

Apeksha Hospital Donor Engagement System

2023-24-100

Status Document 2

Punchihewa S.N

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Intelligent Donor-Driven Inventory System for Essential Items

2023-24-100

Status Document 2

Punchihewa S.N

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Declaration of The Candidate & Supervisor

I declare that this is my work, and this proposal does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Student ID	Signature
Punchihewa S. N	IT20665166	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor:

Date:

Signature of the Co-supervisor:

Date:

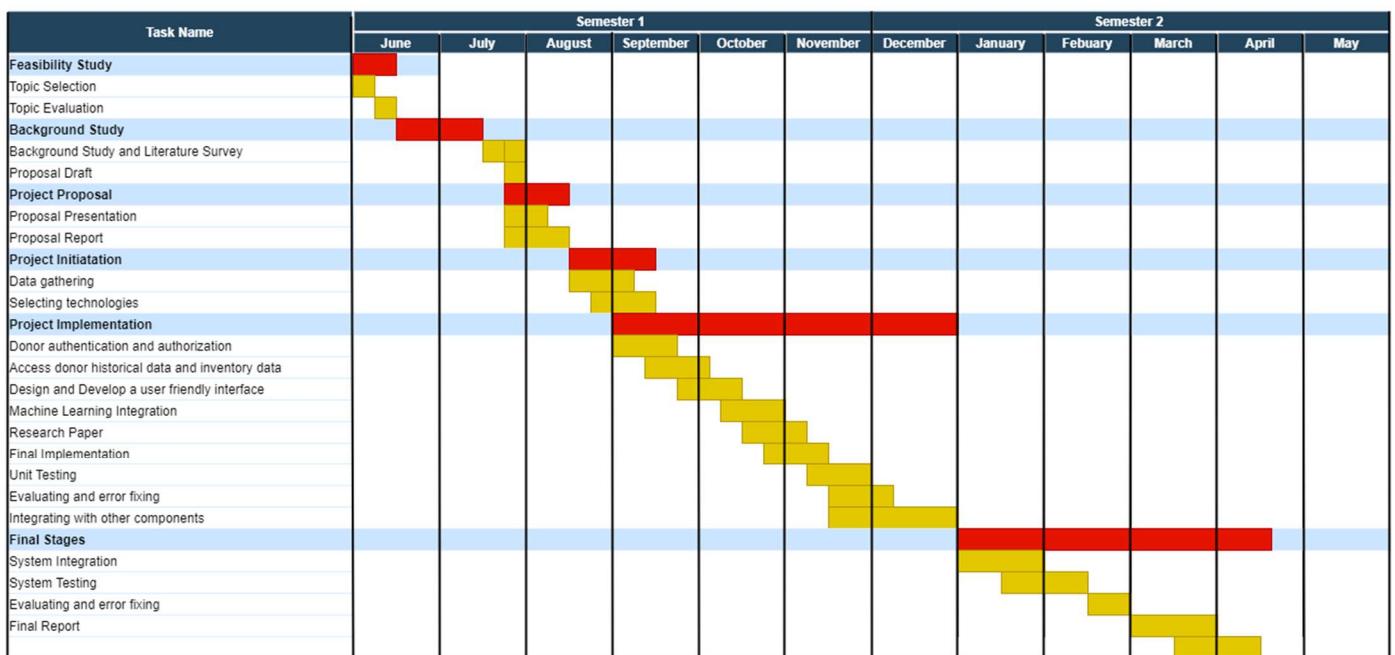
Table of Contents

1.	Introduction.....	IV
2.	Gantt Chart.....	V
3.	Work-Break-Down Chart.....	VI
4.	Research Paper.....	VII
5.	Screenshot of Meetings and Conversations	VIII
6.	Project Management Tools & Screenshot	XII
6.1	Overall Backlog	XII
6.2	Personal Backlog	XIII
7.	Trained Model Screenshot	XV
7.1	Decision Tree Regression	XV
8	Web Implementation	XVI
9	Version Control and Contribution	XXI

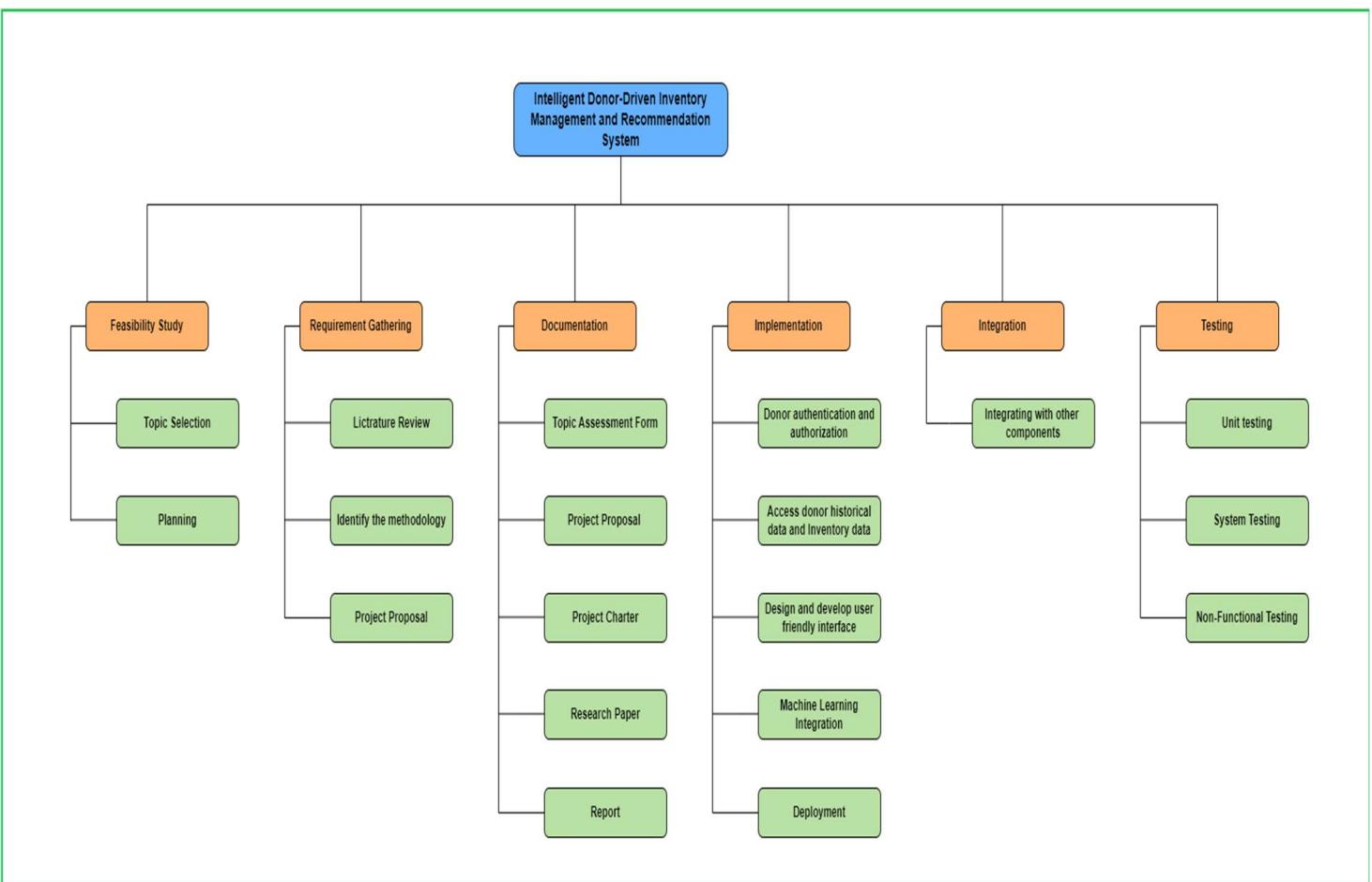
1. Introduction

Apeksha Hospital has the significant problem of enhancing donor involvement and enhancing the management of essential item inventory. To overcome this difficulty, this study proposes a new Intelligent Donor-Driven Inventory System for Essential Supplies. The main goal is to make use of machine learning algorithms to better understand donor habits, preferences, and giving patterns over time. As a result of this knowledge, the system will be in a position to skillfully suggest essential products to prospective donors, therefore stimulating their donations that are in line with the hospital's immediate demands. The ultimate goal of this research is to provide an engaging platform that integrates donor engagement with inventory control. Donors will be able to see the direct results of their donations and get personalized product recommendations from the system. By using these strategies, we want to increase donor pleasure and engagement while establishing a positive cycle of engagement. The research being conducted hopes to make the present of essential items more useful by concluding the gap between what donors want to do and what hospitals need. This collaborative method, which is made possible by machine learning and a donor-centered interface, not only improves the way resources are used but also makes the experience for donors as a whole better. Because of this, the results of this research are likely to make a big difference in how people give donations to Apeksha Hospital.

