Capstone Final Review

Decentralized Healthcare Network Platform

Team Members:
Shankha Shubhra Sarkar - (18BCE2453)
Anindya Sen - (18BCE2382)

Aim

The exponential growth in health data from a variety of sources, such as electronic medical records and image databases, makes it difficult to integrate information for optimized decision-making that meets the highest possible standards of healthcare. Patients need to collect all the services and arrange all the decision-making and communications themselves.

Our primary AIM is to create A digital platform that assembles and structures a wide variety of healthcare services under a single user-friendly platform that has the potential to remedy this situation.

Objective

Primary objective of this project is to make healthcare easily accessible. In order to get there, we need to solve couple of secondary objectives. Such as –

- User can get hospital services, ambulance services, pharmaceuticals services and all other healthcare related stuff through a single platform.
- This platform needs to be accessible through app or website from anywhere of the world.
- Platform need to be very secure as medical records require utmost privacy for anyone.
- User need to be able to share or give certain access permission or revoke to any other parties as they please.
- Hospitals need to have their own management services maintained and easily maintainable including their internal users with their own database access and control.
- Upon connecting to the services all service holders should get their own data privacy including regular backups and security.

Motivation

New technologies, including artificial intelligence, big data analysis, mobile applications and devices, and cloud storage, among others, have the power to significantly contribute to improving any health system. However, for many health service users, the majority of hospitals are far from being able to offer these technological benefits with what they have at present.

On the other hand, various latest technologies for similar purposes are being used in other industries such as banks, crypto currencies, online taxi services, online shopping & courier services and so on. By utilizing these techs and modify their existing algorithms to match our objectives of this project we can redesign the way people access healthcare services.

Literature Survey

Paper	Review of the Paper
Decentralized secure storage of medical records using Blockchain and IPFS: A comparative analysis with future directions (2021)	This study evaluated IPFS and Blockchain-based healthcare safe storage. This examination of existing solutions will aid IPFS and Blockchain research. In order to grow Consortium Blockchain, few scholars have established a means for independent enterprises to join the Network. No insider threats or access control. Decentralizing storage is difficult. A closer look finds hints of centralization. HTTP and DNS must be changed for true decentralization. IPFS and IPNS (Inter Planetary Name System) may revolutionize the paradigm.
Inter-Planetary File System Enabled Blockchain Solution For Securing Healthcare Records (2020)	This article proposes a blockchain-based architecture for the secure sharing of patient PHRs across healthcare providers. It also leverages IPFS for faster PHR retrieval. They built an Ethereum-based system. They used an index file with a SHA-256 IPFS hash to store reports. Their technology employs blockchain-based compute nodes to store client-centric patient health data. JMeter analyzes a participant's data read/write access and database retrieval. Data is stored in IPFS hash for faster retrieval, with numerous copies to eliminate a single point of failure.

Literature Survey

Paper	Review of the Paper
Blockchain for Emergency Vehicle Routing in Healthcare Services: An Integrated Secure and Trustworthy System (2021)	It uses the Ethereum Blockchain and Open-Source Routing Machine to route emergency vehicles (OSRM). The OSRM provides the most dependable routes. In addition, the strategy reduces emergency vehicle journey time. First, the dispersed network architecture authenticates the device. The DAPP helps the OSRM construct less crowded routes. The new route is the shortest of all congestion-free paths
Blockchain and Smart Contracts for Support the Interaction between the Actors in the Regional Innovation System (2018)	The system includes a distributed register of transactions with digital copies of intellectual assets. It keeps track of things like innovation item registration, author payment, and digital passport access. Then smart contracts. They enable multitasking in an innovative system. We anticipate smart contracts for various activities, transit services for quick information exchange, and a new state monitoring and control system.

Literature Survey

Paper	Review of the Paper
Securing Pharmaceutical Supply Chain using Blockchain Technology (2021)	Data interchange, storage, transparency, and traceability are assured by the hyper ledger fabric technology. Smart contracts governed sender-receiver interactions on Ethereum. Using blockchain for tracking and monitoring prevents counterfeit pharmaceuticals from entering the supply chain and reaching consumers. To generate a QR code for each product, an ID was assigned. An affordable and workable solution was identified. Consumer feedback helped assess and rank supply chain participants.
Chatbot for Healthcare System Using Artificial Intelligence (2020)	The chatbot program is meant to provide rapid replies. It relieves the answer supplier by providing the solution directly to the user. The user may save time by calling doctors or experts. A user query was analyzed using N-grams and TF-IDF. Weigh each phrase to achieve the best answer. The online interface allows users to submit inquiries. The application now protects users and characters when answering enquiries.

Existing System Gap

- Centralized System

Almost all of the existing services that trying to solve the similar issues are centralized. This causes large infrastructure, costly, inefficient system.

- Vulnerable to Cyber Attack

As majority of the hospitals doesn't have cybersecurity specialists, they prone to attain a single time installation system. This leaves old systems open to exposed vulnerabilities.

- Single Point of Failure

Storing all data in a single server with traditional onsite backups are dangerously vulnerable to single point of failure. Any sort of damage can cause permanent data loss or data leak.

- Privacy Concern

Current centralized system doesn't ensure privacy as they own the user data and experiment as they want.

- Compatibility with Existing or Multiple Systems

Existing systems tends to make a single types of service which may not be compatible with existing infrastructure or not even utilize the existing systems properly.

Proposed System

- Decentralized System

Data will be stored in a decentralized our very own INFS with will completely use the existing systems without requiring to install any new large infrastructure.

- Secure Storage

The whole network will be secured by the state-of-the-art blockchain based security. This will prevent any kind of data integrity related issues.

- Automated Software Management

Hospitals along with other nodes will get over the air software updates and management services like OS.

- Secure Automated Backups

IHFS will provide regular RAID backups as well as offsite backups to prevent any permanent data loss.

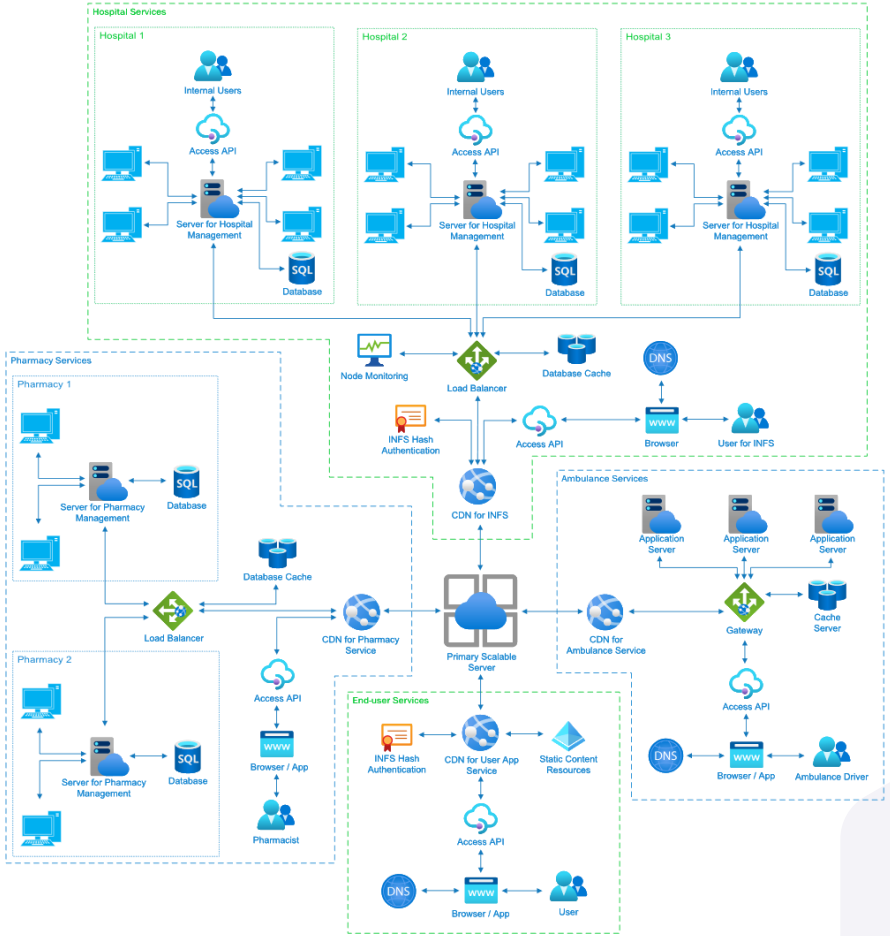
- Owner with full control of the data

Owners of the medical records will have full control over the access, modification or monitoring.

