Business Slides

Application components

- Website
- Core Application
- AI

Website

To give online presence we will need static website

With Phases we can introduce below modules to website

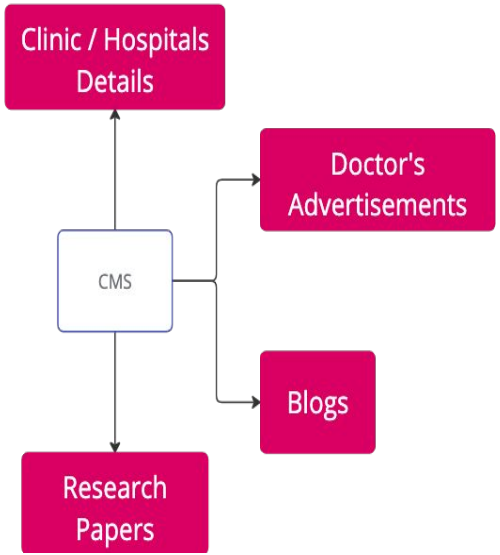
- CMS
- Chatbot etc... as a value added product
- Paid user with additional features like Doctor / Hospital suggestion etc...

Path to Enhance Website to make it feature rich

Web presence **Phase 1**

Static Website

Web presence **Phase 2**



Web presence **Phase 3**

Chatbot for Patient to validate it's symptoms

We can have more versions as well

Resource and timeline for Static Website

In my opinion, we can outsource website development. because

- The website is not our core focus.
- We need to hire a permanent designer for our product, which will increase unnecessary headcount and cost.
- Negotiations can be done with the vendor to get the best deal.

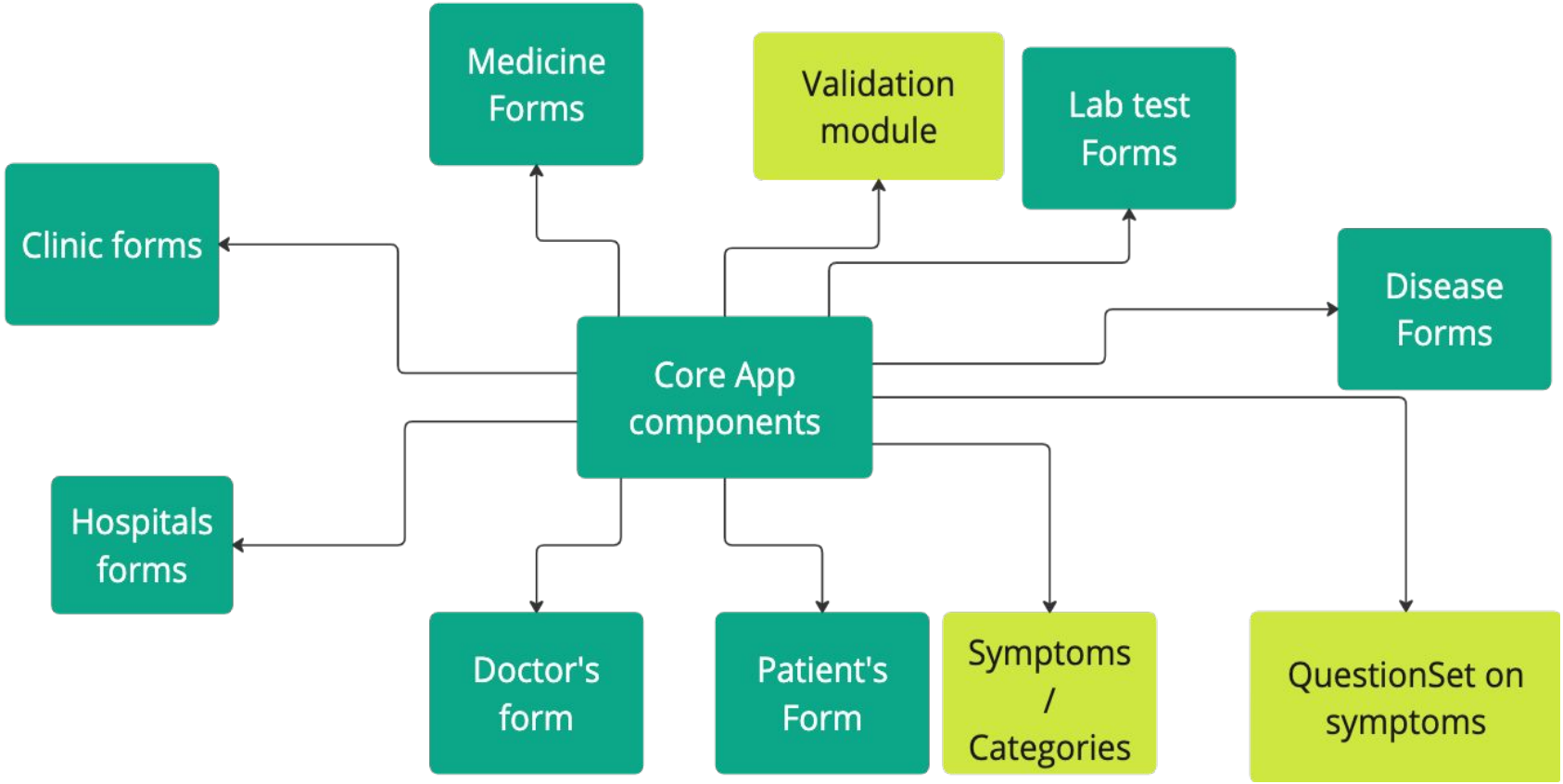
In case of inhouse development it will take 1 resource around 15-20 working days with prior planning