2. Gantt Chart



3. Work-Break-Down Chart

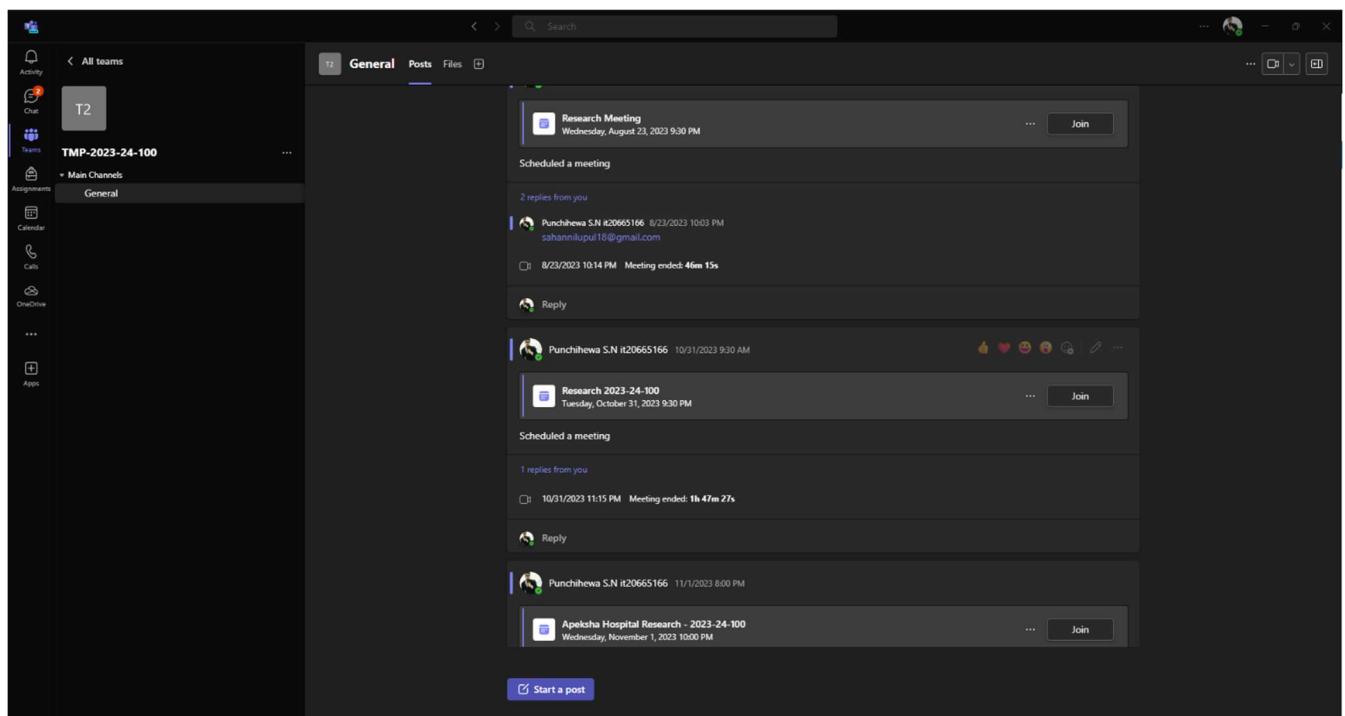
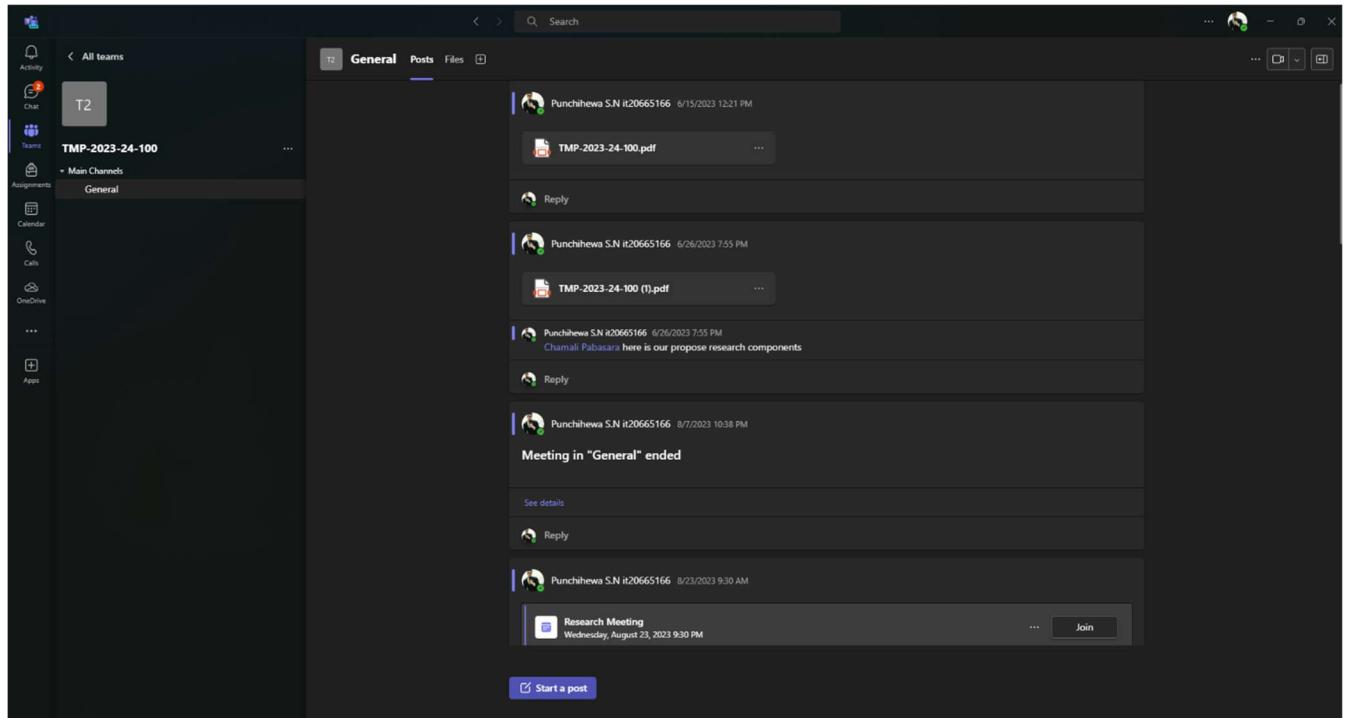


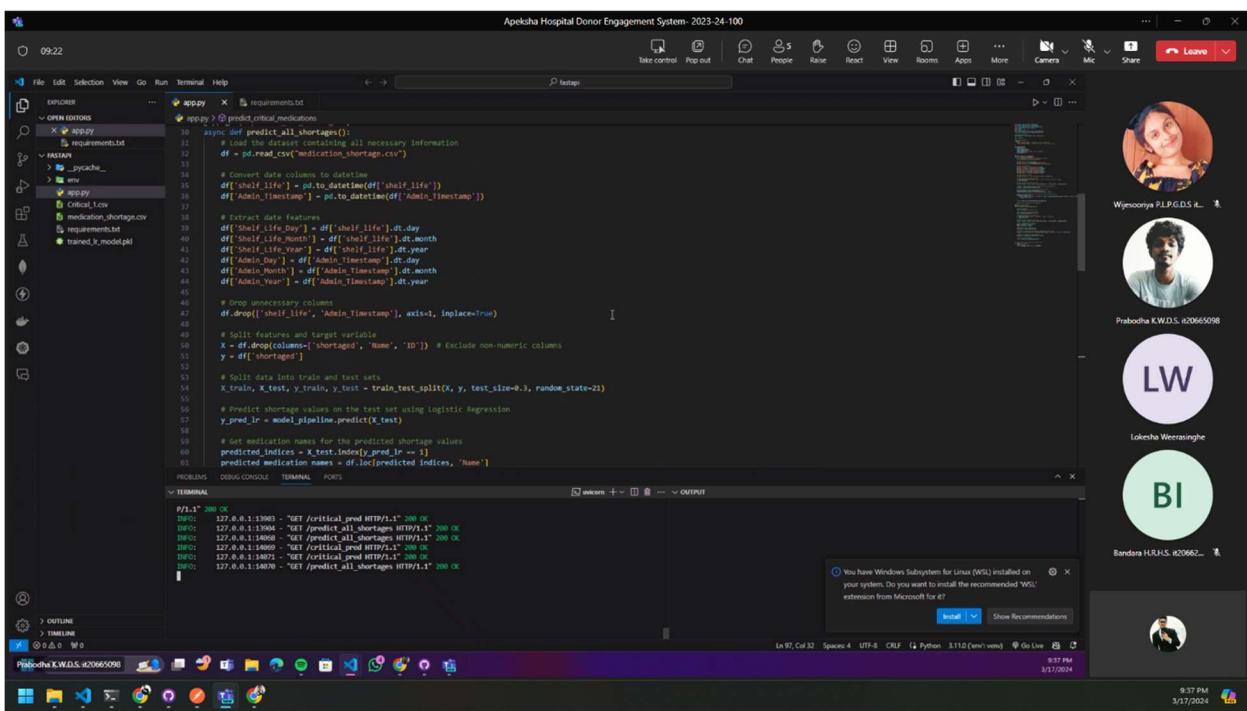
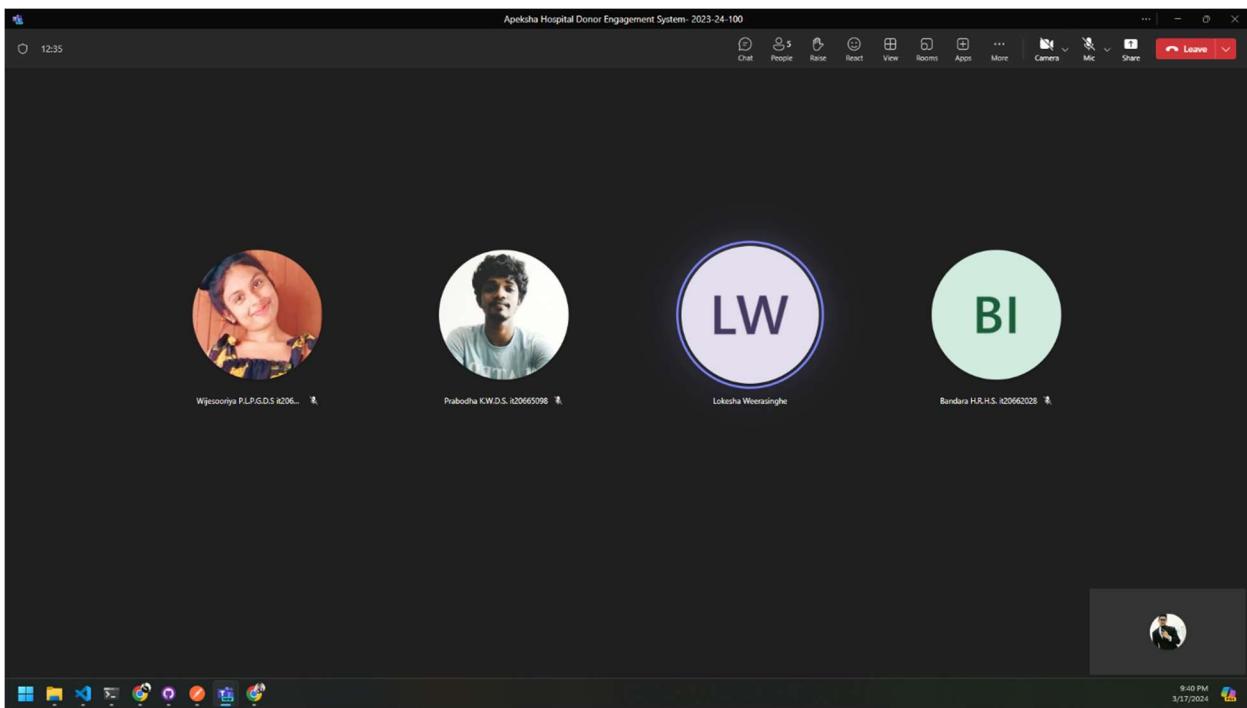
4. Research Paper