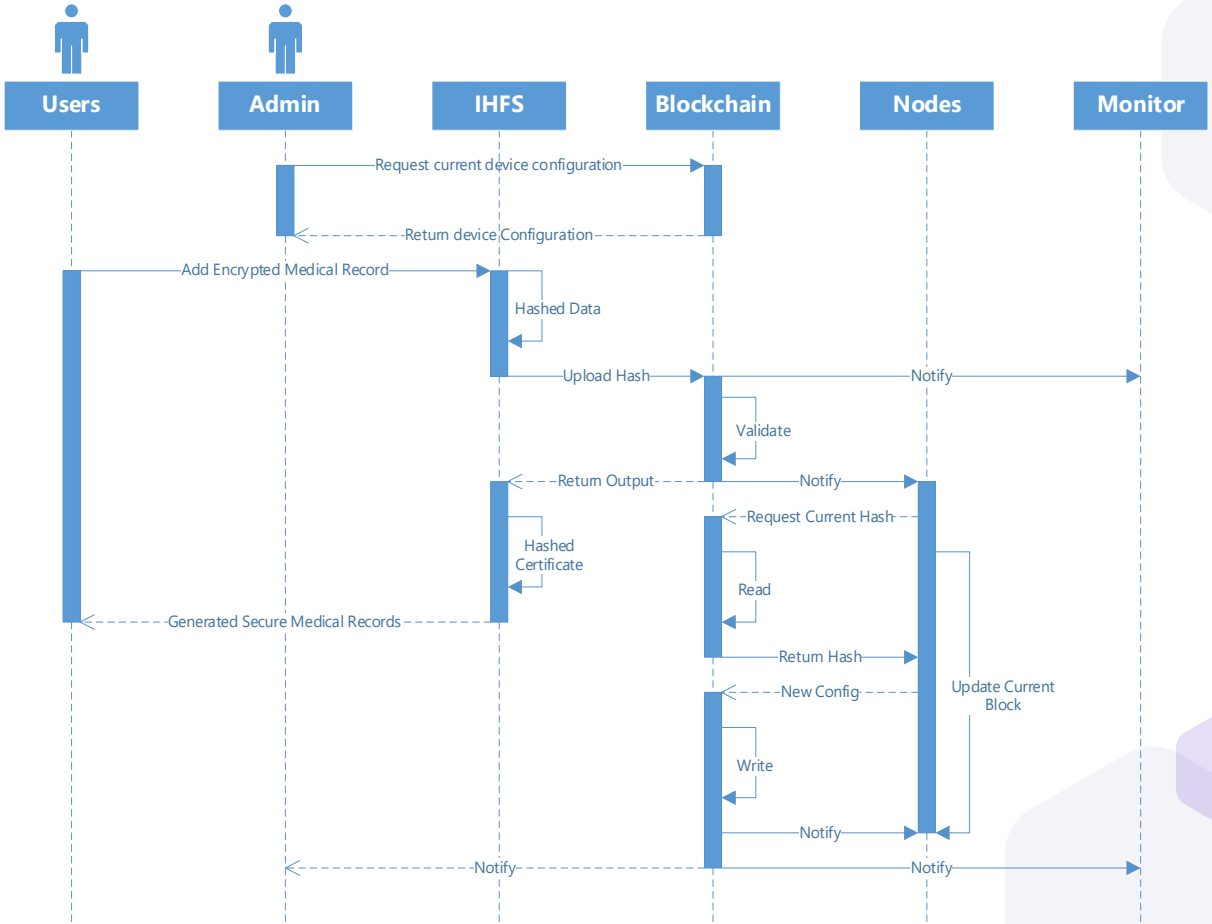
- Utilizing Existing Services Compatibility

As this service will include all services under single roof, whole healthcare departments will be in perfect sync.

ArchitectureProposed System Diagram



UML Diagram of the Proposed System



Proposed System Analysis & Design

Functional Requirements

- ***Product perspective :*** Distributed servers, Fast network connectivity, Proper Status Monitoring of the system.
- ***Product features :*** Fast, Convenient, and Low maintenance Distributed Database, Easy but Secure Access.
- ***User characteristics :*** Little to no training for Users familiar with Smart Phones or Internet Browsers. App will provide hands on tutorial on first launch with optional repetition.
- ***Assumption & Dependencies :*** As it is a platform-independent distributed system, just one terminal is required to access the service. Hospitals will need some setup, but users will have plug and play experience.
- ***Domain Requirements :*** This project development in accordance with openEHR (Open Electronic Health Record)
- ***User Requirements :*** As mostly users will be patients, a fully customizable UI with human machine interface for disabled people such as coded sound/vibration signals with proper SOS service.

Proposed System Analysis & Design

Non-Functional Requirements

- ***Efficacy :*** The system provides acknowledgment in just one second once the 'patient's information is checked. The system needs to support at least 1000 people at once. The user interface acknowledges within five seconds. The system offers the efficiency for data backup.
- ***Reliability :*** As the system is completely distributed through blockchain, it's available all the time.
- ***Portability :*** A single handheld terminal such as smartphone is enough to operate all the emergency services along with the transactions and monitoring.
- ***Usability :*** Any modifications like insert, delete, update, etc. for the record can be synchronized quickly and executed only by the proper authenticated user. Every kind of users from separate modules have their own interconnected portals for better usability.

Proposed System Analysis & Design

Organizational Requirements

- ***Implementation Requirements:*** Primary scalable server on the cloud needs to be deployed for the start of the service and monitoring. Once enough hospitals are subscribed for the IHFS, the system will become selfsustained and all the databased will be maintained by the distributed file system. Hospitals will require to setup an intranet among the computers situated within & rest of the service holders will require to setup apps in their computer or phones for connectivity to the service.
- ***Engineering Standard Requirements :*** We will use ISO 18308:2011 (Health informatics - Requirements for an electronic health record architecture) for this project. ISO 18308:2011 outlines the standards for an EHR architecture, a system that processes, maintains, and transmits EHR data. These EHRs must be clinically valid and trustworthy, ethically sound, satisfy current legal requirements, promote excellent clinical practice, and permit data analysis for a variety of objectives.

Modules to be Implemented

- **Patient Service Network**
  All of the medical services that a patient requires are integrated into the overall infrastructure. Declaring an emergency, looking for a hospital, calling an ambulance, purchasing medicine, paying costs, maintaining medical records, and so on.
- **Hospital Internal Network**
  A complete hospital management system will be included in this module.
- **Pharmaceutical Services**
  This module will create a separate network between all the pharmacy and their stocks. They can provide online shopping service as well as hospital supplies.
- **Ambulance Services**
  Registered ambulances will have own network & will be able to answer on patient's emergency declarations.
- **Data storage and maintenance**
  Our state-of-the-art decentralized file management system will be implemented in this module.

System Requirements

Hardware Requirements

- ***End User H/W :*** Any kind of internet connected PC or Mobile with Browser or IHFS App.
- ***Subscribed Hospitals H/W :*** IHFS connected Central Server system, Individual desk computers and Stuff computers in the hospital connected to intranet of the central server.
- ***Subscribed Pharmaceuticals :*** IHFS connected PC for small shop, IHFS connected central terminal with intranet among the shop's other PCs for larger pharmacy shops.
- ***Ambulance Drivers H/W :*** Internet connected Ambulance Service Providing App installed Smart Phone.