Core / Business App

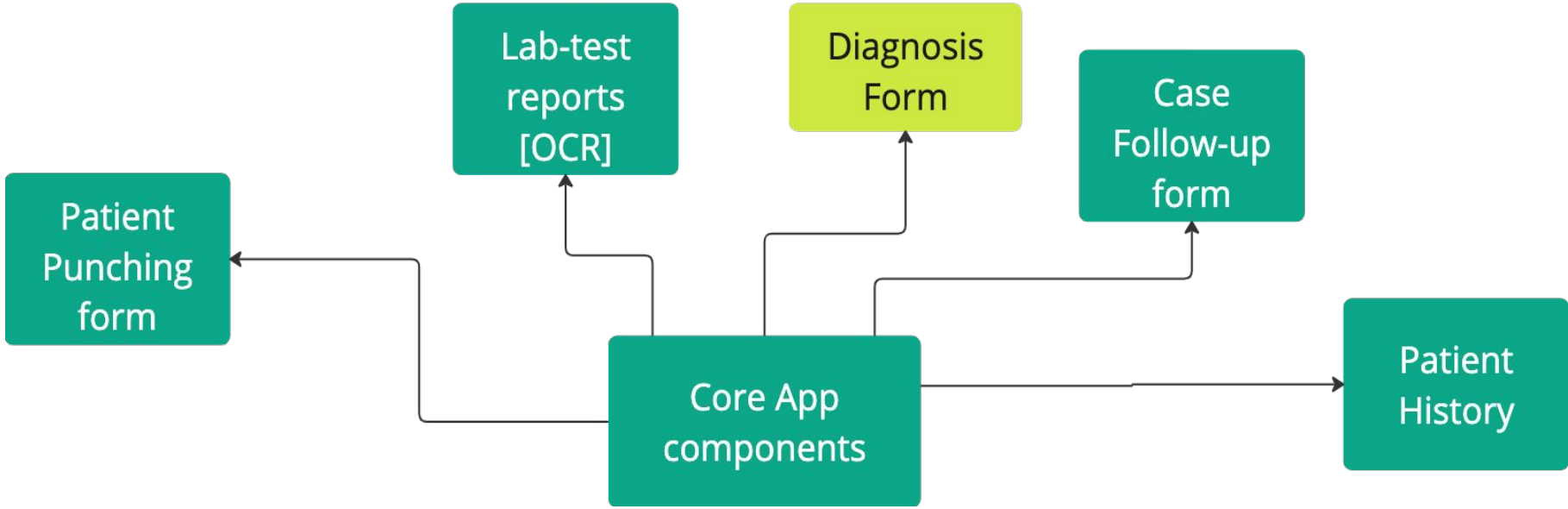
The core app will be divided into two parts.