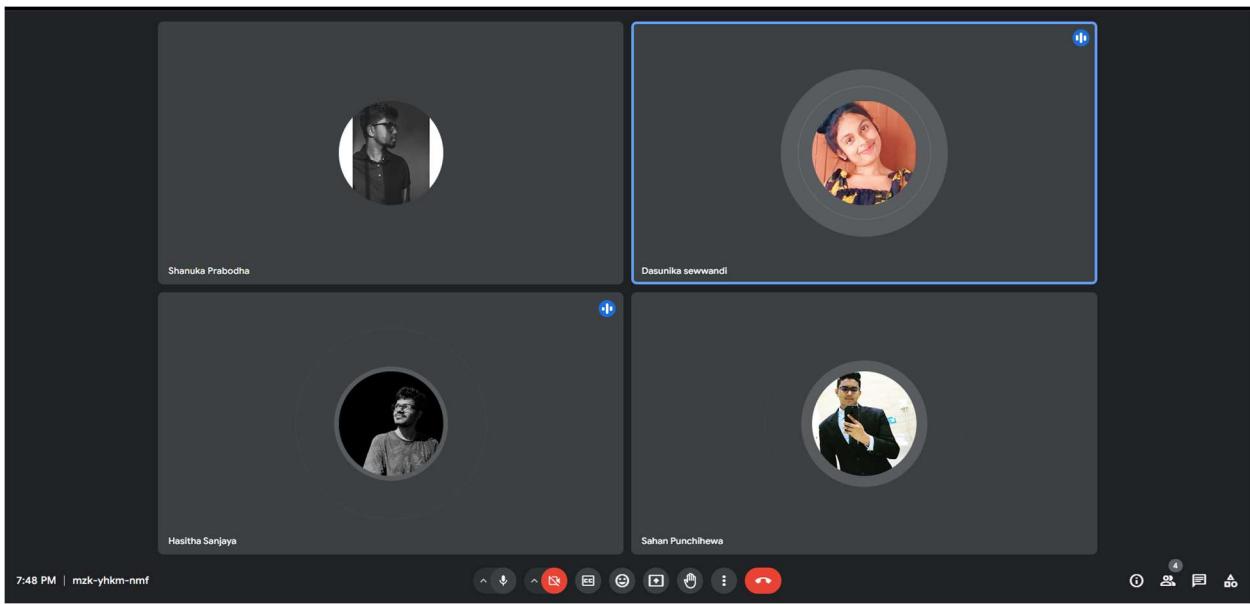
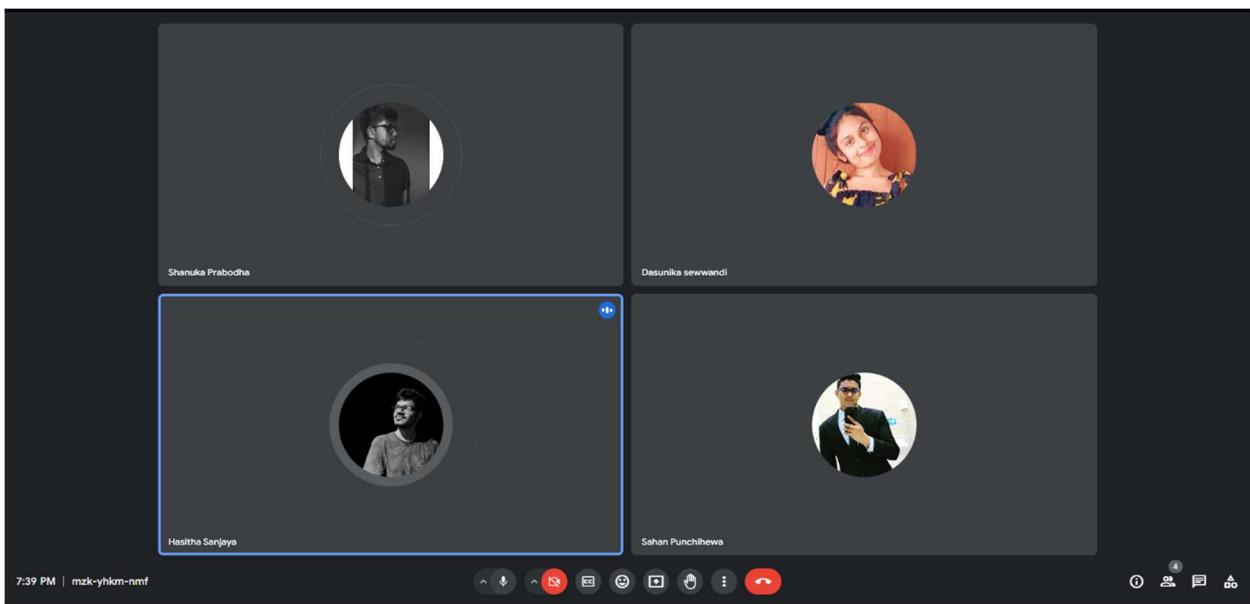
The screenshot shows a Microsoft Word document with the following details:

- Title:** Donor Management and Scarce Medical Resource Allocation in a Cancer Hospital in Sri Lanka
- Authors:**
 - S.N Punchihewa, Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka, sahamapulip18@gmail.com
 - K.W.D.S Praboda, Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka, shanmukaryawasam3@gmail.com
 - H.R.H.S Bandara, Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka, hanthanausaj12@gmail.com
 - W.A.C Peiris, Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka, lokuhewa.w@slit.lk
- Abstract:** The role of healthcare management, optimizing donor engagement, inventory control, and resource allocation are paramount for ensuring efficient patient care. This research aims to address these challenges by integrating predictive analytics and machine learning algorithms into the management of essential item inventories and critical medication allocation at Apukha Hospital. The first component focuses on enhancing donor involvement and inventory management through an Inventory Donor Database System for Essential Items. By leveraging machine learning algorithms, this system identifies donor donation patterns and usage behavior patterns to suggest essential products effectively. By integrating donor engagement with inventory control, the system seeks to increase donor satisfaction and retention. The second component of engagement, ultimately bridging the gap between donor intentions and hospital needs. The second component targets the optimization of critical medication allocation using machine learning analytics. By analyzing historical data and donor behavior patterns, predictive models are developed to anticipate potential medication shortages and predict donor needs. This approach aims to address challenges such as low donor turnout and restricted availability of medical supplies, particularly a high demand for blood, crystal for Apukha Hospital's medical treatments. The third component focuses on medication resource management, featuring a medication shortage model and a critical medication identification model. These models utilize historical data and usage patterns to anticipate and mitigate medication shortages. This information is crucial for the hospital based on medical conditions and urgency levels, thus contributing to enhancing patient care through efficient medication resource management.
- Text:** Data-driven observations. This will improve both efficiency and patient care [1].
The main goal is to make a system that is easy for anyone to use and that takes into account both the needs of the hospital and the interests of donors. By integrating machine learning algorithms for new ways to solve problems in healthcare, we can look at important issues like identifying donors involved and how much of their time they are willing to contribute. We can also look at how to use the data to recommend products and figure out how to get more donors, make better use of resources, and make critical items more available [2].
In the dynamic landscape of healthcare, the availability of essential medications stands as a cornerstone for effective patient care. However, managing the supply chain of critical medications presents a formidable challenge, risking compromised treatment and adverse patient outcomes. Apukha Hospital faces significant difficulties in the task of identifying and prioritizing medications prone to shortages.
This research introduces a pioneering solution—the “Critical Medication Resource Recovery System.” By integrating advanced machine learning techniques and predictive analytics, this system aims to proactively address medication shortages and optimize resource allocation. Previous studies in this field, such as Rhodes(2016) [3] and Veen(2021) [4], which emphasize the significance of predictive models in identifying medication shortages and addressing them, serve as a foundation for developing a solution tailored for Apukha Hospital.
The aim is to create a system that tracks shortages, identifies future trends, and predicts needs, assessing compatibility with donations, and optimizing recommendations based on factors like urgency, availability, and expiration. Leveraging historical data, the system empowers Apukha Hospital to efficiently manage medication resources for timely and effective patient care.