System Requirements

Software Requirements

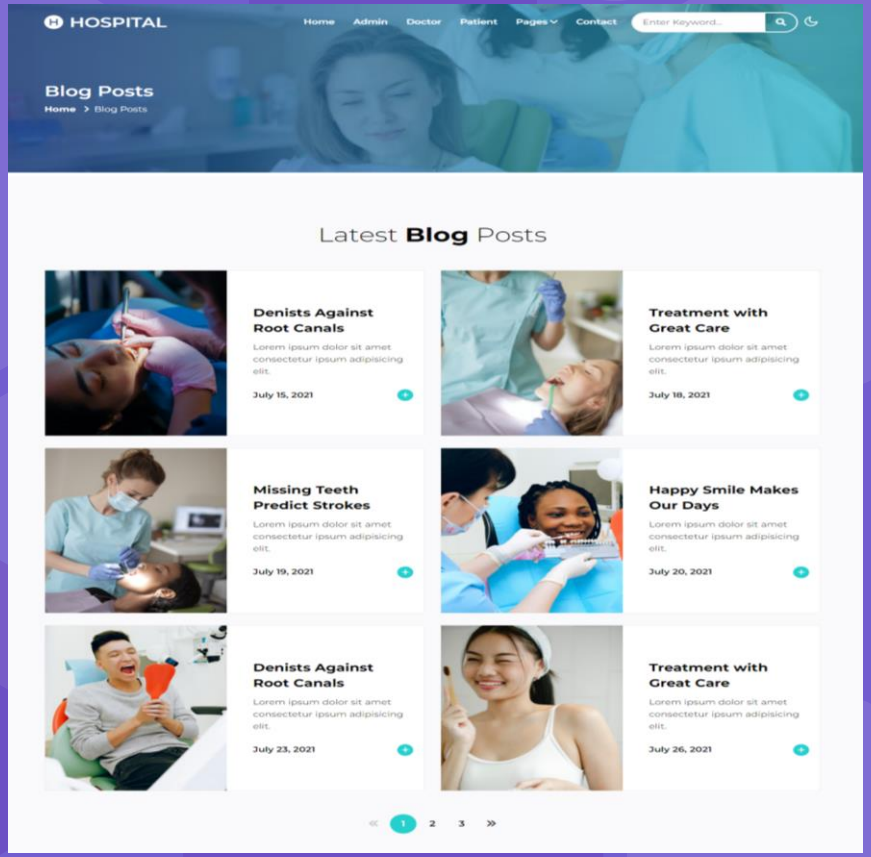
- ***Programming Languages:***
  - JavaScript, Python, Java, Solidity, HTML, CSS
- ***Technologies:***
  - REST API, Django, Microsoft Azure Cloud Engine, Ethereum VM, IPFS
- ***Software & IDE:***
  - VS Code, Cloud SSH, Remix, IntelliJ IDEA

Project Timeline Chart

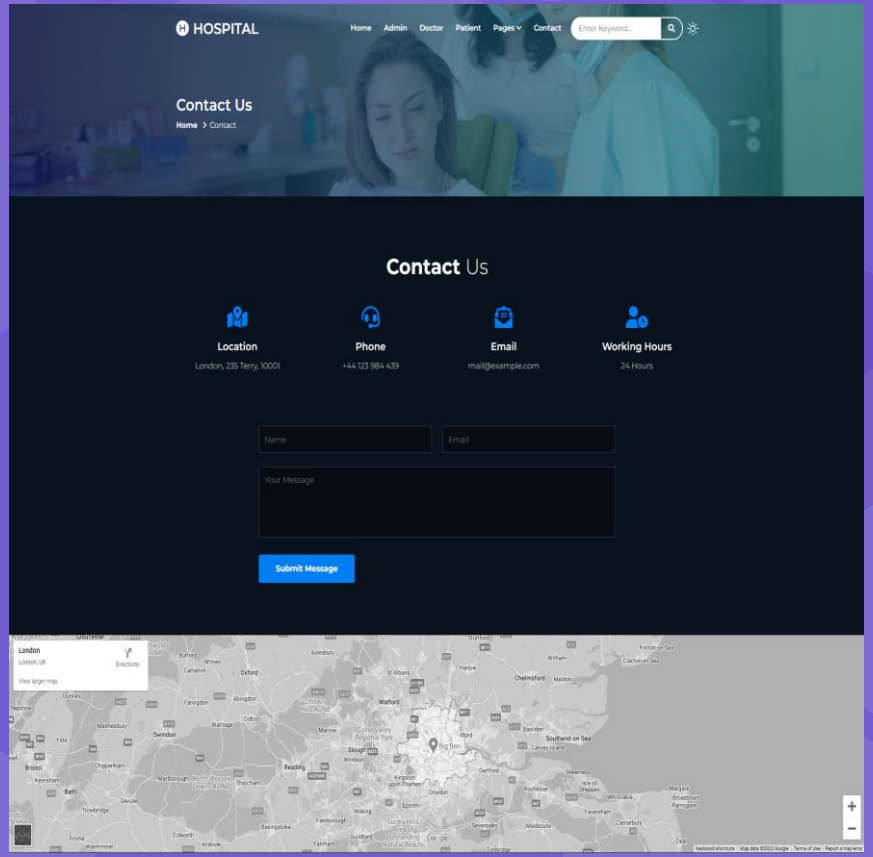
	Task Mode	Task Name	Work	Duration	Start	Finish	Predecessors	Resource Names
1		Start	0 hrs	0 days	06 January 2022 8:00 AM	06 January 2022 8:00 AM		
2		**Requirement Analysis**	**72 hrs**	**4.75 days**	**06 January 2022 8:00 AM**	**12 January 2022 3:00 PM**		
3		Market Research	16 hrs	1 day	06 January 2022 8:00 AM	06 January 2022 5:00 PM	1	Anindya Sen,Shankha Shubhra Sarkar
4		Paper Research	48 hrs	3 days	07 January 2022 8:00 AM	11 January 2022 5:00 PM	3	Anindya Sen,Shankha Shubhra Sarkar
5		Brain Storming	4 hrs	0.25 days	12 January 2022 8:00 AM	12 January 2022 10:00 AM	4	Anindya Sen,Shankha Shubhra Sarkar
6		Feature List	4 hrs	0.5 days	12 January 2022 10:00 AM	12 January 2022 3:00 PM	3,5	Shankha Shubhra Sarkar
7		Requirement Complete	0 hrs	0 days	12 January 2022 3:00 PM	12 January 2022 3:00 PM	3,4,5,6	
8		**Project Process Diagram**	**4 hrs**	**0.5 days**	**12 January 2022 3:00 PM**	**13 January 2022 10:00 AM**		
9		Feature Prioritaization	1 hr	0.13 days	12 January 2022 3:00 PM	12 January 2022 4:00 PM	7	Anindya Sen
10		Architecture	3 hrs	0.38 days	12 January 2022 4:00 PM	13 January 2022 10:00 AM	9	Shankha Shubhra Sarkar
11		Diagram Complete	0 hrs	0 days	13 January 2022 10:00 AM	13 January 2022 10:00 AM	9,10	
12		**Design**	**40 hrs**	**4.5 days**	**13 January 2022 10:00 AM**	**19 January 2022 3:00 PM**		
13		Interface Desgin	24 hrs	3 days	13 January 2022 10:00 AM	18 January 2022 10:00 AM	11	Shankha Shubhra Sarkar
14		Software Design	8 hrs	1 day	18 January 2022 10:00 AM	19 January 2022 10:00 AM	13	Anindya Sen
15		Design Specification	8 hrs	0.5 days	19 January 2022 10:00 AM	19 January 2022 3:00 PM	14	Anindya Sen,Shankha Shubhra Sarkar
16		Design Complete	0 hrs	0 days	19 January 2022 3:00 PM	19 January 2022 3:00 PM	13,14,15	
17		**Development**	**536 hrs**	**37 days**	**19 January 2022 3:00 PM**	**11 March 2022 3:00 PM**		
18		Develop Frontend	80 hrs	10 days	19 January 2022 3:00 PM	02 February 2022 3:00 PM	16	Anindya Sen
19		Develop Server Engine	80 hrs	10 days	19 January 2022 3:00 PM	02 February 2022 3:00 PM	16	Shankha Shubhra Sarkar
20		Develop Backend	320 hrs	20 days	02 February 2022 3:00 PM	02 March 2022 3:00 PM	16,19	Anindya Sen,Shankha Shubhra Sarkar
21		Integrate System Modules	32 hrs	4 days	02 March 2022 3:00 PM	08 March 2022 3:00 PM	20	Shankha Shubhra Sarkar
22		Perform Initial Testing	24 hrs	3 days	08 March 2022 3:00 PM	11 March 2022 3:00 PM	21	Anindya Sen
23		Development Complete	0 hrs	0 days	11 March 2022 3:00 PM	11 March 2022 3:00 PM	18,19,21,22,20	
24		**Testing**	**120 hrs**	**9 days**	**11 March 2022 3:00 PM**	**24 March 2022 3:00 PM**		
25		Perform System Testing	16 hrs	2 days	11 March 2022 3:00 PM	15 March 2022 3:00 PM	23	Anindya Sen
26		Document Issues Found	8 hrs	1 day	15 March 2022 3:00 PM	16 March 2022 3:00 PM	25	Shankha Shubhra Sarkar
27		Correct Issues Found	96 hrs	6 days	16 March 2022 3:00 PM	24 March 2022 3:00 PM	26	Anindya Sen,Shankha Shubhra Sarkar
28		Testing Complete	0 hrs	0 days	24 March 2022 3:00 PM	24 March 2022 3:00 PM	25,26,27	
29		**Deployment**	**58 hrs**	**4.25 days**	**24 March 2022 3:00 PM**	**30 March 2022 5:00 PM**		
30		Manage Resources	8 hrs	1 day	24 March 2022 3:00 PM	25 March 2022 3:00 PM	28	Shankha Shubhra Sarkar
31		Onsite Installation	24 hrs	3 days	25 March 2022 3:00 PM	30 March 2022 3:00 PM	30	Anindya Sen
32		Setup CDN	24 hrs	3 days	25 March 2022 3:00 PM	30 March 2022 3:00 PM	30	Shankha Shubhra Sarkar
33		Import Live Data	2 hrs	0.25 days	30 March 2022 3:00 PM	30 March 2022 5:00 PM	31,32	Anindya Sen
34		Deployment Complete	0 hrs	0 days	30 March 2022 5:00 PM	30 March 2022 5:00 PM	30,31,32,33	
35		Project Outcome	128 hrs	8 days	31 March 2022 8:00 AM	11 April 2022 5:00 PM	34	Anindya Sen,Shankha Shubhra Sarkar
36		End	0 hrs	0 days	11 April 2022 5:00 PM	11 April 2022 5:00 PM	35	

