

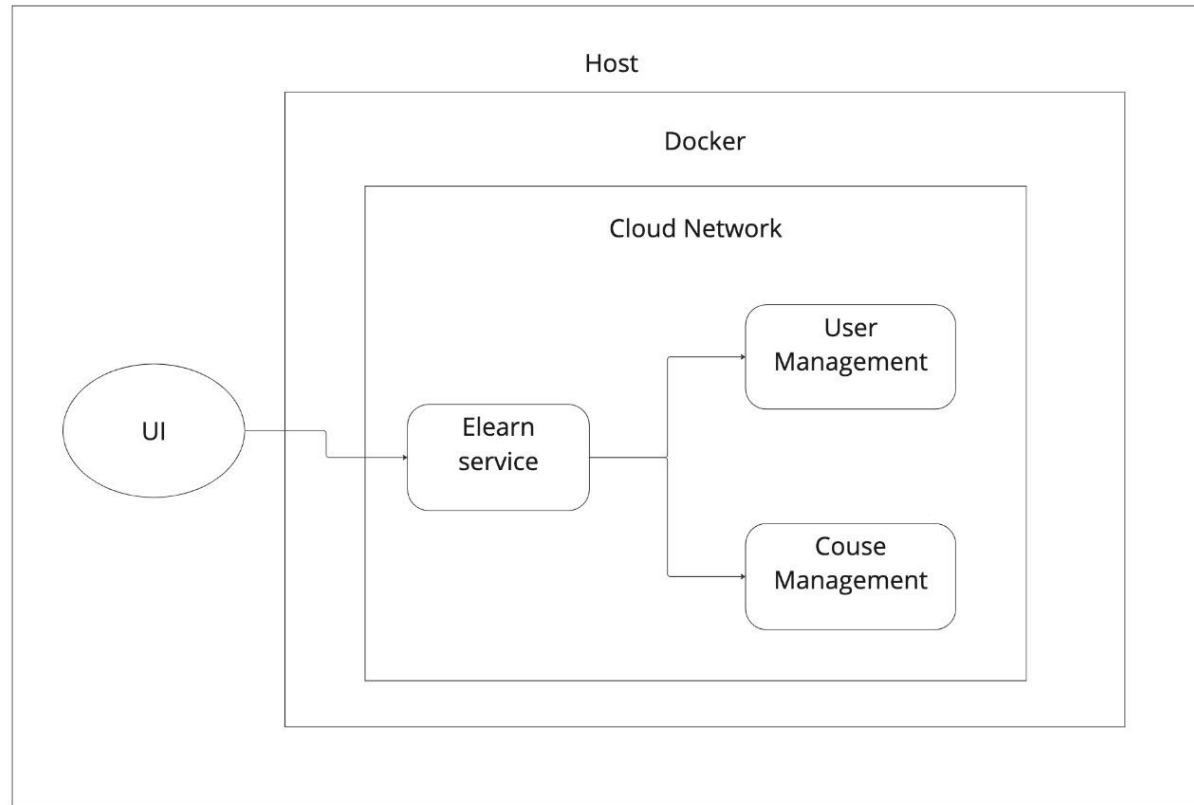
E- LEARNING PLATFORM

Janani Rangaraj(A25329086)

INTRODUCTION

- Two main services
 - User Service
 - Course Management
- Elearn Service acts as a bridge between the two services
- React application for user to interact with Elearn Service.

ARCHITECTURE DIAGRAM:



USER SERVICE

- Admin user manages user base
- Create and manage multiple users with login credentials
- Users can log in, view courses, add courses
- User data is securely stored and managed
- Admin can delete users

USER SERVICE : LIST OF APIS

- GET
 - /getuser/{user_id}
 - /getusers
 - /getuserid
- POST
 - /checkuser
 - /deleteuser/{user_id}
 - /adduser
 - /user/{user_id}/addcourses