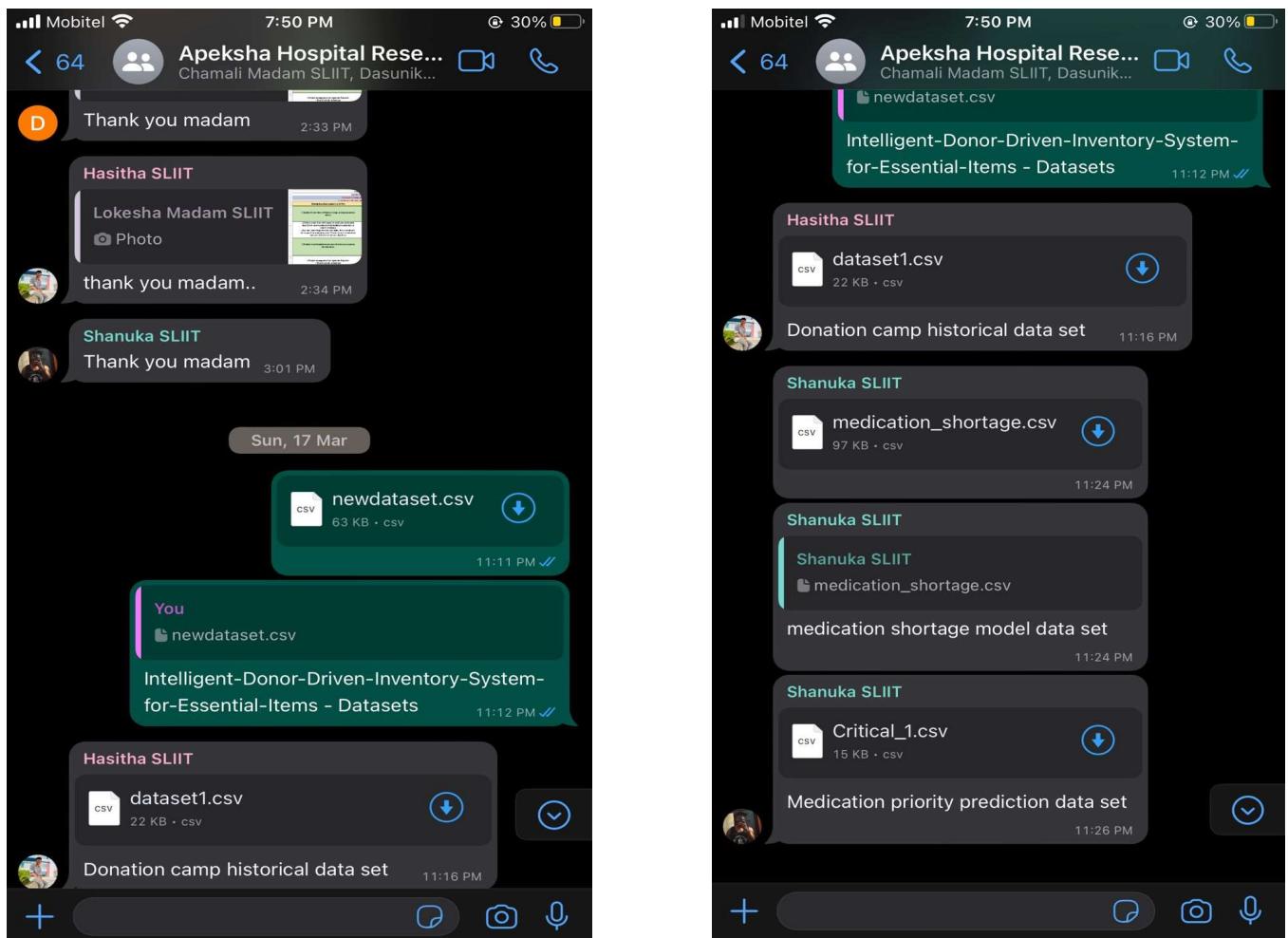
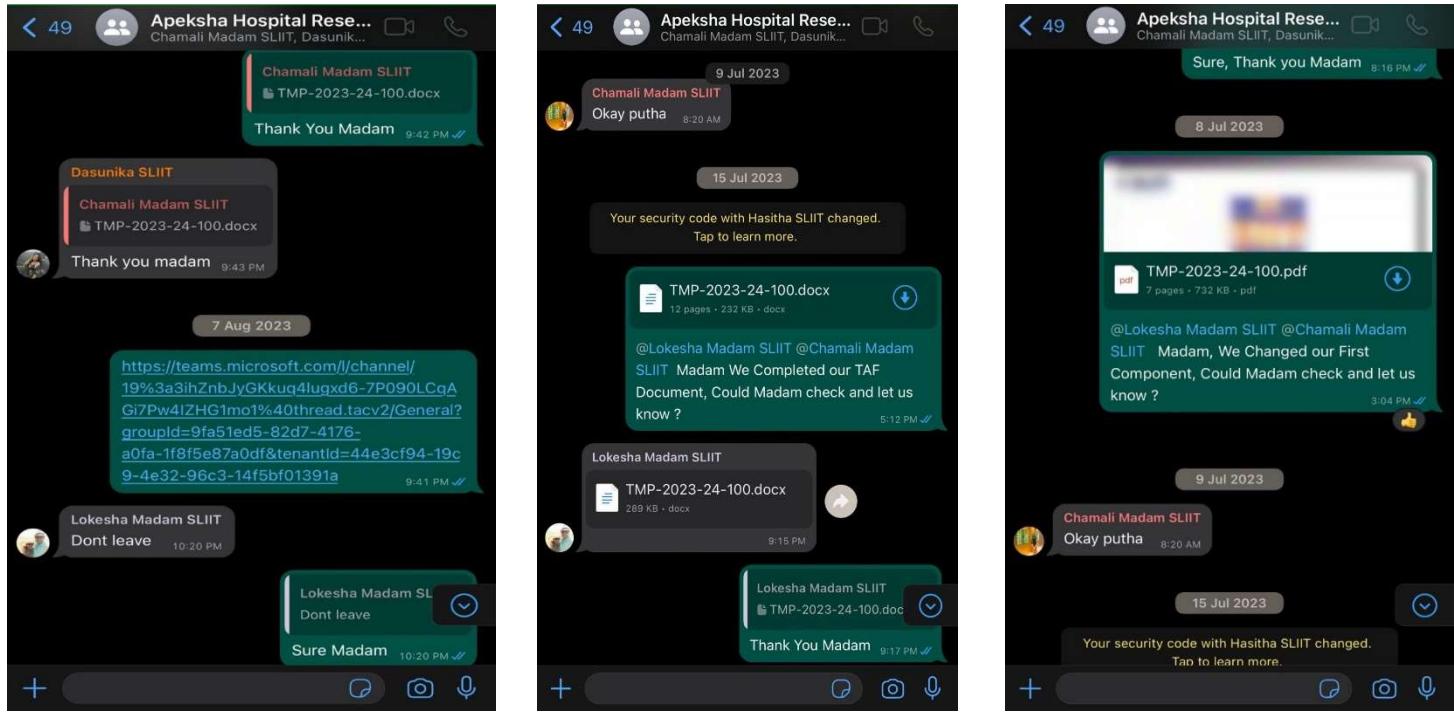
5. Screenshot of Meetings and Conversations







X



6. Project Management Tools & Screenshot

6.1 Overall Backlog

The screenshot shows the Azure DevOps interface for the 'Apexka Hospital Donor Engagement System-2023-24-100 Team'. The left sidebar is filled with various project management options like Overview, Boards, Work items, and Analytics views. The main area is titled 'Backlog' and displays a table of work items. The columns include Order, Work Item Type, Title, State, Effort, Business, Value Area, and Tags. There are 19 items listed, all of which are 'Business' type and have been 'Resolved'. The titles describe tasks such as implementing medication identification models, deploying new models, and integrating hair donation features.

Order	Work Item Type	Title	State	Effort	Business	Value Area	Tags
1	Feature	Implementing the Critical Medication Identification model u...	New		Business		
2	Feature	Deploy the new model in Render platform	New		Business		
3	Feature	Implementing Attendees Prediction amount display UI	New		Business		
4	Feature	Implementing Donor interaction UI-Medication	New		Business		
5	Feature	Implementing the new Attendees prediction model using R...	New		Business		
6	Feature	Pre-Process the new Dataset	New		Business		
7	Feature	Creating model using dataset from apexka hospital to fe...	New		Business		
8	Feature	System Deployment(hair Donation)	Active		Business		
9	Feature	Hair Donation Integration with the system	Active		Business		
10	Feature	UI Improvement in staff dashboard	Active		Business		
11	Feature	Create meaningful report for essential items usage	Active		Business		
12	Feature	Model Implemented to identify Color of Hair	Resolved		Business		
13	Feature	Creating test data set to feed the model	Resolved		Business		
14	Feature	Pre-process the Data set	Resolved		Business		
15	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business		
16	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business		
17	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business		
18	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business		
19	Feature	Deploy machine learning model in render server	Resolved		Business		

This screenshot shows the same Azure DevOps backlog as the previous one, but with a different set of 30 work items. All items are of the 'Business' type and have been 'Resolved'. The titles include tasks like 'Model Implemented to identify Color of Hair', 'Creating test data set to feed the model', and 'Deploy machine learning model in render server'. The interface is identical to the first screenshot, with the 'Backlog' tab selected in the top navigation bar.

Order	Work Item Type	Title	State	Effort	Business	Value Area	Tags
12	Feature	Model Implemented to identify Color of Hair	Resolved		Business		
13	Feature	Creating test data set to feed the model	Resolved		Business		
14	Feature	Pre-process the Data set	Resolved		Business		
15	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business		
16	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business		
17	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business		
18	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business		
19	Feature	Deploy machine learning model in render server	Resolved		Business		
20	Feature	Creating test data set to feed the model	Resolved		Business		
21	Feature	Pre-process the test Data set	Resolved		Business		
22	Feature	Implementing the Attendees Prediction model using Rando...	Resolved		Business		
23	Feature	Implement the Prediction Dashboard User Interface.	Resolved		Business		
24	Feature	Implement Donor-Driven Inventory System for Essential it...	Resolved		Business		
25	Feature	Implement machine learning model with localize dataset	Resolved		Business		
26	Feature	Implement Staff Dashboard	Resolved		Business		
27	Feature	Create a meaningful bar graph for essential items using a tr...	Resolved		Business		
28	Feature	Implement Machine Learning model using FastAPI	Resolved		Business		
29	Feature	Implementation of current stock of essential list and hospita...	Resolved		Business		
30	Feature	Implement items prediction dashboard	Resolved		Business		

Azure DevOps 2023-24-100-Apeksha-Hos... / Apeksha Hospital Donor En... / Boards / Boards

Search

Boards Analytics View as backlog

New Active 4/5 Resolved 24/5 Closed

+ New item

58 System Deployment(hair Donation) Active IT20660352 Wijesooriya P.L.P.G.D.S

59 Hair Donation Integration with the system Active IT20660352 Wijesooriya P.L.P.G.D.S

67 UI Improvement in staff dashboard Active Sahan Punchihewa

70 create meaningful report for essential items usage Active Sahan Punchihewa

74 Deploy the new model in Render platform Resolved Hasitha Sanjaya

75 Implementing out-side Organizer Management part Resolved Hasitha Sanjaya

71 Implementing the new Attendees prediction model using Random Forest Algorithm. Resolved Hasitha Sanjaya

73 Implementing Attendees Prediction amount display UI Resolved Hasitha Sanjaya

66 Creating model using dataset from apelkha hospital to feed the model Resolved Hasitha Sanjaya

69 Pre-Process the new Dataset Resolved

32 Implement the Prediction Dashboard User Interface. Closed Hasitha Sanjaya

23 Pre-process the test Data set Closed Hasitha Sanjaya

26 Creating test data set to feed the model Closed Hasitha Sanjaya

60 Creating data sets using localized data set from Apeksha Hospital to feed the model Closed Shanuka Praboda

61 Pre-processing new data sets Closed Shanuka Praboda

14 Collecting Real Medication Data from Apeksha Hospital Closed Shanuka Praboda

6.2 Personal Backlog

Azure DevOps 2023-24-100-Apeksha-Hos... / Apeksha Hospital Donor En... / Boards / Boards

Search

Boards Analytics View as backlog

Filter by keyword

Types Sahan Punchihewa States Area Iteration Tags Parent Work Item

New Active 5/5 Resolved 24/5 Closed

+ New item

67 UI Improvement in staff dashboard Active Sahan Punchihewa

70 create meaningful report for essential items usage Active Sahan Punchihewa

68 Deploy machine learning model in render server Resolved Sahan Punchihewa

73 Implement Donor-Driven Inventory System for Essential Items Model Resolved Sahan Punchihewa

62 Implement machine learning model with localize dataset Resolved Sahan Punchihewa

55 Implement Staff Dashboard Resolved Sahan Punchihewa

56 Create a meaningful bar graph for essential items using a trained machine learning model Resolved Sahan Punchihewa

17 Collecting Real Data from Apelkha Hospital for Intelligent Donor-Driven Inventory System for Essential Items Closed Sahan Punchihewa

21 Creating Test data for feed model Closed Sahan Punchihewa

22 Pre-Process data set Closed Sahan Punchihewa

27 Identify suitable algorithms for train machine learning model ... Closed Sahan Punchihewa

16 Collection Test Data - Intelligent Donor-Driven Inventory System for Essential Items Closed Sahan Punchihewa

The screenshot shows the Azure DevOps interface for the 'Apex Hospital Donor Eng...' project. The left sidebar navigation includes 'Overview', 'Boards', 'Work items', 'Backlogs' (selected), 'Sprints', 'Queries', 'Delivery Plans', 'Analytics views', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. The main content area displays the 'Backlog' for the 'Apex Hospital Donor Engagement System-2023-24-100 Team'. The backlog table has the following data:

Order	Work Item Type	Title	State	Effort	Business	Value Area	Tags
5	Feature	UI Improvement in staff dashboard	Active		Business		
6	Feature	Create meaningful report for essential items usage	Active		Business		
21	Feature	Deploy machine learning model in render server	Resolved		Business		
22	Feature	Implement Donor-Driven Inventory System for Essential Items	Resolved		Business		
23	Feature	Implement machine learning model with localize dataset	Resolved		Business		
24	Feature	Implement Staff Dashboard	Resolved		Business		
25	Feature	Create a meaningful bar graph for essential items using a template	Resolved		Business		
26	Feature	Implement Machine Learning model using FastAPI	Resolved		Business		
+	27	Implementation of current stock of essential list and hospital	Resolved		Business		
28	Feature	Implement items prediction dashboard	Resolved		Business		