Website Snapshot (Home page)

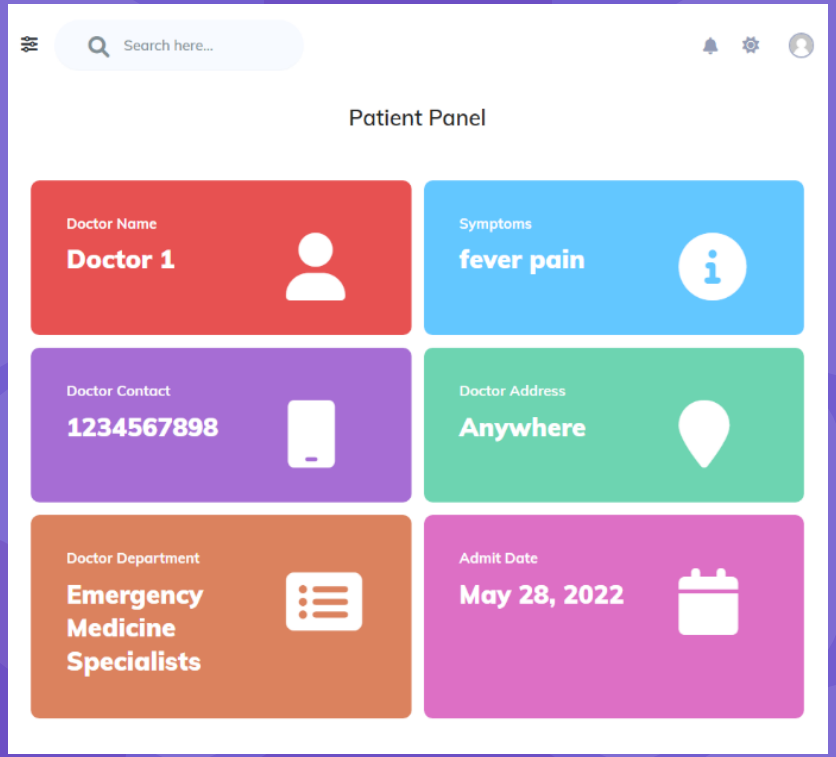
Demo healthy blogs for users



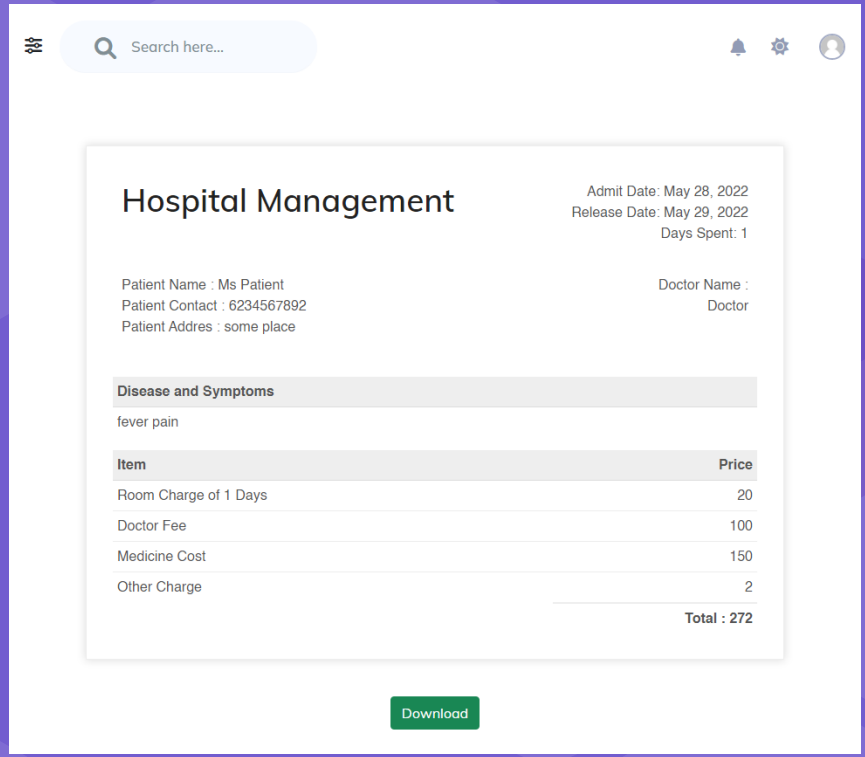
Contact us section with dark mode



Patient Module

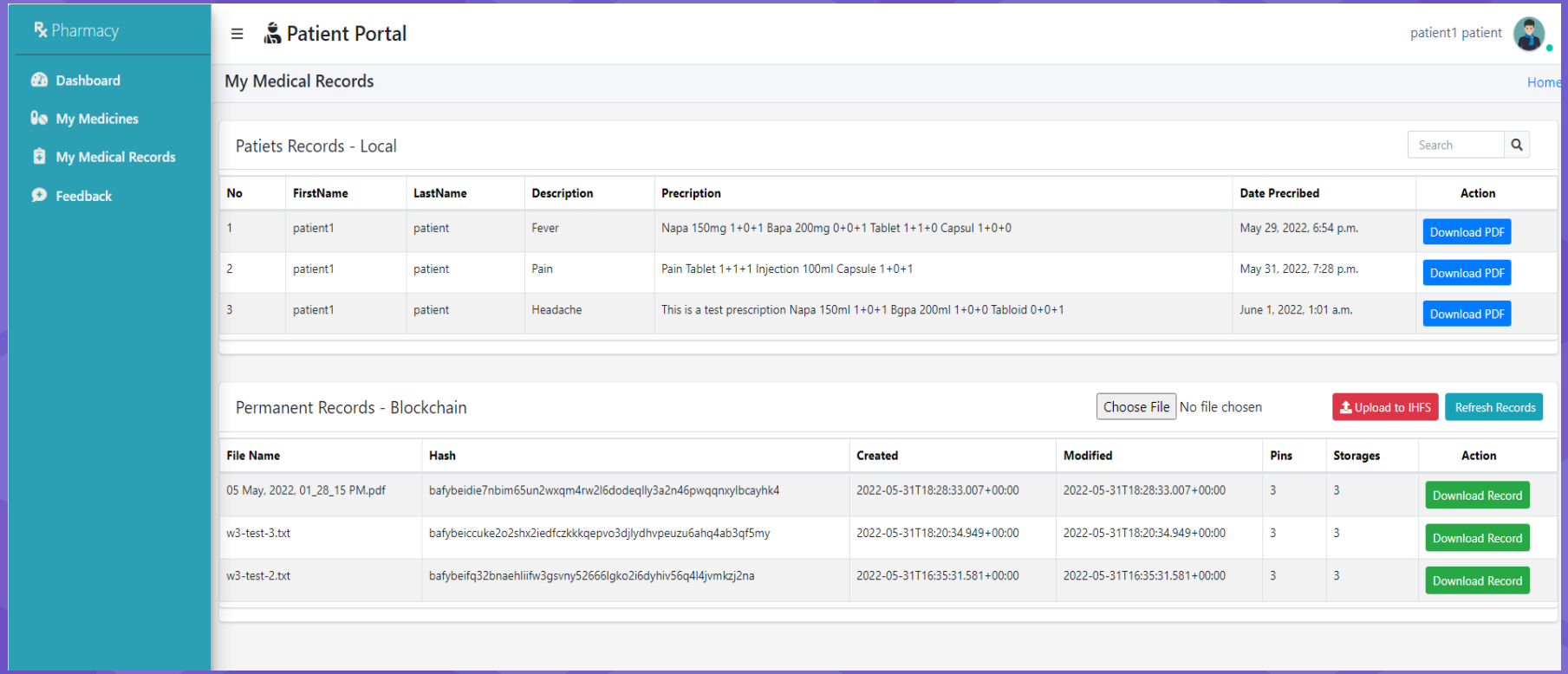