- Front Office Application [**Purpose:** Diagnosis Process at Clinics]
  - Patient Form with General Details
  - Diagnosis module
  - Prescription module, etc...
- Back office application [**Purpose:** data entry and data validation]
  - Doctors', Patients, / Hospital Data entry and login details
  - Symptoms and symptoms-related questions
  - Disease, lab tests, and medicine Data entry
  - Validation module for the in-house doctor, etc…

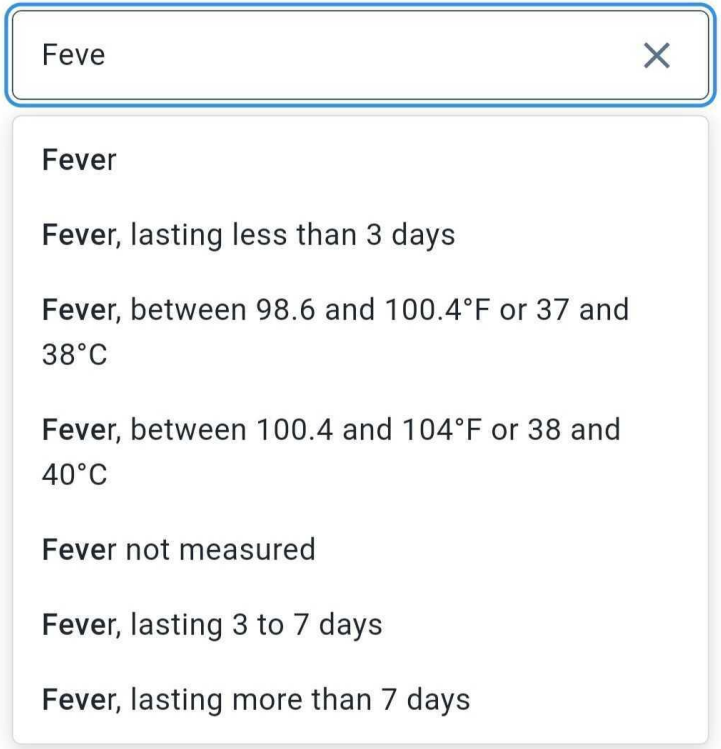
Phase 1 : Back-office Tool [Core Application]



Phase 1 : Front-office Tool [Core Application]



Diagnosis Form



Do you get headaches several times a day, and do they go away and come back after a while?

- Yes
- No
- Don't know

Why am I being asked this?

Report an issue with this question

Back

Resources and timeline for Phase 1

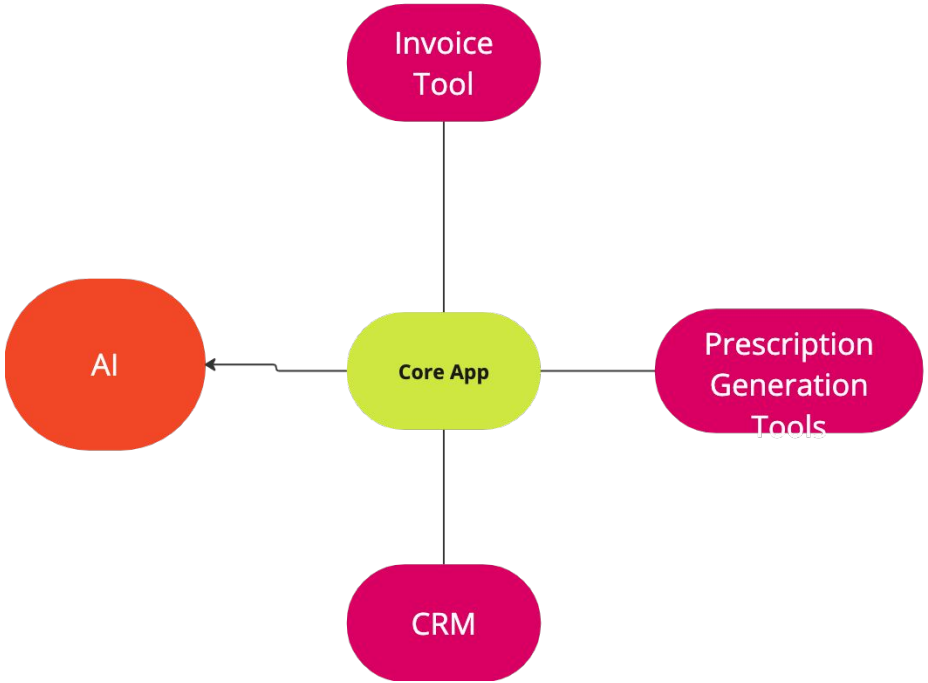
- Agile sprint based Development [incremental Approach]
- Back office tool development
  - Backend development 2 fullstack resources 30 days
  - AI / ML development : 1-2 developer approx [incremental process] [can be analysed and hired later]
- Front Office tools
  - Backend Development : 2 fullstack resources 30-40 days