7. Trained Model Screenshot

7.1 Decision Tree Regression

This screenshot shows a Jupyter Notebook interface with the title "jupyter InventoryPredictionAll Last checkpoint: 02/28/2024 (autosaved)". The notebook has a Python 3 (ipykernel) kernel. The code in the cells is used to load a dataset, preprocess it, define features and target variables, split the data, initialize a Decision Tree Regression model, and train it. The output cell shows the trained model object.

```
In [2]: import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor
from sklearn.preprocessing import LabelEncoder

In [3]: # Load the dataset
df = pd.read_csv('newdataset.csv')

In [4]: # Preprocess the data
le_ItemName = LabelEncoder()
le_Month = LabelEncoder()
df["ItemName"] = le_ItemName.fit_transform(df["ItemName"])
df["Month"] = le_Month.fit_transform(df["Month"])

In [5]: # Define a dictionary to map encoded values to original item names
item_name_mapping = dict(zip(le_ItemName.classes_, le_ItemName.classes_))
month_name_mapping = dict(zip(le_Month.classes_, le_Month.classes_))

In [6]: # Define independent variables (features) and the target variable
X = df[["ItemName", "ItemID", "ItemCategory", "UsageHistory", "Month"]]
y = df["RequestedQuantity"]

In [7]: # Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

In [8]: # Initialize Decision Tree Regression model
model = DecisionTreeRegressor(random_state=42)

In [9]: # Train the Decision Tree model
model.fit(X_train, y_train)

Out[9]: DecisionTreeRegressor(random_state=42)
```

This screenshot shows a Jupyter Notebook interface with the same title and kernel as the previous one. The code in the cells is used to make predictions for all items in the dataset. The output cell displays the predicted values for various items, mapping encoded values back to original item names and months.

```
In [10]: # Make predictions for all items in the dataset
predictions = model.predict(X)

# Print the predictions for all items
for i in range(len(df)):
    item_name = item_name_mapping[df.iat[i][0]] # Map encoded value to original item name
    month_name_original = month_name_mapping[df.iat[i][1]] # Map encoded value to original month
    month = X.iat[i][1]
    prediction = round(predictions[i])
    print("Item Name : ", item_name)
    print("Month: ", month_name_original)
    print("Prediction: ", prediction)
    print()

Item Name : BedCovers
Month: January
Prediction: 30

Item Name : Blankets
Month: April
Prediction: 65

Item Name : Diapers
Month: May
Prediction: 140

Item Name : Jackets
Month: March
Prediction: 450

Item Name : LargeBedSheet
Month: August
Prediction: 120

In [11]: model.score(X_train, y_train)

Out[11]: 0.9886072809011376
```

8 Web Implementation

The screenshot shows the homepage of the LEND A HAND website. At the top, there is a navigation bar with links for Home, Contact Us, About Us, FAQ, Sign-Up (in red), and Sign-In (in green). The main heading "LEND A HAND" is displayed with a heart icon. Below the heading, a large red banner with the text "Join hands for a better tomorrow." is shown. A subtext below it reads: "Your generosity powers miracles. Dive in, explore, and see the difference you make. At Apeksha Hospital's Donor Hub, impact is our currency. Join us, spark change, and be a beacon of hope." To the right of the text is a cartoon illustration of a red blood drop character and a doctor character holding hands, with a red heart-shaped balloon above them. Below the banner, there is a horizontal scrollable image strip showing various photos of people and a globe icon.

The screenshot shows the "Current Essential Items" section of the website. At the top, there is a navigation bar with links for Home, Contact Us, About Us, FAQ, Dashboard, Medication, Essentials, and Logout. Below the navigation bar, there is a search bar labeled "Search Items..." and a dropdown menu labeled "All Months". The main content area displays eight items in a grid:

Item	Description	Quantity Needed	Donate
VimBottle	A green Vim bottle.	Quantity Needed: 10	Donate
WashingPowder	A row of various washing powder boxes (Tide, Rin, Surf Excel, Ariel).	Quantity Needed: 30	Donate
TowelRack	A wooden towel rack with towels hanging from it.	Quantity Needed: 3	Donate
ShoeRack	A red three-tier plastic shoe rack.	Quantity Needed: 4	Donate
PillowCases	Stacked blue pillowcases.	Quantity Needed: 30	Donate
Mettress	A blue folding mattress.	Quantity Needed: 100	Donate
Rexine	A black Rexine box.	Quantity Needed: 40	Donate
ElectricKettle	A black electric kettle.	Quantity Needed: 30	Donate



LEND A HAND

Home Contact Us About Us FAQ Dashboard Medication Essentials [Logout](#)

Make Donation

Item Name

Quantity

User Name

Email Address

Hand Over Date

Donation Type




LEND A HAND

Home Contact Us About Us FAQ Dashboard [Logout](#)

Staff Dashboard

GENERAL

ESSENTIALS

MEDICATION

BLOOD DONATION

Donation Request

[Essential](#) [Medication](#)

Item Name	Quantity	Donor Name	Donor Email	Hand Over Date	Donation Type	Change Status	Status
BedCovers	5		sahanpunchihewa18@gmail.com		ESSENTIAL	Accepted	ACCEPTED
BedCovers	5		sahanpunchihewa18@gmail.com		ESSENTIAL	Rejected	REJECT
BedCovers	5		sahanpunchihewa18@gmail.com		ESSENTIAL	Accepted	ACCEPTED
BedCovers	5		sahanpunchihewa18@gmail.com		ESSENTIAL	Pending	PENDING
BedCovers	5	test	sahanpunchihewa18@gmail.com		ESSENTIAL	Pending	PENDING
BedCovers	5	test	sahanpunchihewa18@gmail.com		ESSENTIAL	Pending	PENDING
BedCovers	10	Sahan	sahanpunchihewa18@gmail.com		ESSENTIAL	Pending	PENDING





Staff Dashboard

GENERAL

ESSENTIALS

MEDICATION

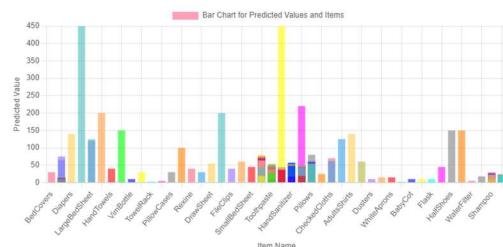
BLOOD DONATION

DONATION

DONORS

ESSENTIALS ITEMS PREDICTION

ESSENTIAL ITEMS



Select Month



Predicted Essential Items

Search Items by name

Search

Filter by Month: All

Item Name	Month	Predicted Value	Priority
BedCovers	January	30	Low
VimBottle	January	10	Low
ShoeRack	January	4	Low
DrawSheet	January	55	Medium
FaceMask	January	53	Medium
Toothpaste	January	50	Medium
HandSanitizer	January	30	Low
BathTowels	February	200	High
Soap	February	150	High
TowelRack	February	3	Low



Image	Item Name	Item Code	Quantity In Stock	Quantity Status	Action	Action
	Diapers	103	115	Good		
	Jackets	104	124	Good		
	LargeBedSheet	105	175	Good		
	BedCovers	100	150	Good		

Item Name	Item ID
<input type="text"/>	<input type="text"/>
Quantity In Stock	
<input type="text"/>	
Image	
<input type="file"/> No file chosen	
<input type="button" value="Create Item"/>	

About Investor Relations Jobs Press Blog



© 2024 - Anuktha Humanitarian Development System



Your Pro Trial has expired. Would you like to upgrade to Pro? [Upgrade Now](#)

Recent Previews

- Env Changes Merge pull request #125 from ...
- Production Merge pull request #126 from ...
- Dev Merge pull request #125 from Research-23-100/...
Merge pull request #127 from Research-23-100/produ...
9d ago on `main`
- Donor section hosted model
- Feature/medication dashboard Merge br...
Merge pull request #86 from ...
9d ago on `main`
- hair-donation Frontend Updated
- blood-donor Merge pull request #51 from ...
Merge pull request #52 from ...
9d ago on `main`

Projects

- apeksha-hospital-donor-engagement-sys...
Merge pull request #127 from Research-23-100/...
9d ago on `main`

Vercel Ship

Render

Overview

Search services

Service Name	Status	Type	Runtime	Region	Last Deployed
Intelligent-Donor-Driven-Inv...	Deployed	Web Service	Python 3	Singapore	2 months ago
Apeksha-Hospital-Donor-En...	Deployed	Web Service	Docker	Singapore	2 months ago

Active 2 Suspended 0 All 2

Feedback [Invite a Friend](#) [Contact Support](#)

XX

9 Version Control and Contribution

The image consists of two screenshots of a GitHub-like platform interface.

The top screenshot shows the "Repositories" page. It has a sidebar on the left with options like All, Public, Private, Sources, Forks, Archived, Mirrors, and Templates. The main area shows a search bar and a list of repositories. There are 2 repositories listed:

- Apeksha-Hospital-Donor-Engagement-System** (Public)
JavaScript · 1 branch · 0 stars · 0 forks · 0 commits · Updated last week
- Documents** (Private)
0 branches · 0 stars · 0 forks · 0 commits · Updated on Oct 17, 2023

The bottom screenshot shows the detailed view of the **Apeksha-Hospital-Donor-Engagement-System** repository. The top navigation bar includes Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. The repository details show:

- Code tab selected
- Branch: main · 13 branches · 0 tags
- Last commit: 1e429db · last week · 240 Commits
- Files: .github/workflows (Docker File Config), backend (Env Changes), frontend (Env Changes), notification (Donor Dashboard), README.md (Initial commit), docker-compose.yml (Docker File Config)
- README content: Apeksha-Hospital-Donor-Engagement-System
- Activity: SahanPunchihewa Merge pull request #127 from Research-2023-23-100/produ...
- Contributors: SahanPunchihewa, hasitha1998, ShanukaPrabodha
- Releases: No releases published · Create a new release
- Packages: No packages published · Publish your first package
- About: apeksha-hospital-donor-engagement-sys... · Readme, Activity, Custom properties, 0 stars, 0 watching, 1 fork, Report repository

SahanPunchihewa / Research-2023-23-100 / Apeksha-Hospital-Donor-Engagement-System