Patient Dashboard



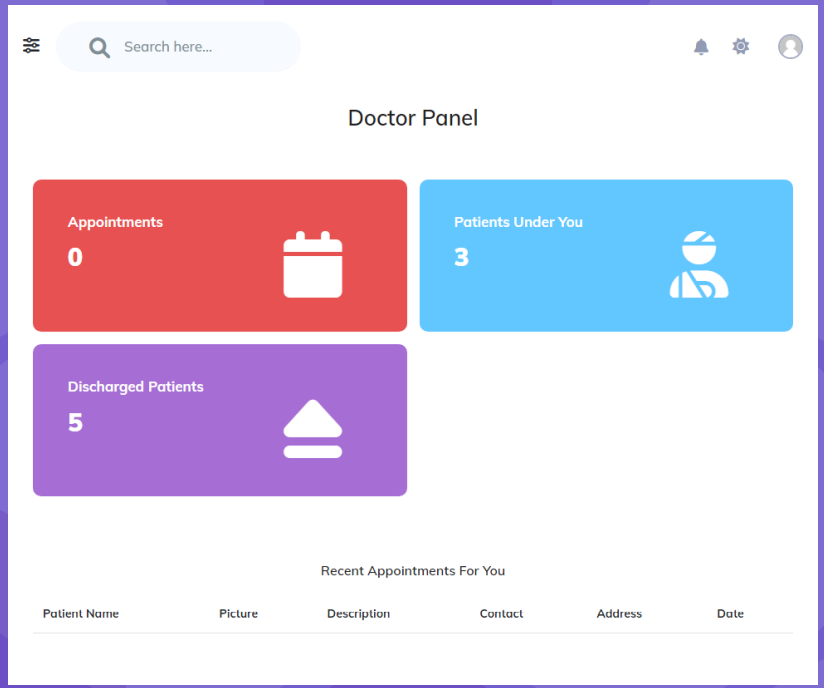
Patient Discharge Receipt

Patient Module

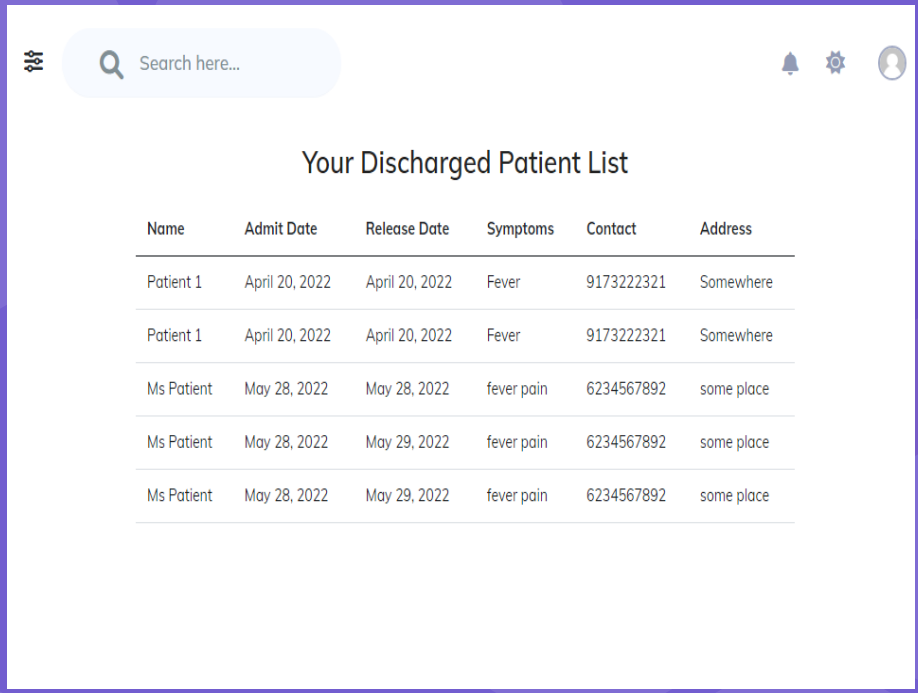


Patient Medical Records Stored Locally (Top Table)
Permanently on Blockchain (Bottom Table)