USER SERVICE : LIST OF APIS



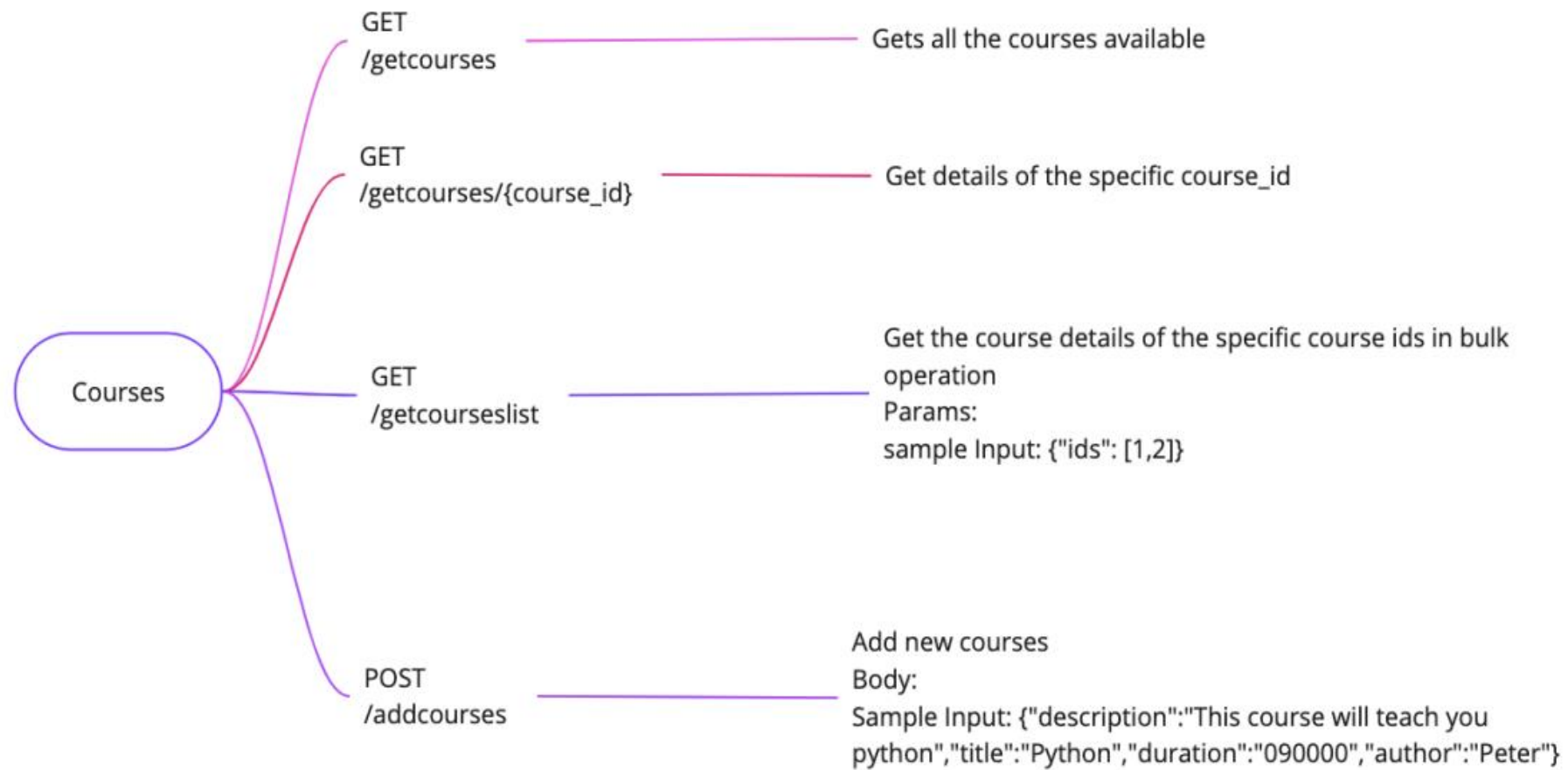
COURSE MANAGEMENT

- Manages courses effectively
- Users can view list of courses, add new courses, delete courses
- Search and browse courses
- Users provide course details for addition
- Users can select the course they want to delete and confirm the deletion

COURSE MANAGEMENT: LIST OF APIS

- GET
 - /getcourses
 - /getcourses/{course_id}
 - /getcourselist
- POST
 - /addcourses

COURSE MANAGEMENT: LIST OF APIS



E-LEARN SERVICE

- Intermediary service between User and Course Management Services
- Connects and enables seamless communication
- Provides centralized access point for all services
- Simplifies application architecture and improves efficiency