Type / to search

Code Issues Pull requests Actions Projects Security Insights Settings

Commits

main

SahanPunchihewa All time

Commits on Apr 29, 2024

Merge pull request #127 from Research-2023-23-100/production

SahanPunchihewa committed last week ✓ 1 / 1

Merge pull request #126 from Research-2023-23-100/dev

SahanPunchihewa committed last week ✓ 2 / 2

Commits on Apr 28, 2024

Merge pull request #125 from Research-2023-23-100/donor-login

SahanPunchihewa committed last week ✓ 3 / 3

Env Changes

SahanPunchihewa committed last week ✓ 3 / 3

Merge pull request #124 from Research-2023-23-100/production

SahanPunchihewa committed last week ✓ 1 / 1

Merge pull request #123 from Research-2023-23-100/dev

SahanPunchihewa committed last week ✓ 2 / 2

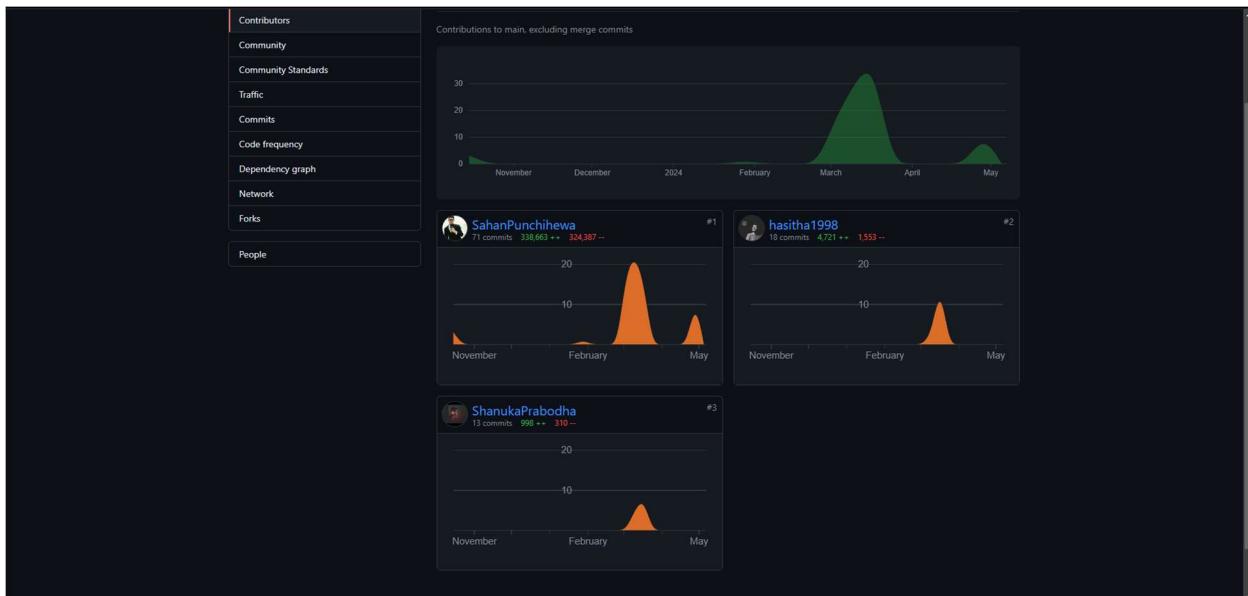
Merge pull request #122 from Research-2023-23-100/donor-login

SahanPunchihewa committed last week ✓ 3 / 3

Env Changes

SahanPunchihewa committed last week ✓ 3 / 3

Merge pull request #121 from Research-2023-23-100/production



The screenshot shows the GitHub Insights interface for the repository 'Research-2023-23-100 / Apeksha-Hospital-Donor-Engagement-System'. The 'People' tab is selected. It displays a list of 4 people with access to the repository, categorized by their role: 'Everyone' (Hasitha Sanjaya, Dassunika Sewwandi, Sahan Punchihewa, Shanuka Prabodha) and 'Outside collaborators' (None). Each user entry includes a profile picture, name, GitHub handle, and their level of access (Admin or Write).

User	Access Level
Hasitha Sanjaya hasitha1998	Admin
Dassunika Sewwandi IT20660352-dasu	Write
Sahan Punchihewa SahanPunchihewa	Admin
Shanuka Prabodha ShanukaPrabodha	Admin

At the bottom of the page, there is a footer with links to GitHub's Terms, Privacy, Security, Status, Docs, Contact, Manage cookies, and a link to 'Do not share my personal information'.

Apeksha Hospital Donor Engagement System

2023-24-100

Status Document 02

Prabodha K.W.D.S

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Critical Medication Priority Recommender System

2023-24-100

Status Document 02

Prabodha K.W.D.S

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Declaration of The Candidate & Supervisor

I declare that this is my work, and this proposal does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Student ID	Signature
Prabodha K.W.D. S	IT20665098	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor:

Date:

Signature of the Co-supervisor:

Date:

Table of Contents

Introduction	IV
Work Break-Down Structure.....	VI
Research Paper	VI
Screenshots of meetings and conversations with supervisor	VII
Personal Backlog	X
Overall Backlog.....	XI
Implemented Machine Learning Models.....	XII
Trained Using Test Data	XII
Critical Medication Identification Model	XII
Shortage Prediction Model	XII
Trained Using Localized Dataset Collected from Apeksha Hospital	XIII
Critical Medication Identification Model	XIII
Shortage Prediction Model	XIII
Web Implementation.....	XIV
Deployment and Hosting	XVI
Python Model-Render.....	XVI
Version Control and Contribution.....	XVIII

Introduction

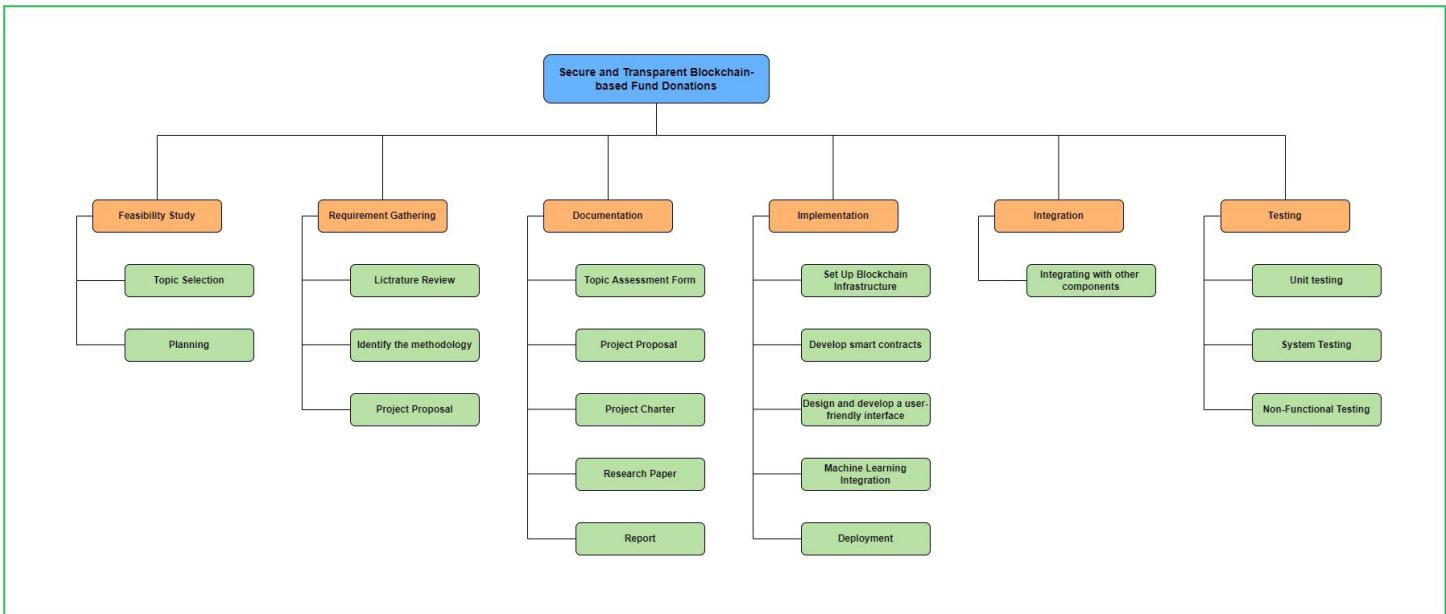
The "Critical Medication Priority Recommender System" presents a novel machine learning-based solution aimed at addressing the persistent challenge faced by Apeksha Hospital in identifying and prioritizing essential medications in low supply, ultimately affecting patient care quality. This research component outlines the development of a medication recommendation platform that strategically assesses historical medication usage, donation records, and medication attributes. The system integrates two pivotal machine learning models: a "Shortage Prediction" model to anticipate medication scarcities, a "Critical Medication Identification" model ranking medications based on medical significance. This process is further enhanced through optimization algorithms that rigorously prioritize medications by importance, availability, and expiration. Leveraging existing medication donation frameworks, this research not only fills a critical gap in medication allocation but also bolsters resource efficiency, offering a comprehensive solution for ensuring high-quality patient care in the face of limited resources.

Keywords: Medication shortage, Machine learning-based system, Shortage prediction model, Donation utilization

Gantt Chart

Task Name	Semester 1						Semester 2					
	June	July	August	September	October	November	December	January	February	March	April	May
Feasibility Study	Red											
Topic Selection	Yellow	Yellow										
Topic Evaluation												
Background Study	Red	Red	Yellow									
Background Study and Literature Survey												
Proposal Draft												
Project Proposal		Yellow	Yellow	Yellow								
Proposal Presentation												
Proposal Report												
Project Initiation			Red	Red								
Data gathering			Yellow	Yellow	Yellow							
Selecting technologies												
Project Implementation				Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	
Set up Blockchain Infrastructure												
Develop smart Contracts												
Design and Develop a user friendly interface												
Machine Learning Integration												
Research Paper												
Final Implementation												
Unit Testing												
Evaluating and error fixing												
Integrating with other components												
Final Stages									Red	Red	Red	
System Integration									Yellow	Yellow		
System Testing										Yellow		
Evaluating and error fixing											Yellow	
Final Report											Yellow	Yellow

Work Break-Down Structure



Research Paper

Donor Management and Scarce Medical Resource Allocation in a Cancer Hospital in Sri Lanka

S.N Punchihewa
Faculty of Computing
Sri Lanka Institute of Information
Technology
Malabe, Sri Lanka
sabumukhawaya.vansun@gmail.com

K.W.D.S Prabodha
Faculty of Computing
Sri Lanka Institute of Information
Technology
Malabe, Sri Lanka
shumukhawaya.vansun@gmail.com

H.R.H.S Bandara
Faculty of Computing
Sri Lanka Institute of Information
Technology
Malabe, Sri Lanka
hasthasinghav12@gmail.com

P.L.P.G.D.S Upeneswara
Faculty of Computing
Sri Lanka Institute of Information
Technology
Malabe, Sri Lanka
dusunukarajeewasena@gmail.com

K.M.P. Wannigeke
Faculty of Computing
Sri Lanka Institute of Information
Technology
Malabe, Sri Lanka
lakshmi.w@slut.ac.lk