Doctor Module

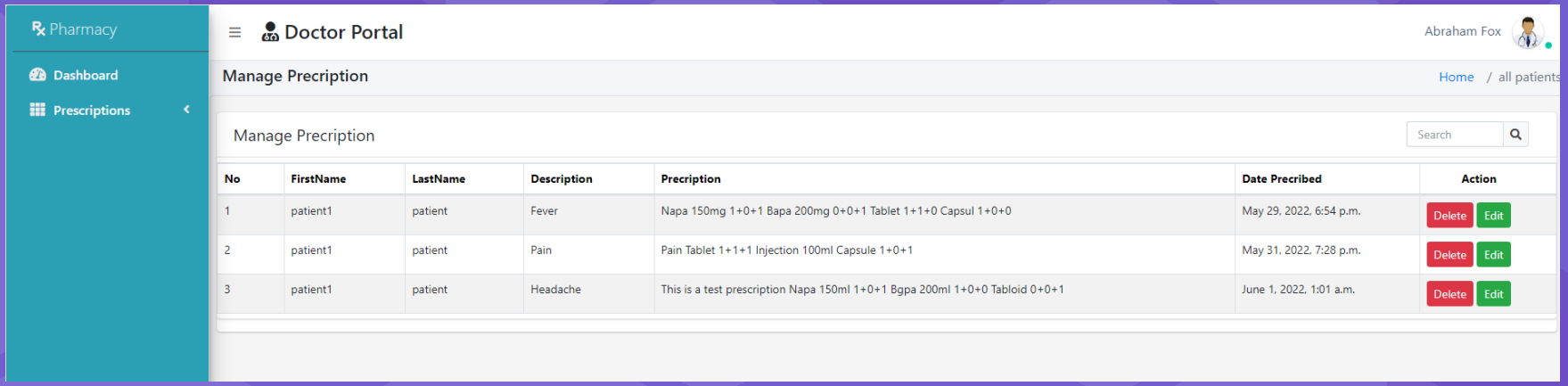


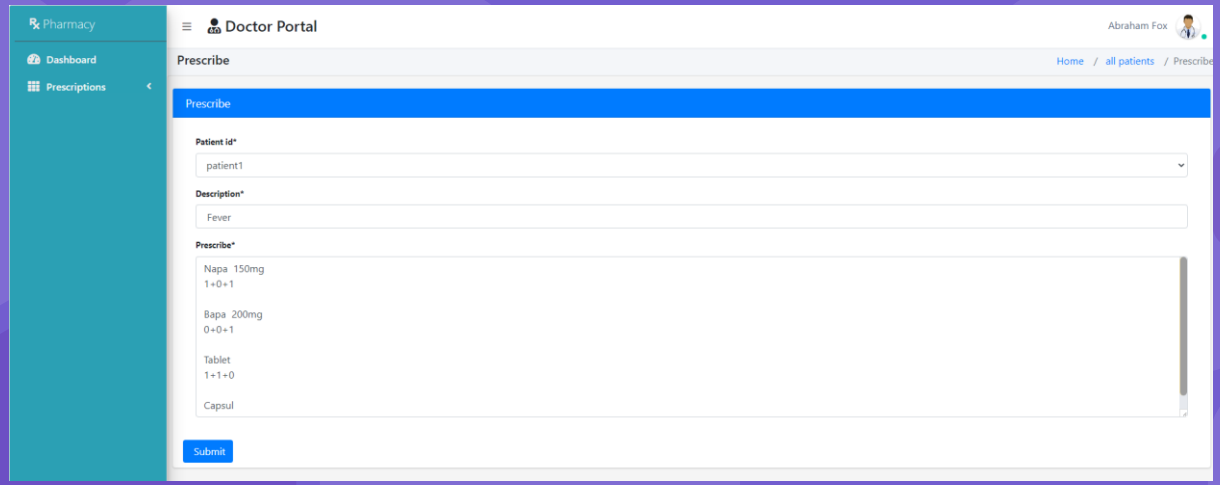
Doctor Dashboard



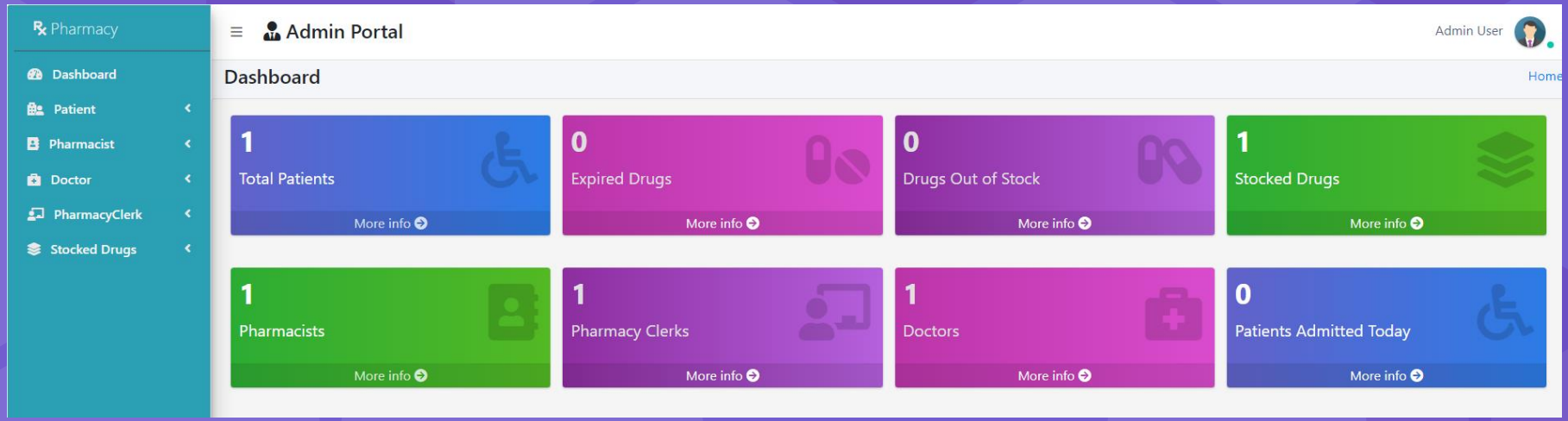
Patient List

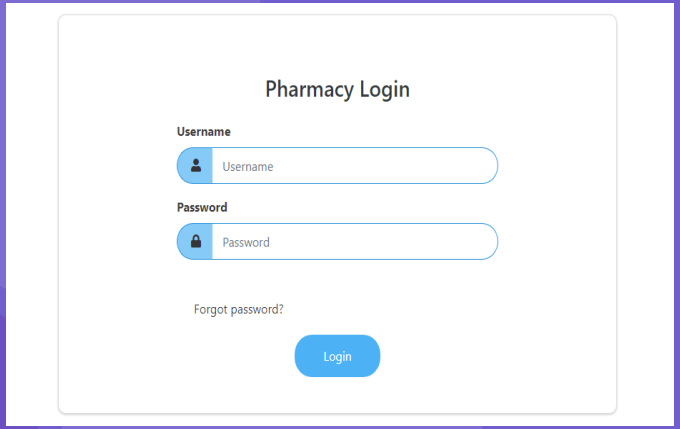
Patients Prescriptions History and Form



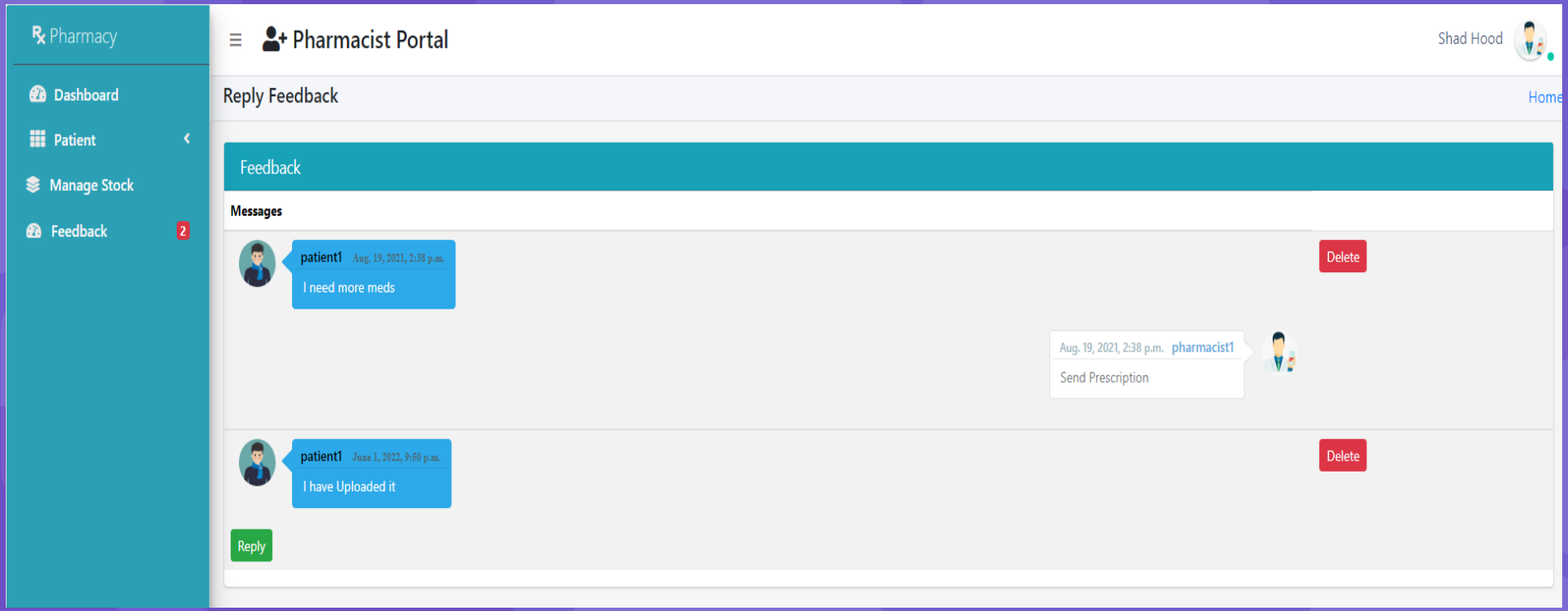


Pharmacy Admin Dashboard and Login

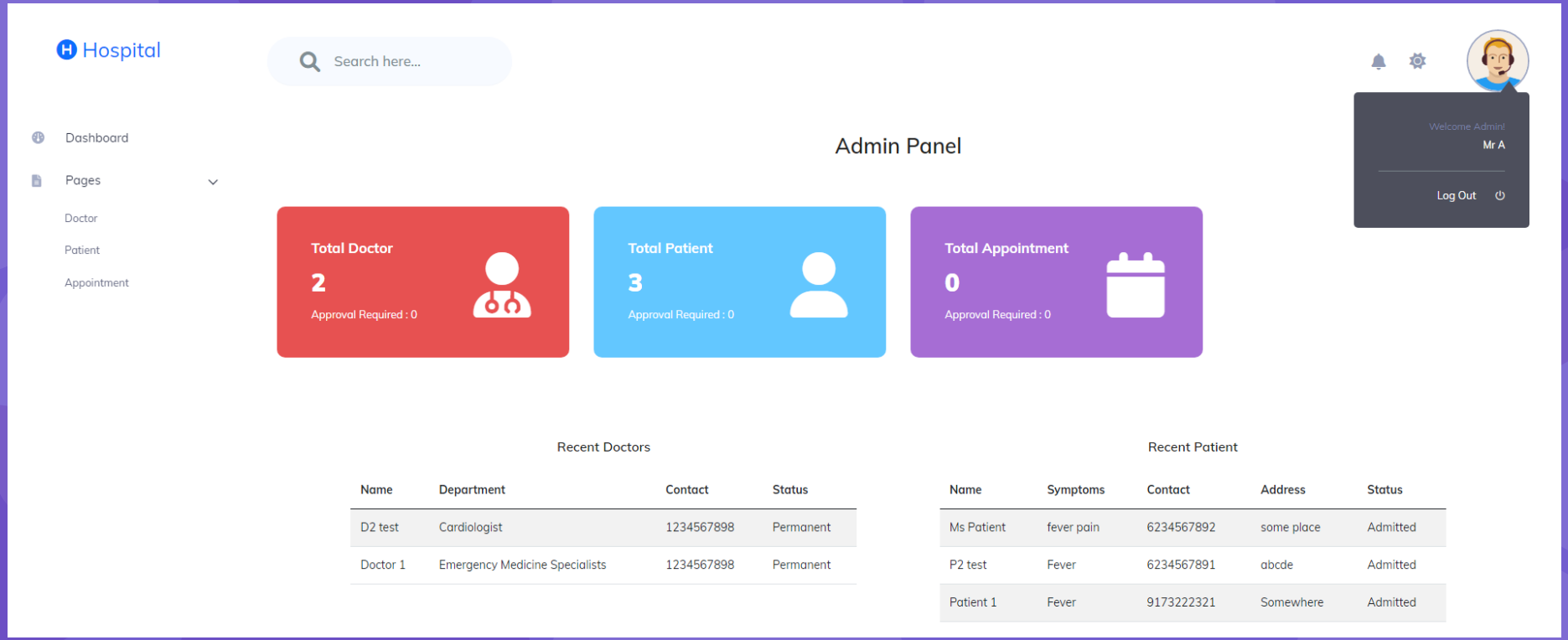




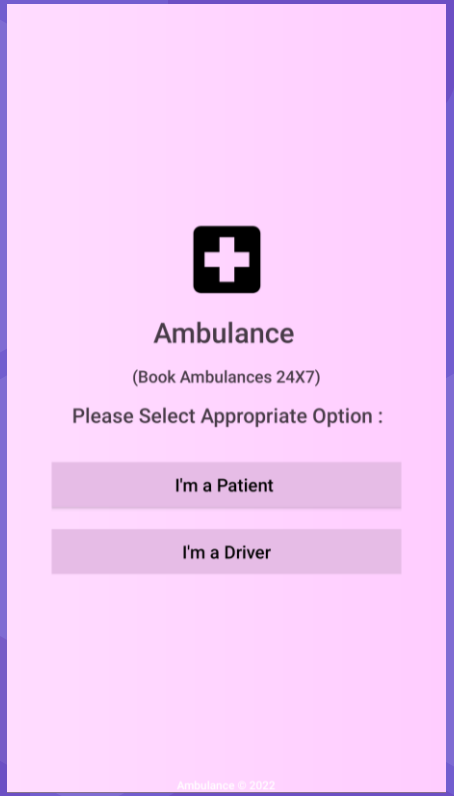
Portal for chatting with Pharmacists



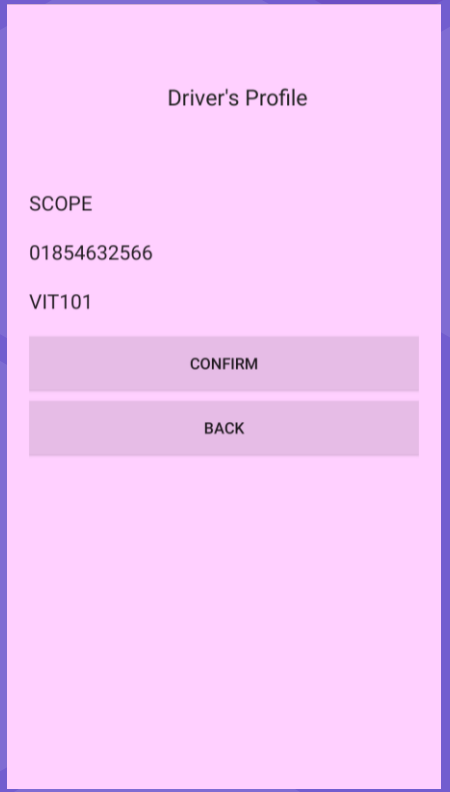
Hospital Admin Dashboard



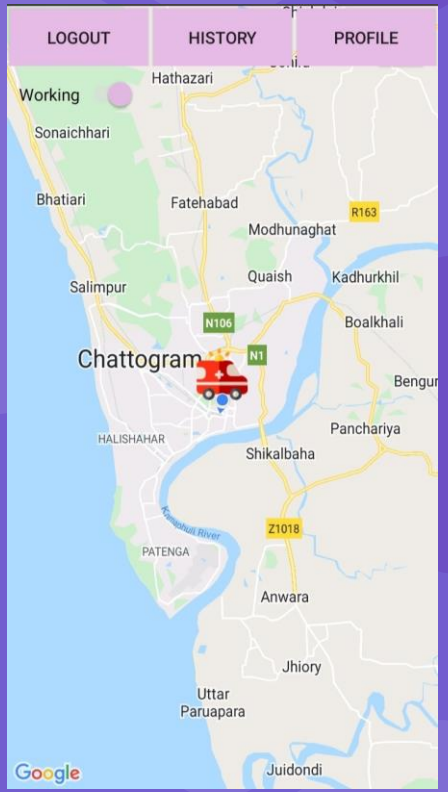
Ambulance App Snapshot



Login Page

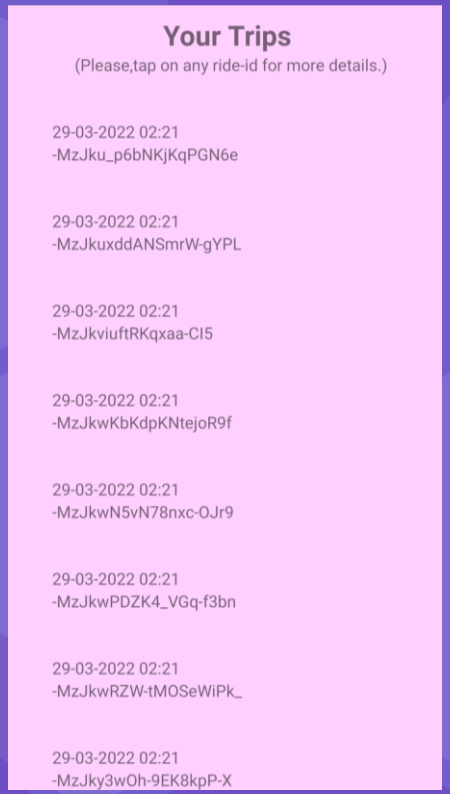


Driver's Profile Page



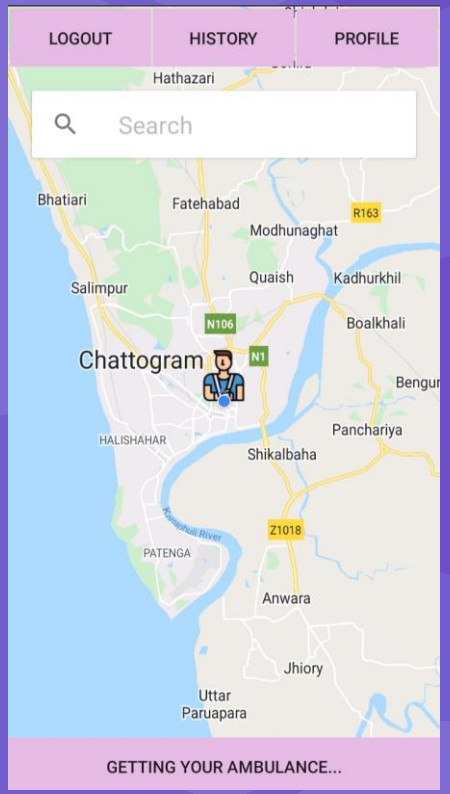
Driver Home Page

Ambulance App Snapshot



Trip History Page

Patient's Profile Page



Patient Home Page

IHFS Snapshot

index.js | globals.css

pages > index.js > Home > handleSubmit

```
1  import React, { useState, useReducer } from 'react'
2  import { Web3Storage } from 'web3.storage'
3
4  export default function Home () {
5    const [messages, showMessage] = useReducer((msgs, m) => msgs.concat(m), [])
6    // const [token, setToken] = useState('')
7    const token = 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJkaWQ6ZXRocjoweDhmNjBCODIzZThEMzhiM0I
8    const [files, setFiles] = useState([])
9
10   async function handleSubmit (event) {
11     // don't reload the page!
12     event.preventDefault()
13
14     const client = new Web3Storage({ token })
15
16     showMessage('chunking and hashing the files (in your browser!) to calculate the Content ID')
17     const cid = await client.put(files, {
18       onRootCidReady: localCid => {
19         showMessage(`locally calculated Content ID: ${localCid} `)
20         showMessage('sending files to IHFS storage ')
21       },
22       onStoredChunk: bytes => showMessage(`sent ${bytes.toLocaleString()} bytes to IHFS storage`)
23     })
24     showLink(`https://dweb.link/ipfs/${cid}/${files[0].name}`)
25   }
26
27   function showLink (url) {
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
wait  - compiling...
event - compiled successfully in 12.2s (278 modules)
wait  - compiling / (client and server)...
event - compiled client and server successfully in 1745 ms (308 modules)
wait  - compiling...
event - compiled client and server successfully in 1570 ms (305 modules)
wait  - compiling / (client and server)...
event - compiled client and server successfully in 5.3s (308 modules)
wait  - compiling...
event - compiled client and server successfully in 1457 ms (305 modules)
wait  - compiling / (client and server)...
event - compiled client and server successfully in 1899 ms (308 modules)
```

Pick files to store

Choose files MyFile.pdf

Submit

Log:
waiting for form submission...