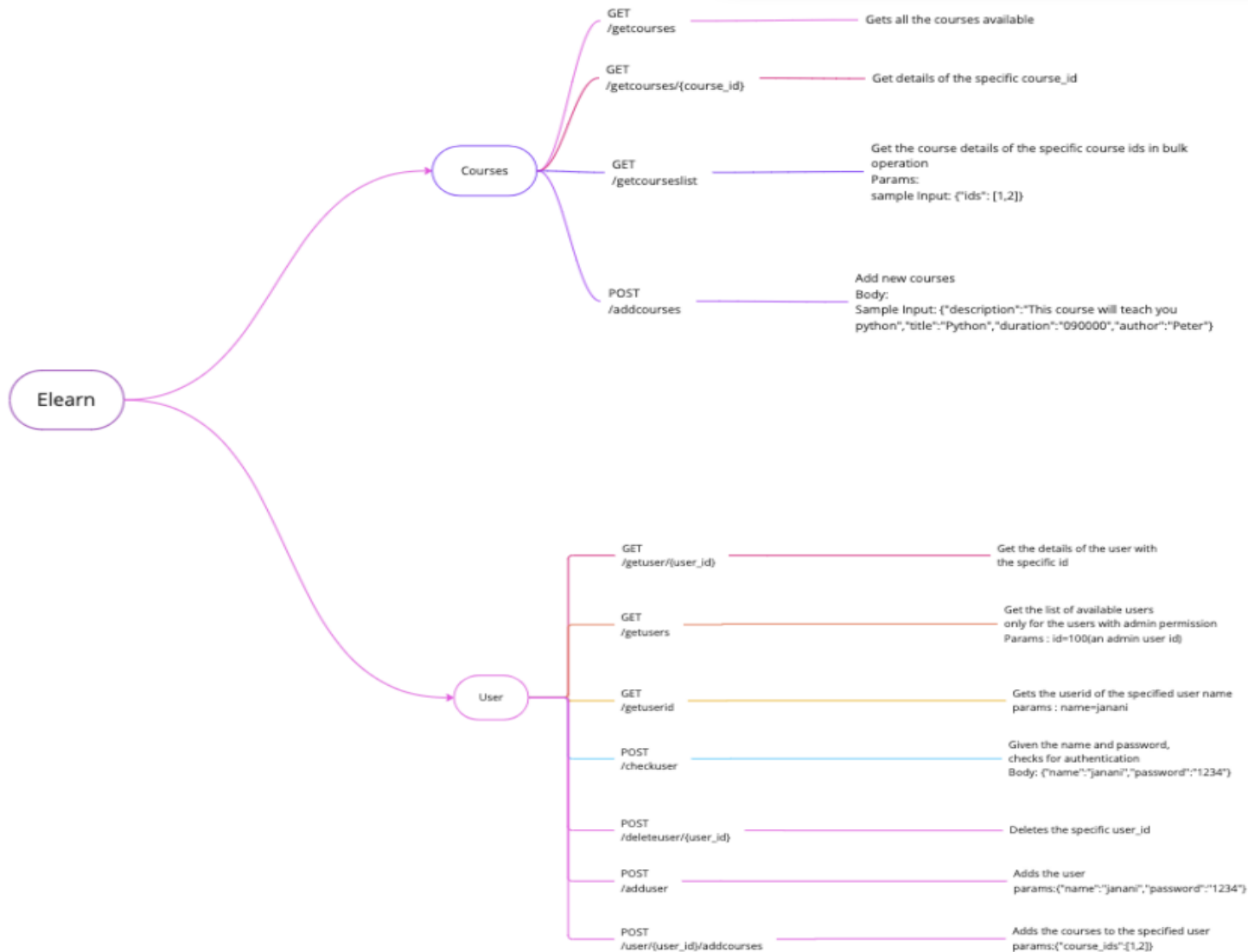
E-LEARN SERVICE: LIST OF APIS

- GET
 - /elearn/getuser/201
 - /elearn/getusers?id=
 - /elearn/getusers?name=
 - /elearn/getcourses
 - /elearn/getcourses/{course_id}
 - /elearn/getcourselist
 - /elearn/addcourses

E-LEARN SERVICE: LIST OF APIS

- POST
 - elearn/checkuser
 - /elearn/deleteuser
 - /elearn/adduser
 - /elearn/user/200/addcourse

E-LEARN SERVICE: LIST OF APIS



REACT APPLICATION

- Simple React application to interact with Elearn Service
- Users can access User and Course Management functionalities
- Provides user-friendly interface and smooth user experience

SOFTWARE USED:

- Python : Backend logic handling
- Cloud Firestore : Storing the user, course details
- Fast API : Provides API support for python
- Uvicorn : Asynchronous Server Gateway Interface to run the FAST API
- Docker : Hosting the services
- React : Front-End Application

DIFFICULTIES EXPERIENCED

- Connecting Python to Firebase Cloud:
- New platform, required time and effort to establish connection and persistence in the database.
- CORS policy issues:
- Services couldn't be accessed by another service running on a different port on local machine.
- Resolved after extensive research on the Internet.
- Docker setup challenges:

DIFFICULTIES EXPERIENCED

- Issues with printing logs and running the container.
- Took time and effort to resolve.
- Cloud network setup after deployment:
- Trial-and-error methods needed to set up.
- React application development challenges:
- Navigating pages with user details and making them persisted throughout the session.
- Issues with creating dynamic div element for displaying courses and availability for enrollment.