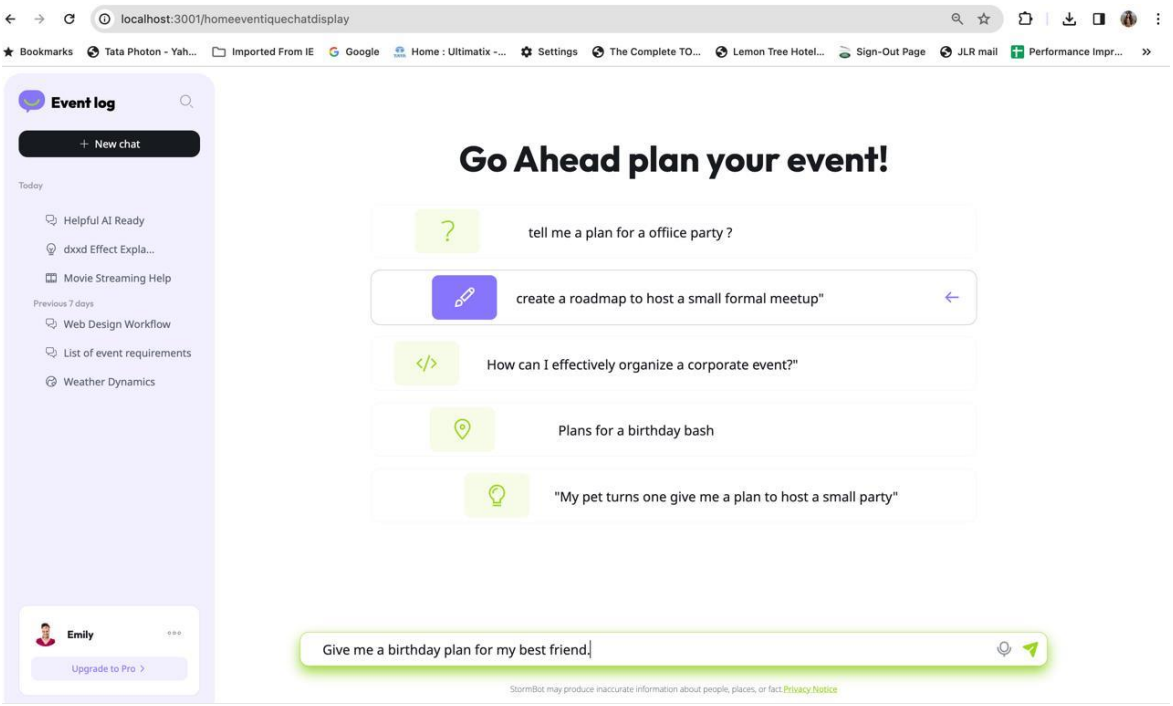
Utilized Cloudrun to see System Metrics for the instances we are running