chunking and hashing the files (in your browser!) to calculate the Content ID
locally calculated Content ID:
bafybeifyiuha7b3cc7nu7joxp3nlhzma5prulbp3prhach2elu2r6vu2kq
sending files to IHFS storage
sent 200 bytes to IHFS storage
https://dweb.link/ipfs/bafybeifyiuha7b3cc7nu7joxp3nlhzma5prulbp3prhach2elu2r6vu2kq/MyFile.txt
chunking and hashing the files (in your browser!) to calculate the Content ID
locally calculated Content ID:
bafybeiezhbes4sjo5ghkzszzjn5aluspt3tyqha4mk5hemlcttjoqc575u
sending files to IHFS storage
sent 201 bytes to IHFS storage
https://dweb.link/ipfs/bafybeiezhbes4sjo5ghkzszzjn5aluspt3tyqha4mk5hemlcttjoqc575u/MyFile2.txt
chunking and hashing the files (in your browser!) to calculate the Content ID
locally calculated Content ID:
bafybeifxtot75tornk3yybwo3o5zq2ds4jceposwguuqyg2666yg3ljmoi
sending files to IHFS storage
sent 14,091 bytes to IHFS storage
https://dweb.link/ipfs/bafybeifxtot75tornk3yybwo3o5zq2ds4jceposwguuqyg2666yg3ljmoi/MyFile.pdf

Research Work

In practice, it's tough to execute a large decentralized project of this sort and maintain all important data as a project prototype for IHFS implementation and user database work. For the database, we utilized IPFS for prototyping, and for the ambulance service, we used a simulator to calculate shortest route and communicate real-time data with the hospital and other parties. Distance, speed, energy usage, and speed are tracked. We modelled our architecture using SUMO, NS-3, and OSM. We created osm.sumocfg in OSM to produce a trace file for NS-3 simulation. We ran SUMO on Dijkstra and A*. Then, for 50 minutes, we ran the trace files on NS-3 for eight nodes and acquired performance data, including distance, time, fuel, and emission metrics. Time-Fuel chart shows outcomes. During the trial, eight automobiles were despatched at various intervals, including one non-peak hour break to employ solo GPS and V2I communications. By comparing these findings to past attempts, we can observe that our performance was almost as efficient as other researchs, despite using three unique technologies to ascertain exact real-time position together with major navigation services' default navigation algorithms.

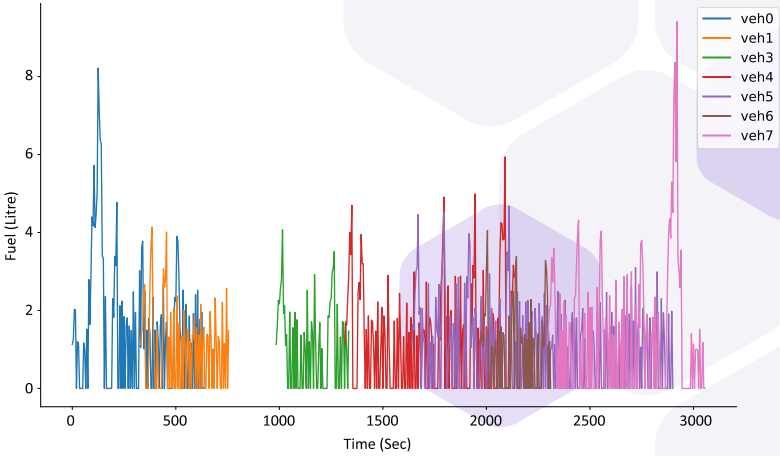


Figure: Simulation Result Plot of proposed network architecture ran on our Institution Map

Conclusion

- Using state-of-the-art blockchain technology, we have built a totally decentralized system in which data is saved in our proprietary file system called IHFS, which is entirely protected from data security and integrity concerns. This prevents any form of data integrity issues from occurring.