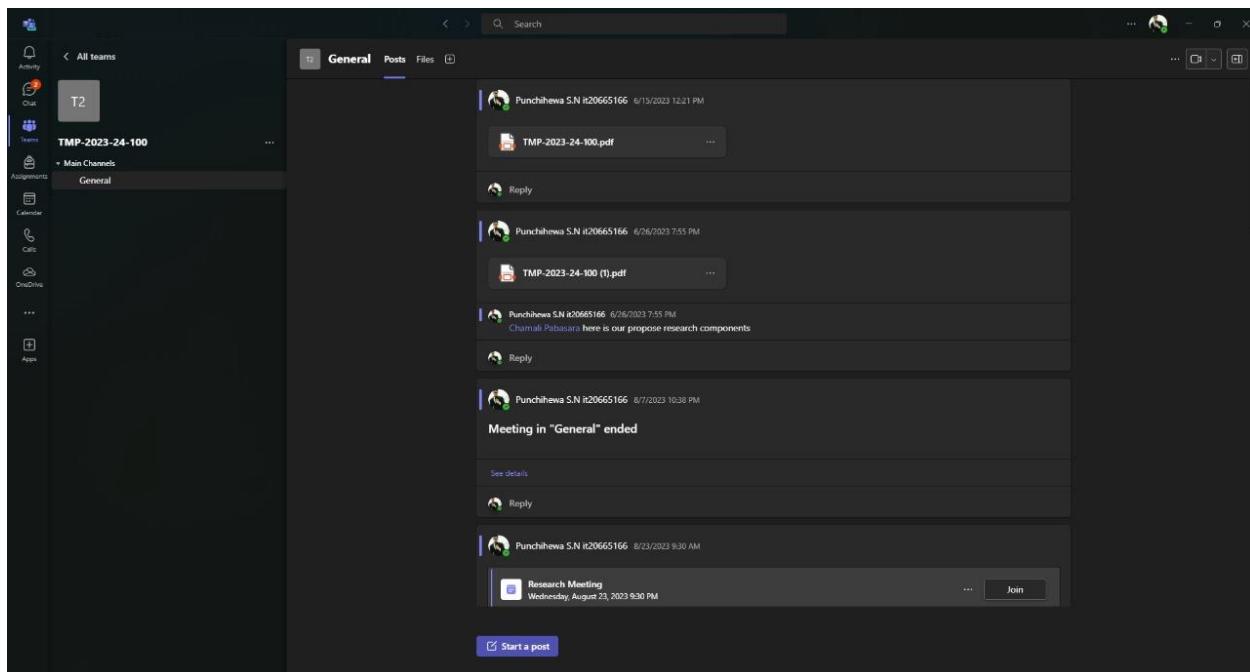
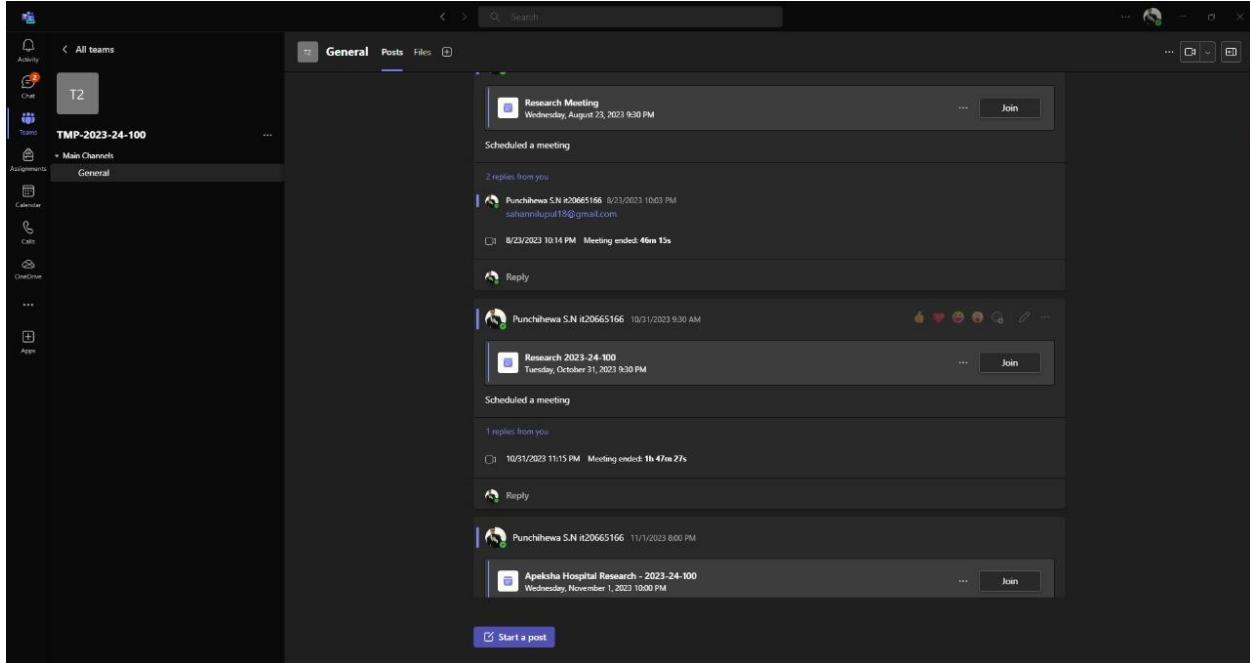
W.A.C Palansara
Faculty of Computing
Sri Lanka Institute of Information
Technology
Malabe, Sri Lanka
chamali.p@slut.ac.lk

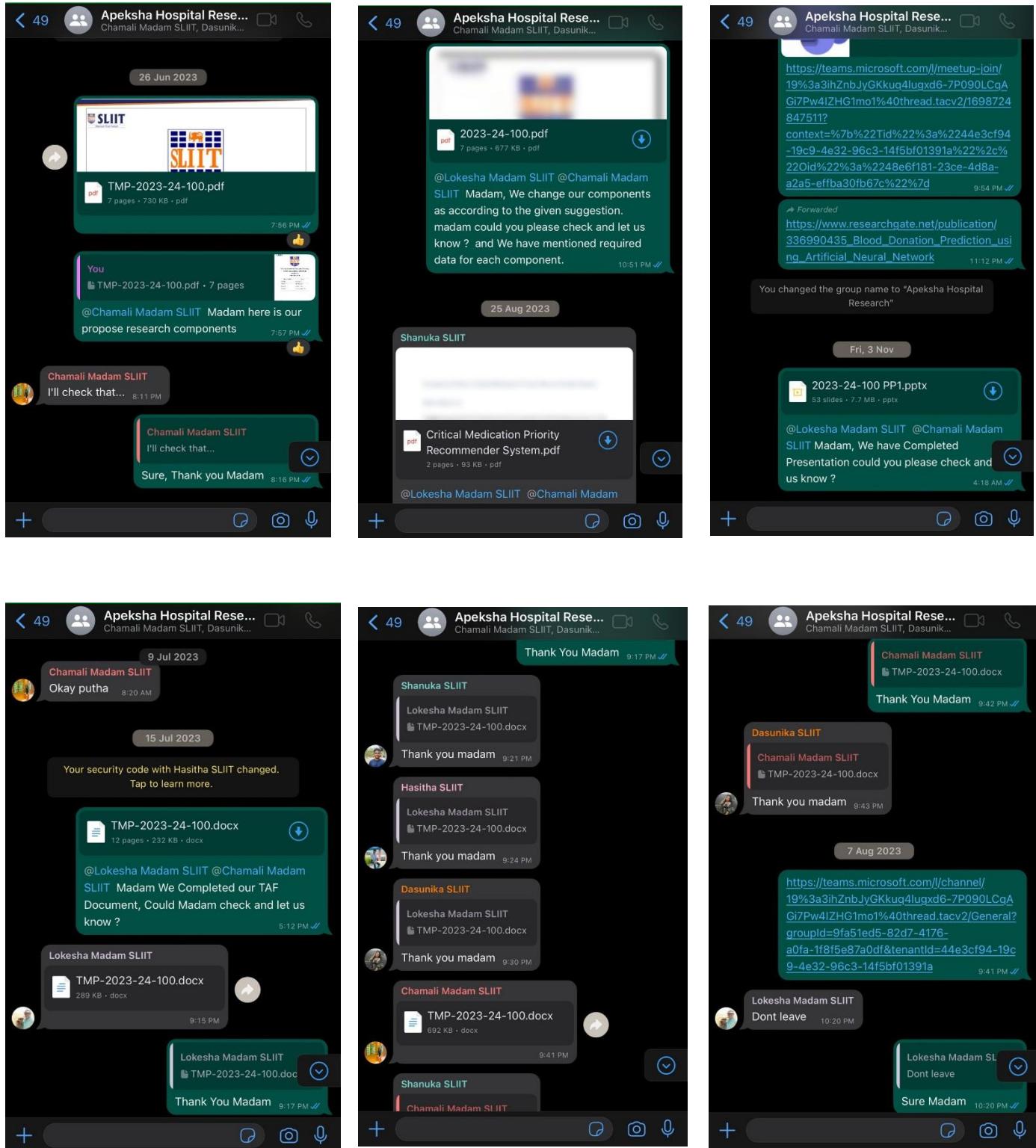
Abstract: In the realm of healthcare management, optimizing donor engagement, inventory control, and resource allocation are fundamental for ensuring patient care. This research proposes a comprehensive approach to address these challenges by integrating predictive analytics and machine learning algorithms into the management of essential item inventories and donor resources. The first component focuses on donor management and resource allocation at Apoksha Hospital. The first component focuses on enhancing donor involvement and inventory management through an Intelligent Donor-Driven Inventory System for Hospitals. By understanding donor behavior, demographic patterns, and giving patterns, the system aims to understand inventory preferences and giving patterns to suggest essential products effectively. By integrating donor engagement and inventory control, the system seeks to increase donor satisfaction and establish a positive cycle of engagement, ultimately bridging the gap between donor intentions and hospital needs. The second component targets the optimization of blood donation campaigns through predictive analytics. By analyzing historical data and donor engagement patterns, predictive models are developed to anticipate campaign success factors and optimize resource allocation. This approach helps to address challenges such as low donor turnout and resource waste. The third component focuses on maintaining and heightened supply of blood, crucial for Apoksha Hospital's medical treatments. The third component focuses on medication resource management by developing a predictive model and a critical medication identification matrix. These models utilize historical data and usage patterns to anticipate and mitigate potential shortages, as well as rank medications based on their importance, availability, and urgency levels, thus contributing to enhancing patient care through efficient medication resource management.