Backend on Phase 2

- Use of AI to work on evaluate corner cases and decision-making.
- The Core App can be integrated into CRM to enhance the customer experience.
- The core app can be extended to integrate an invoice management system.
- The core app can be extended to integrate OCR for lab-test report reading.
- For separation of concerns, we can have microservices only if the team expands, as they have overhead in communication and maintenance.
- Clinic and hospitals tie-up and promotion application
- Paid user

Path to Enhance Business App

Phase 2 onwards: Core App Extension



Use case for an AI

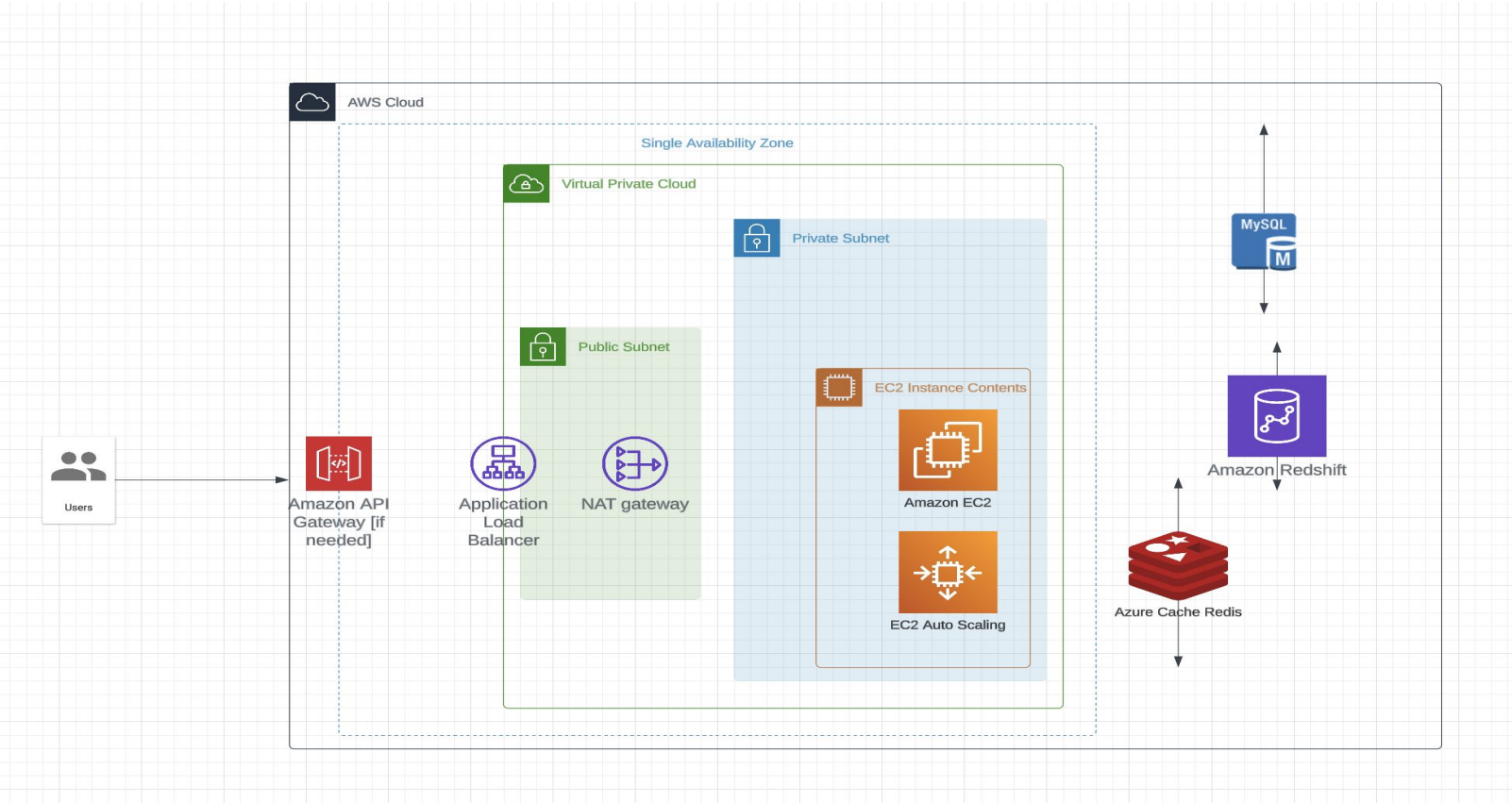
- Based on symptoms, ask/trim a set of questions.
- Based on the patient's history, suggest suitable medicines.
- Eliminate or trim the probability of multiple diseases based on the received input.
- Based on the trained model, past experience suggests an effective lab test.