- We have developed a route planning system for ambulance services that is both efficient and quick. Once the whole service is up and running with a small number of subscriber hospitals, the database-related tasks will become entirely self-sustaining, and it will monitor all occurrences, including backups, on a continuous basis. Our secure pipeline access via the App provides customers with simple but very secure access to the network, as well as a straightforward method of granting or revoking access to other parties that are engaged in the transaction. It also makes use of the hospitals' current system architecture as well as the extremely popular smartphone infrastructure. This will help up finalizing the platform which will connect all the major healthcare services.

References

1. A. Rejeb, H. Treiblmaier, K. Rejeb and S. Zailani, "Blockchain research in healthcare: a bibliometric review and current research trends".
2. X. Du, B. Chen, M. Ma and Y. Zhang, "Research on the Application of Blockchain in Smart Healthcare: Constructing a Hierarchical Framework".
3. K. Fan, S. Wang, Y. Ren, H. Li and Y. Yang, "MedBlock: Efficient and Secure Medical Data Sharing Via Blockchain".
4. G. Yang, C. Li and K. Marstein, "A blockchain-based architecture for securing electronic health record systems".
5. F. Alam Khan, M. Asif, A. Ahmad, M. Alharbi and H. Aljuaid, "Blockchain technology, improvement suggestions, security challenges on smart grid and its application in healthcare for sustainable development".
6. P. Pandey and R. Litoriya, "Securing and authenticating healthcare records through blockchain technology".
7. L. Athota, V. Shukla, N. Pandey and A. Rana, "Chatbot for Healthcare System Using Artificial Intelligence".
8. U. Bharti, D. Bajaj, H. Batra, S. Lalit, S. Lalit and A. Gangwani, "Medbot: Conversational Artificial Intelligence Powered Chatbot for Delivering Tele-Health after COVID-19".

References

9. S. Kumar, A. Bharti and R. Amin, "Decentralized secure storage of medical records using Blockchain and IPFS : A comparative analysis with future directions".
10. R. Kumar and R. Tripathi, "A Secure and Distributed Framework for sharing COVID-19 patient Reports using Consortium Blockchain and IPFS".
11. R. Marangappanavar and M. Kiran, "Inter-Planetary File System Enabled Blockchain Solution For Securing Healthcare Records".
12. A. Azaria, A. Ekblaw, T. Vieira and A. Lippman, "MedRec: Using Blockchain for Medical Data Access and Permission Management".
13. R. Kaurav, R. Rout and S. Vemireddy, "Blockchain for Emergency Vehicle Routing in Healthcare Services: An Integrated Secure and Trustworthy System".
14. A. Finogeev, S. Vasin, L. Gamidullaeva and D. Parygin, "Blockchain and Smart Contracts for Support the Interaction between the Actors in the Regional Innovation System".
15. V. Lingayat, I. Pardikar, S. Yewalekar, S. Khachane and S. Pande, "Securing Pharmaceutical Supply Chain using Blockchain Technology".

Thank You!