

Tech Stack and Architecture Choice:

- Frameworks used – Django Rest Framework for backend and React.JS for frontend.
 - React is a go-to choice for SPA.
 - Django has good support for authentication, caching and ORM to handle SQL data
- Data Flow
 - Patient (Authenticated user of role Patient) uploads a pdf file and additional information related to the file. The file is saved to local storage on the server side. The metadata is stored into file-metadata table along with unique id and file path.
 - When a patient or clinician chooses to download a file, a call is made to the file-metadata table with file id parameter and the file is sent in the response.
- API endpoints
 - POST /login
 - POST /register
 - POST /register-patient
 - POST /register-clinician
 - GET /patients
 - GET /patients/:id
 - GET /clinicians
 - GET /clinicians/:id
 - POST /patients/:id/documents/upload
 - GET /patients/:id/documents/:id
 - DELETE /patients/:id/documents/:id
 - GET /patients/:id/documents
 - GET /patients/:id/documents/:id/download
- Key Considerations
 - Scalable
 - Files are stored locally. Can be extended to use S3 bucket.
 - Handles error

- Security – Authorization is validated before sharing any data.
- CI/CD Pipeline Design
 - Can use Github Actions
- Infrastructure as Code (IaC) Snippets:
 - Docker sets up and runs the project