Tech Slides

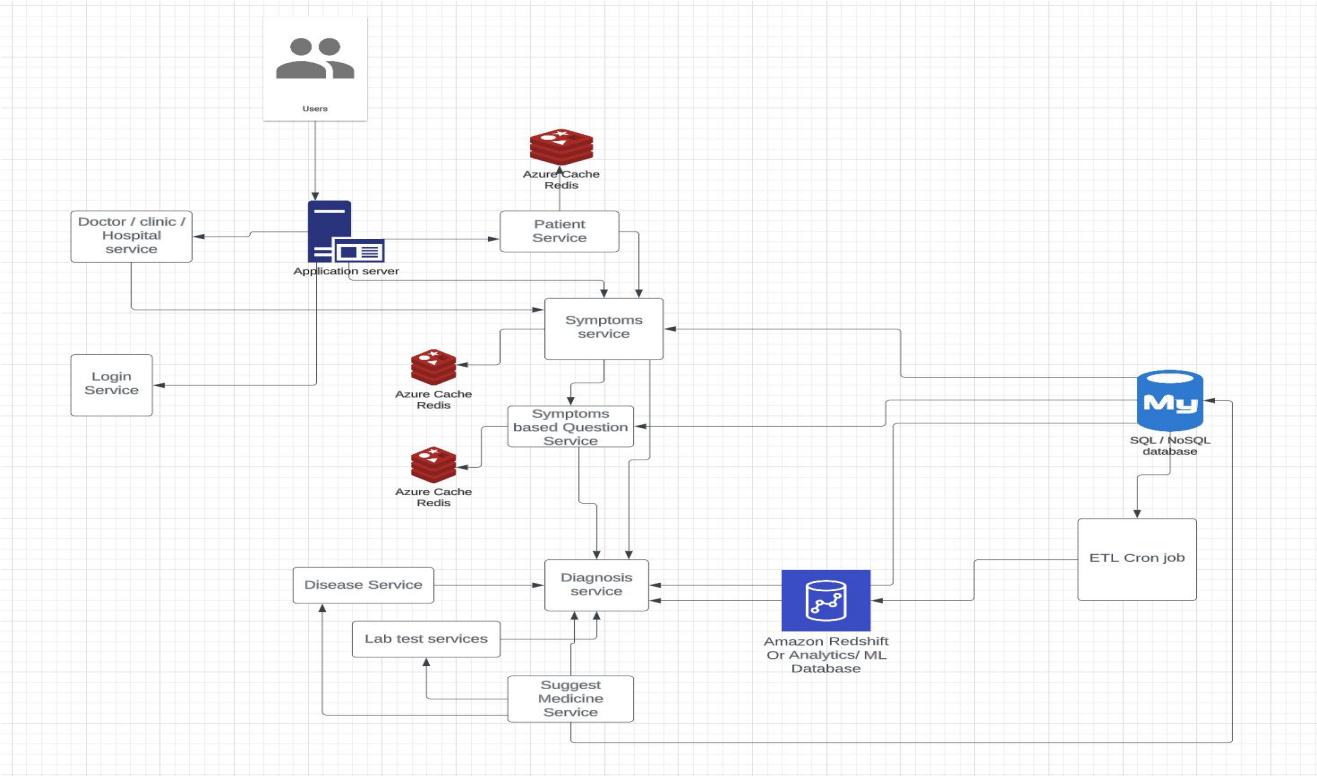
Tech stacks

- FE: React / Angular [SPA]
- API : Go / Kotlin [Rest/ GraphQL in certain use case]
- Cloud Infrastructure : AWS / Azure
- CI / CD : Gitlab / github / CircleCI
- OLTP/ SQL / RDBMS D/B : MySQL & NoSQL [as per use case]
- OLAP D/B : [Analytics] Cassandra / Redshift as per cost bandwidth
- Redis and ElasticSearch for caching
- Apache Kafka
- Ticket management Tool
- Error management Tool
- Monitoring tool [on later phase]

