cmyHealth: A Centralized Electronic Medical Records System

Peachtree Ridge High School
Tejas Shah, Mohamed Suufi, Adit Sharma

PROJECT DESCRIPTION

- In 2018, one large hospital in the US, Mayo Clinic, spent a total of $1.9 billion on their electronic medical records system, server, and database [2].
- In 2020, during the CoVID-19 pandemic, many hospitals struggled to find essential supplies including respiratory masks and gowns. Many more hospitals struggled to find space to place patients with the virus as funding was limited [1].
- The huge sum of money spent on electronic medical records system across the nation is better used if reallocated towards research, facility upgrades, and purchasing essential supplies especially during pandemics as we see today.
- A large chunk of information (only 72.9% of prehospital information is logged by the receiving hospital staff) is lost as patients are transferred from EMS personnel to the emergency department at the hospital [4].
- Transferring data across different medical institutions requires time and effort some may not have as some data is stored on CD-ROMs or flash drives when being sent to another medical institution or formatted differently compared to the receiving institution resulting in lost time and money as registrars manually input the information [5].
- Patients do not have transparent access to their own medical records or who accessed them.
- This project aims to to create a cheaper and easy to access healthcare database that can safely and efficiently transfer patient records while coming with a NFC component that incentivizes healthcare institutions to use a centralized system and grants medical professionals access to a patient's medical records on-the-go saving time, money, and lives.

OBJECTIVE

The main objective of this engineering project is to create a cost effective centralized electronic medical records system that can help hospitals make better use of their money while also being an efficient way to transfer medical data across different medical institutions

DESIGN CRITERIA

1. System must be able to support many different types of data across many different hospitals
2. System must have at least 10 types of basic essential information (Name, Number, Conditions, etc.)
3. System must have the ability to include at least 100 different hospitals, all seamlessly interconnected.
4. System must be able to hold medical imaging files like x-rays, ultrasounds, MRIs, and other medical documents.
5. System must have an easy-to-use interface, accomplished via React.js and Terra framework
6. System must load and transmit data fast - operations should take no longer than 20 seconds
7. System must have an authorization system built with JWT and Auth0 to help ease security and privacy concerns
8. Interact easily with the NFC-tagged card
   - people who have never used the product before should be able to figure out how to connect the Card to the Application
9. System must comply with HIPAA regulations in terms of handling of data and account information

Specification	Prototype	Final Design
Holds all file types	Yes	Yes
Allows for 100+ users with no strain	A little network strain	Yes
System swiftly loads information	Yes	Yes
User-friendly	No	Yes
User Profiles	Only ~10 test cases	Multiple
Authorization system	No	Yes

PLATFORM DESIGN

DATABASE DESIGN

- A prototype was constructed using solely a single MongoDB collection which could store patient data in the form of multiple JSON documents
- This was not a web application, but rather a console-based test system with no front end access.
- The data is organized into different categories including but not limited to:
  - User
  - Patient
  - MedicalHistoryDiagnosis
  - Procedure


```
_id: ObjectId("5dc5b80c98e2e116e67b9c9a")
> contact_info: Object
> race: Array
  active: true
  language: "ENGLISH"
  name: "Tejas Shah"
  preferredName: "ConTejas"
  birthdate: 2002-08-07T00:00:00.000+00:00
  sex: "MALE"
  bloodType: "A"
  ethnicity: "NOT_HISPANIC"
> emergency_contacts: Array
  __v: 0
```


Figure 1. Sample of a Patient Data Object

TESTING

- In order to test the system's ability to hold data, we used the console interface to load the database with as many different file types and file sizes as possible to see what would work and what didn't - all files we threw at it were successfully loaded and accessible via the Document Schema
- To check the security of the API itself, we used a tool called Postman in order to try and get into the data while not "logged in" in various ways, such as using GET, DELETE, and POST requests
- While the database was very secure and would not allow us to get in whatsoever, we still felt that more security was needed for the application itself, so we utilized JSON Web Tokens and an Auth0 authentication system in order to further make sure no user data could be accessible except by registered Doctors or the Patients themselves

OVERALL DESIGN

- Front-end utilizes a Medical-System design tool called Terra in order to better streamline our application alongside many other industry-grade systems which would already be in place in other hospitals all around the country.
- If the user who's signing in is a Patient, they will only be able to see their own profile and all the information of the accounts whom they are connected to via Family.
- The NFC-tagged card contains a URL which references a patient's profile. If you attempt to access it without being logged in, you will be forced to login before being able to see any medical information.
- Role-based authentication system that keeps medical information only in the hands of those who need it per their job.
- After the authentication page, the doctor/user gets navigated to their dashboard, where they are able to see any scanned patient's crucial medical information and all documents related to their health which are stored in the system
- MongoDB collections follow a schema that's simple to understand and easy to migrate to from most existing data formats including SQL, JSON, CSV, and more.
- Physicians can look users up using either their unique patient ID or scan their NFC chip which can be read through article of clothing and thick wallets.
- Uses Auth0 and a JWT system to maintain security with the implementation of SSL certificates and closed ports to keep data secure.
- Patients can see who accessed their data to ease privacy concerns.
- Able to store ANY file type using the Document MongoDB Schema object
- Card only references ObjectID to a patient in the system and has no personally identifiable information.
- Complies with all HIPAA and COPPA regulations due to our use of Auth0 and the general security of the database [3].