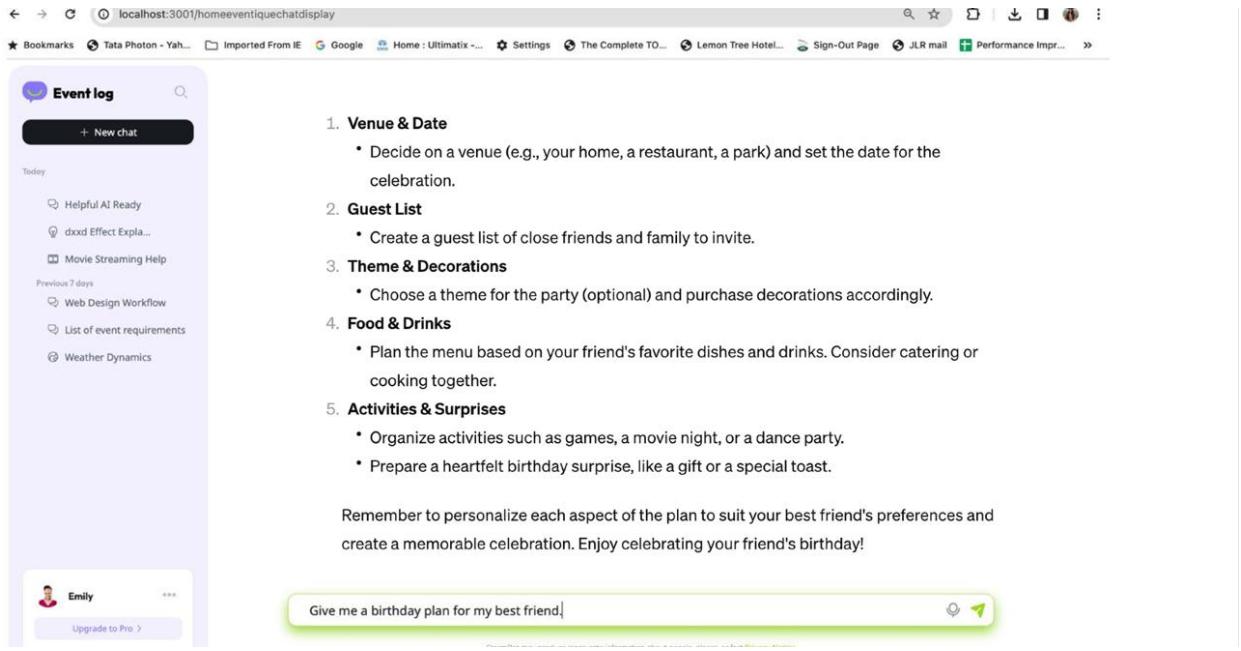




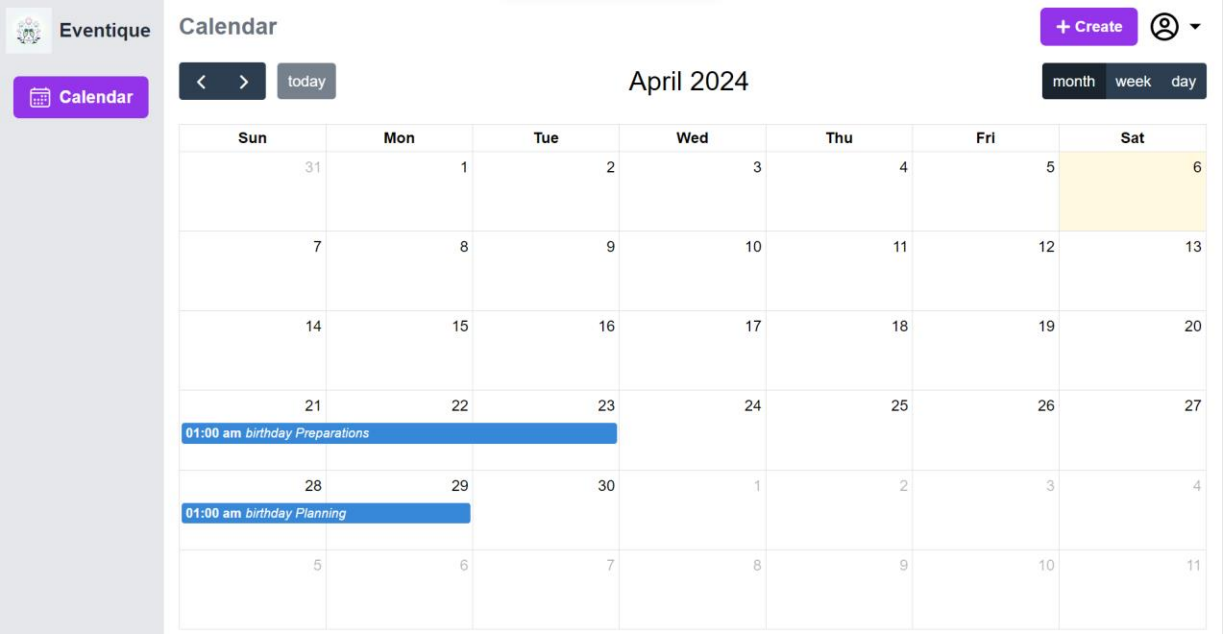
Application Demo

Chatbot Screen – User story - **Event Plan Recommendation**





Homepage – User Story - **Homepage having tasks on a Calendar**



DevOps Measurements

Process Measurement Metrics

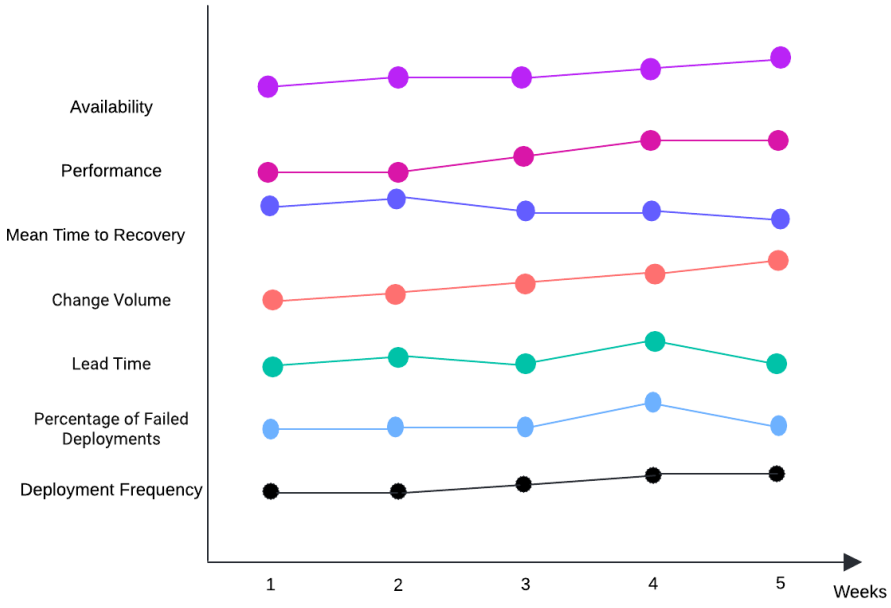
1. **Deployment Frequency:** Eventique maintains a deployment frequency of 2 deployments per week (3 in the last 2 weeks), reflecting our commitment to agility and responsiveness in addressing stakeholder feedback promptly.
2. **Percentage of Failed Deployments:** Out of 10 deployments in a month, only 1 deployment failed, resulting in a low failure rate of 10%, highlighting the effectiveness of our testing practices and deployment automation on a smaller scale.
3. **Lead Time from Development to Deployment:** The average lead time from merging code changes to deploying them to our development server is 10 minutes, showcasing our ability to rapidly move changes through the deployment pipeline.

4. **Change Volume:** Over the past month, our team has seen an average of 50 commits and 30 pull requests per month on GitHub, indicating consistent development activity and collaboration among team members.
5. **Mean Time to Recovery (MTTR):** In the event of an incident, our team can recover service within an average of 14 minutes, minimizing disruption to project development and ensuring timely resolution of issues.

Service Measurement Metrics

1. **Performance:** The average response time for key project functionalities, such as accessing project resources or submitting changes, is 800 milliseconds, providing a smooth user experience for project contributors.
2. **Availability/Reliability:** Eventique has achieved an availability rate of 99.5% over the past month, with only 2 hours of scheduled downtime for maintenance and updates, ensuring project resources are consistently accessible to team members. This metric is guaranteed by GCP.

DevOps Metrics Trends Chart



Week	Deployment Frequency	Failed Deployments (%)	Lead Time (hours)	Change Volume	MTTR (minutes)	Performance (ms)	Availability (%)
1	1	0	1	20	15	750	99.3
2	1	0	1.5	25	20	750	99.3
3	2	0	1.1	30	12	850	99.4
4	3	10	2	34	11	900	99.6
5	3	0	1.1	36	10	900	99.5