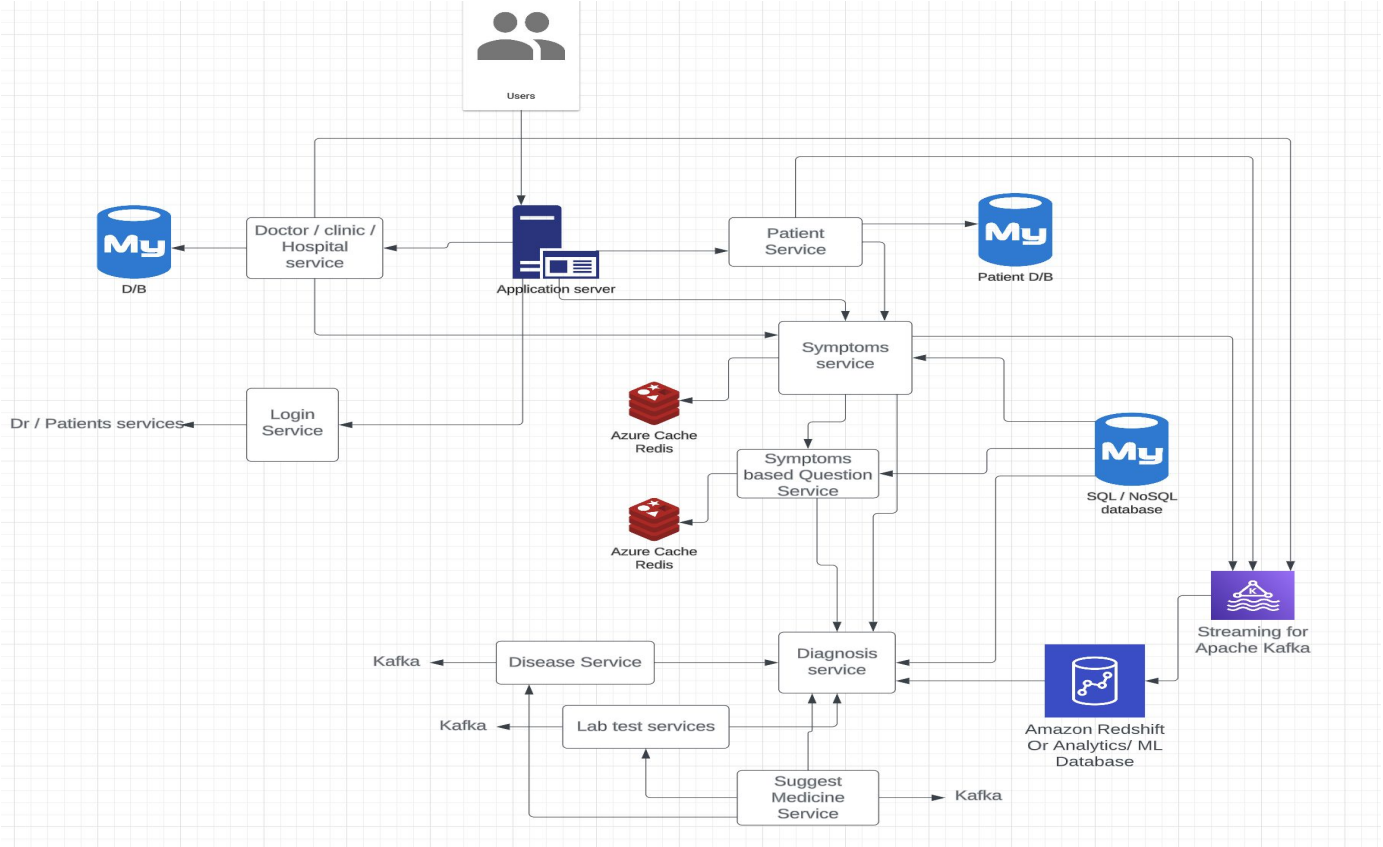
Application Architecture Diagram



service Architecture Diagram [Initial Phase]



Service Architecture Diagram [if microservice architecture]



More details

- Data feeding and validation can be manual as well as Paid API can be used to feed data
- Once the Product and the budget grows then we can proceed to have **micro-service based architecture implementation** as it has cost and communication overhead
- I will rely on Automated test cases later if needed only then we can plan to onboard QA
- In future product will required error tracking and monitoring tools for robust performance
- Service architecture diagram does not include minor services like location etc.

Thank You

Herat Dhruv