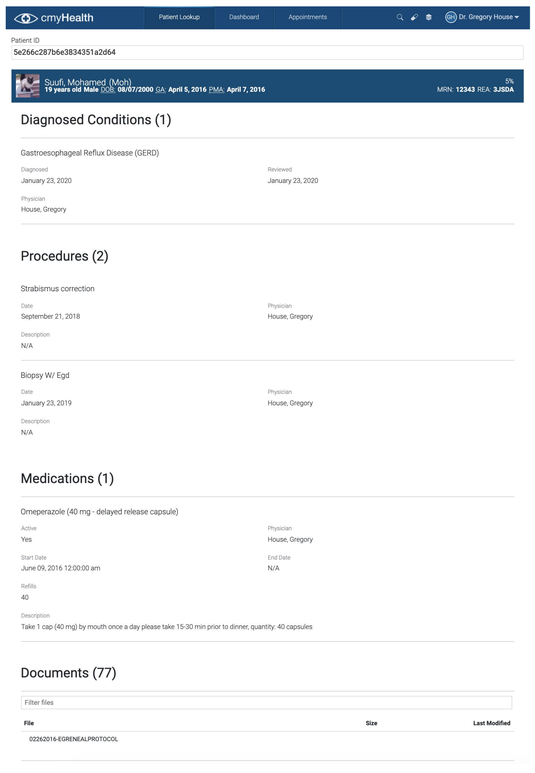

Figure 3. User Landing Page, pulled up via the scanning of an NFC card

ENGINEERING ANALYSIS

- The main concept on which this project rests is the concept of network connectability - to easily have your app be implemented in almost any hospital setting and be integrated with the current infrastructure
- In this way, we can have a standardized way of storing data across as many hospitals as possible, reducing the frustration users might experience when switching hospitals or medical providers
- This standardization also heavily reduces the insane costs that hospitals spend on setting up their own network systems run off of physical servers or rented cloud storage space
- Terra UI allows us to make the frontend identical for all hospitals
- Auth0 authentication allows Hospitals to plug in with existing authentication systems via already existing user profiles - Auth0 is one of the most flexible ways to set up a user management system and that's why we utilize it.
- Storing the data securely on the cloud ensures that no hardware failure whatsoever can cause the erasure of data, especially since this data is so vital to the lives of people all around the world
- The NFC Card Integration gives an additional incentive for more hospitals to join the centralized network, and this is why we believe that we will easily be able to get many hospitals on board with the concept
- Finally, we have the ability to scale very quickly due to our use of MongoDB JSON and Schemas, one of the most popular ways of storing data.

CONCLUSIONS

- Allows Patients to feel more secure about their medical conditions and the ability of doctors to diagnose and treat on-the-spot
- Allows Doctors to easily enter, modify, and retrieve data from the system
- Reduces overall costs of medical infrastructure and technology
- Easily integrated into existing systems, with added security and authentication

FUTURE WORK

- There is a total of 2314 Exabytes (2.314×10^{12} Gigabytes) of medical data in the United States. Our platform is only designed to accommodate 32 Terabytes per hospital due to the limitations of MongoDB itself. In order to combat this issue, we can use compression algorithms in order to take down this number into a more manageable number
- We can also use Machine Learning to optimize the most current compression algorithms and allow us to hold all of the country's data on the cloud. This would require the engineering of something like a Neural Network or a Genetic Algorithm. Machine learning also allows us to train a model to automatically migrate existing databases from hospitals into the cmyHealth platform.
- Once documentation for the API is written, other medical IT professionals can utilize it and actually implement elements of our application in real-life hospitals through the open-source software

ACKNOWLEDGEMENTS

The successful completion of this project was only possible with the support and encouragement of the following people:

- Mr. Crawford
- The Terra developers
- The Auth0 developers

REFERENCES

[1] Evans, M., & Safdar, K. (2020, March 18). Hospitals Facing Coronavirus Are Running Out of Masks, Other Key Equipment. Retrieved from https://www.wsj.com/articles/hospitals-facing-coronavirus-are-running-out-of-masks-other-key-equipment-11584525604
[2] Mayo Clinic's Epic EHR go-live: A look at major milestones leading up to big switch. (2018, May 1). Retrieved from https://www.healthcareitnews.com/news/mayo-clinics-epic-ehr-go-live-look-major-milestones-leading-big-switch
[3] Health Information Privacy Law and Policy. (2018, September 19). Retrieved from https://www.healthit.gov/topic/health-information-privacy-law-and-policy.
[4] Carter, A. J. E., Davis, K. A., Evans, L. V., & Cone, D. C. (2009). Information loss in emergency medical services handover of trauma patients. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/19499462
[5] van Ooijen, P. M. A., Guignot, J., Mevel, G., & Oudkerk, M. (2005, September). Incorporating out-patient data from CD-R into the local PACS using DICOM worklist features. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3046716/