MediPal by Pat Piedad

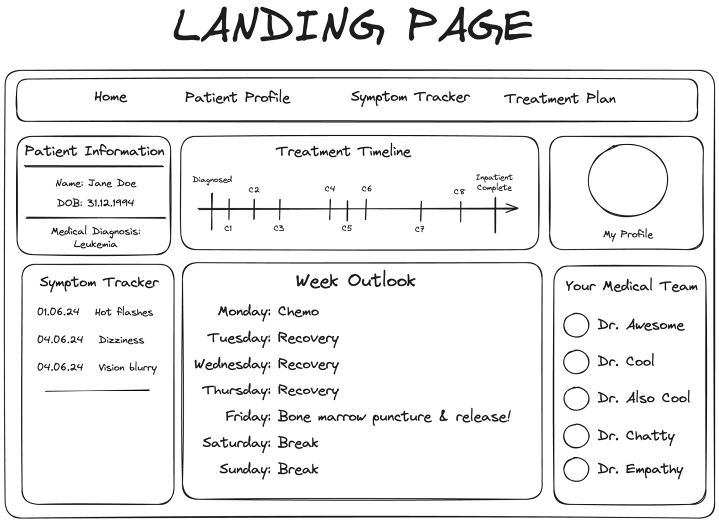
Thursday, July 25, 2024 07:16

<u>Program Idea:</u>
This program will serve as a type of "patient management software" that can be used for both patients and medical teams (to include doctors, nurses, social workers, etc…).

<u>Official Description:</u>
The official repository for MediPal. MediPal is open source medical software that gives users true access to their holistic health picture. This application is meant to simplify getting the important information to the person in ways they are able to understand.

The initial program will be a simple application geared towards the patient, where the patient is in long term care. This application will display the following:

- Name
- DOB
- Medical Diagnosis
- Treatment Timeline
- Personal Profile2
- Symptom Tracker
- Week Outlook
- Your Medical Team



LANDING PAGE

Home   Patient Profile   Symptom Tracker   Treatment Plan

Patient Information
Name: Jane Doe
DOB: 31.12.1994
Medical Diagnosis: Leukemia

Treatment Timeline

My Profile

Symptom Tracker
01.06.24 Hot flashes
04.06.24 Dizziness
04.06.24 Vision blurry

Week Outlook
Monday: Chemo
Tuesday: Recovery
Wednesday: Recovery
Thursday: Recovery
Friday: Bone marrow puncture & release!
Saturday: Break
Sunday: Break

Your Medical Team
○ Dr. Awesome
○ Dr. Cool
○ Dr. Also Cool
○ Dr. Chatty
○ Dr. Empathy

- The patient and/or medical team can upload treatment plans, medical reports, medicine plans, appointments, and other pertinent documents.

- The patient is able to log items into their symptom tracker. The symptom tracker will have a specific format to help readability and information sorting.

- The week outlook can either be manually populated OR, ideally, queried from the database that holds the treatment plan and appointment schedules

- Possible integration of "secure mail" that allows for direct communication with the medical team

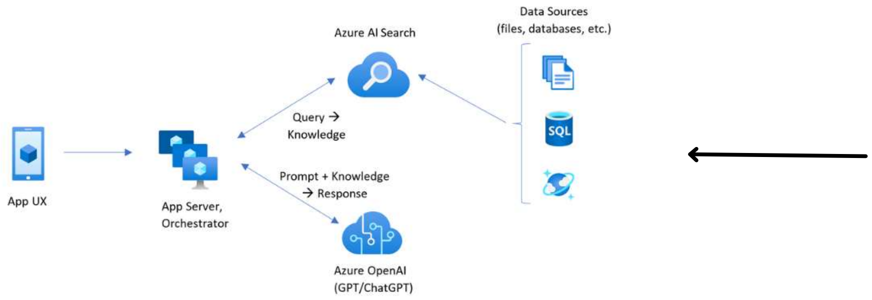
- All data can be exported into a readable format, most likely excel or PDF

- Database will be created to house all data

- Application server will be created to interface with additional modules of the application

---

The next stage of the application development process is the integration of RAG (retrieval augmented generation), Azure AI Search, and Azure Open AI.



<u>Technical Process Flow</u>

1. MediPal's app UX greets user with designated interface and populates that interface with information pulled through the app server from the database.
2. The App UX provides the prompt to the app server.
3. The app server provides the prompt to the Azure AI Search API.
4. Azure AI Search queries the patient's information in the database, indexes the information, and provides it back to the app server in a consumable format.
5. The app server pushes the original prompt and retrieved patient information to Azure Open AI.
6. Azure Open AI uses it trained LLM to process the data against the prompt and provide a proper response back to the app server.
7. The app server relays the response to the App UX.
8. The App UX displays the response to the patient.

Implementation is best described through a use case example:

<u>Use Case</u>

1. Marie is a long term patient diagnosed with Acute Lymphoblastic Leukemia (ALL). She is currently in the inpatient phase of her treatment which lasts roughly 12 months. Her medical team visits her regularly throughout the week but she finds that she often has general medical questions throughout the day that can't be immediately addressed. She also finds herself forgetting questions during her medical team interactions and they are not always available in a short time period for personal meetings due to the volume of patients in the cancer ward.

2. Marie logs into "MediPal" with her own username and password. She is greeted by the program's user interface that displays a lot of useful information. However, she wants to know how her last complete blood count (CBC) compares to the previous few results as she is feeling a bit weaker but does not know if it is just a "mental thing" or if her cells are actually lower than before.

3. Marie opens the chat function in the program and asks "I'm feeling a bit weaker today. How is my blood doing?"

4. MediPal answers with a detailed response describing her blood latest blood report that was uploaded into the database on her profile, comparing it to the last 3x records already in the database. After quantifying the major differences in blood count, it describes the possible symptoms that she could experience based on her current blood picture.

5. Marie reads MediPal's response and feels reassured that she is experienced physical symptoms due to her decreased blood count. This makes sense as she just received chemotherapy three days ago, and she normally begins to feel those effects in roughly three to five days.

Phases:

This project will be executed in phases:

Phase 1 (Prototype)

- Create app interface using .NET MAUI Blazor Hybrid or Blazor Web Assembly
- Single role (patient), future phases and iterations will add other role such as doctor, nurse, and social worker
- Hardcode any data that needs to be displayed in the app
- Find collection class to use for records
- Add login screen, hard code username and password (placeholder for future authentication with database)

Phase 2 (Add database)

- Look at SQL Lite as database to be used
- Add proper login procedures
- Add authentication, possibly consider 2FA
- Add LOTS of data to database, both "personal data" specific to the patient AND "base data" to act as comparison information for future AI APIs to utilize for comparison models

Phase 3 (AI Integration)

- Integrate Azure AI Search API
- Integrate Azure Open AI API
- Possibly integration of periodic "doctor check" of responses

----

Inspiration:

This project is inspired by the experience of my wife and I working through her cancer diagnosis and always having communication issues / information disconnects with the medical team. We were at the regional University Hospital that had the best team, but due to an overburdened medical system and the natural friction that occurs on teams, we encountered a lot of friction in knowing what was going on.

I want this project to not only provide relief to patient pain points, but also to provide a relief to the medical team's burden to keep patient's informed and constantly cared for.

Questions

☑ What framework am I using to create this project?
  - ○ MAUI Hybrid Blazor
  - ○ Blazor Web Assembly

□ How does Azure AI Search work, API?

□ How does Azure AI Search query and pass information?

□ How does Azure Open AI receive, process, and pass information?

□ Is MAUI/Blazor compatible with Azure AI Search / Azure Open AI?