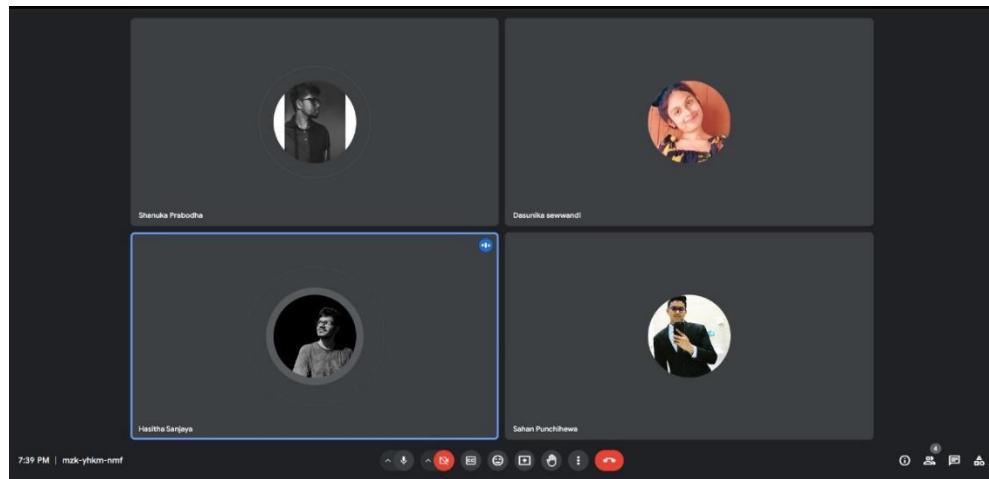
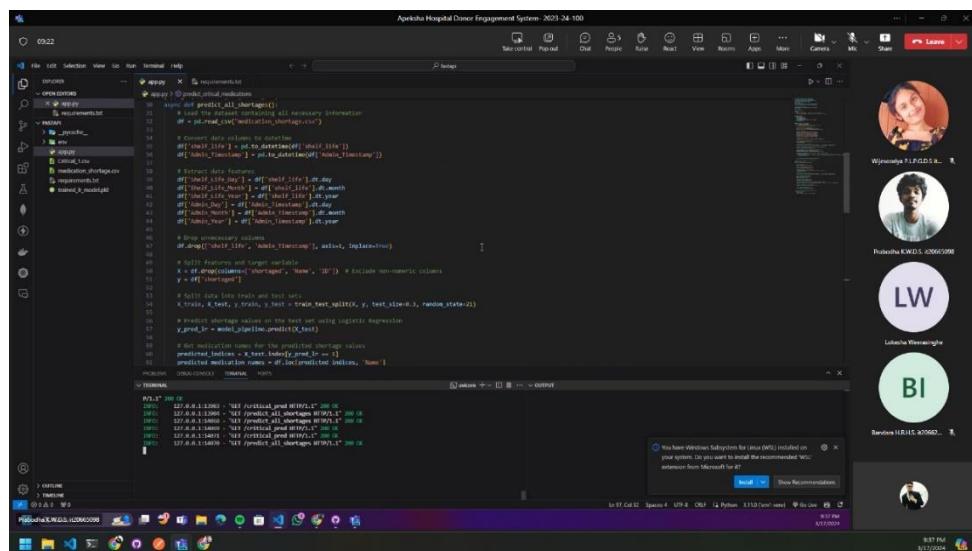
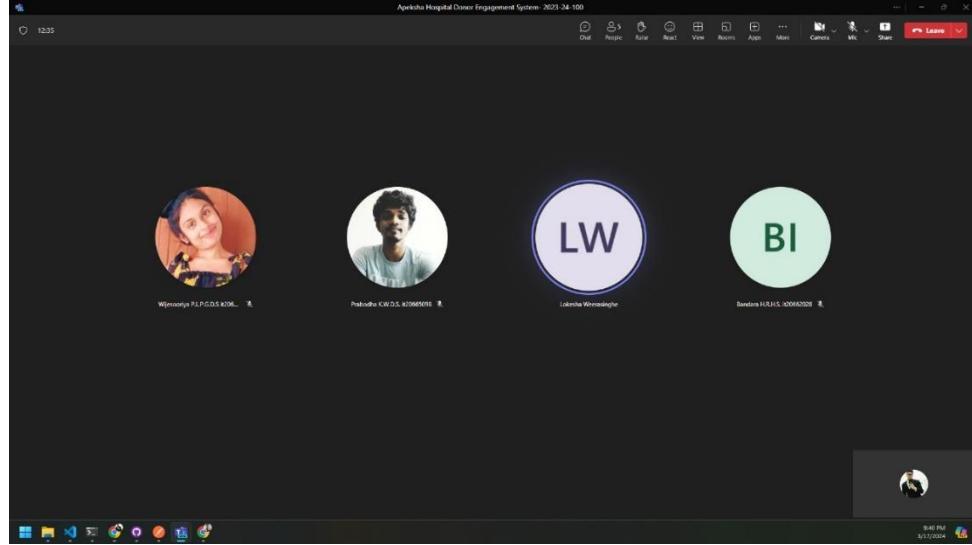
The research introduces a pioneering solution—the “Critical Medication Priority Recommender System”. By harnessing advanced machine learning techniques and predictive analytics, this system aids in predicting medication shortages and optimize allocation strategies. Building upon previous studies such as Rhodes(2016) [3] and Vest(2021) [4], which emphasize the importance of prioritizing and recommending essential medications for patient care, this research endeavors to develop an intelligent solution tailored for Apoksha Hospital.

The aim is to create a system predicting shortages, identifying vital medications as per patient needs, assessing compatibility with donor resources, and recommending based on factors like importance, availability, and expiration. Leveraging historical data, the system empowers Apoksha Hospital to efficiently manage medication resources for timely and effective patient care.

Screenshots of meetings and conversations with supervisor







Personal Backlog

Try the New Boards Hub for improved performance, accessibility, and new features. [Click here](#) to learn more.

Apexka Hospital Donor Engagement ... ★ ⓘ

Board Analytics View as Backlog

Filter by keyword

Types Shanuka Prabodha Tags Iteration Area Parent Work Item

New	Active	Resolved	Closed
+ New item	3/5	10/5	
<p>29 Implement the Medication Dashboard User Interface.</p> <p>Shanuka Prabodha</p> <p>State: New</p>	<p>14 Collecting Real Medication Data from Apexka Hospital</p> <p>Shanuka Prabodha</p> <p>State: Active</p>	<p>19 Creating test data set to feed the model</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>20 Pre-process the Data set</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>28 Implementing the Critical Medication Shortage Prediction model using Logistic Regression Algorithm.</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>24 Implementing the Critical Medication Identification model using Logistic Regression Algorithm.</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p>	<p>13 Collecting Test Data-Critical Medication Identification and shortage</p> <p>Shanuka Prabodha</p> <p>State: Closed</p>

Try the New Boards Hub for improved performance, accessibility, and new features. [Click here](#) to learn more.

Apexka Hospital Donor Engagement ... ★ ⓘ

Board Analytics View as Backlog

Filter by keyword

Types Shanuka Prabodha Tags Iteration Area Parent Work Item

New	Active	Resolved	Closed
+ New item	6/5	22/5	
		<p>60 Creating data sets using localized data set from Apexka Hospital to feed the model</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>63 Implementing the Critical Medication Identification model using Random Forest Classifier.</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>29 Implement the Medication Dashboard User Interface.</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>14 Collecting Real Medication Data from Apexka Hospital</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p> <p>61 Pre-processing new data sets</p> <p>Shanuka Prabodha</p> <p>State: Resolved</p>	<p>19 Creating test data set to feed the model</p> <p>Shanuka Prabodha</p> <p>State: Closed</p> <p>20 Pre-process the Data set</p> <p>Shanuka Prabodha</p> <p>State: Closed</p> <p>24 Implementing the Critical Medication Identification model using Logistic Regression Algorithm.</p> <p>Shanuka Prabodha</p> <p>State: Closed</p> <p>65 Implementing the Critical Medication Shortage Prediction model using Logistic Regression Algorithm.</p> <p>Shanuka Prabodha</p> <p>State: Closed</p> <p>28 Implementing the Critical Medication Shortage Prediction model using Logistic Regression Algorithm.</p> <p>Shanuka Prabodha</p> <p>State: Closed</p>

X

Overall Backlog

New	Active	Resolved	Closed
44 Combine all the models to implement the final model. New IT20660352 Wijesooriya P.L.P.G.D.S	46 Model Implemented to identify Dryness of Hair Active IT20660352 Wijesooriya P.L.P.G.D.S	53 Model Implemented to identify Dandruff & lice of Hair Resolved IT20660352 Wijesooriya P.L.P.G.D.S	45 Collection Test Data- Promoting Quality Hair Donation for Cancer Patients Closed Bandara H.R.H.S. IT20660166
29 Implement the Medication Dashboard User Interface. New Shanuka Prabodha	48 Implement the Web Application with User-friendly Interface. Active IT20660352 Wijesooriya P.L.P.G.D.S	52 Model Implemented to identify length type of Hair Resolved IT20660352 Wijesooriya P.L.P.G.D.S	15 Collecting Test Data - Previous Donation campaign attendees count and date Closed Bandara H.R.H.S. IT20660208
30 Implement items prediction dashboard New Punchihewa S.N. IT20665166	47 Model Implemented to identify bleached of Hair Active IT20660352 Wijesooriya P.L.P.G.D.S	51 Model Implemented to identify Color of Hair Resolved IT20660352 Wijesooriya P.L.P.G.D.S	16 Collection Test Data - Intelligent Donor-Driven Inventory System for Essential Items Closed Punchihewa S.N. IT20660166
32 Implement the Prediction Dashboard User Interface. New Bandara H.R.H.S. IT20660228	14 Collecting Real Medication Data from Apeksha Hospital Active Shanuka Prabodha	50 Data Pre-Processing and Create Data Set With Data Augmentation Resolved IT20660352 Wijesooriya P.L.P.G.D.S	13 Collecting Test Data-Critical Medication Identification and shortage Closed Shanuka Prabodha
	17 Collecting Real Data from Apeksha Hospital for Intelligent Donor-Driven Inventory System for Essential Items Active Bandara H.R.H.S. IT20660228	19 Creating test data set to feed the model Resolved Shanuka Prabodha	
	18 Collecting Real Donation Campaign historical data from Apeksha Hospital Active Bandara H.R.H.S. IT20660228	49 Identification of best architecture for transfer learning Resolved IT20660352 Wijesooriya P.L.P.G.D.S	
	12 Feature Model Implemented to identify Color of Hair	10 Feature Resolved	11 Feature Business
	13 Feature Creating test data set to feed the model	11 Feature Resolved	11 Feature Business
	14 Feature Pre-process the Data set	11 Feature Resolved	11 Feature Business
	15 Feature Implementing the Critical Medication Identification model u...	11 Feature Resolved	11 Feature Business
	16 Feature Implementing the Critical Medication Shortage Prediction ...	11 Feature Resolved	11 Feature Business
	17 Feature Implementing the Critical Medication Identification model u...	11 Feature Resolved	11 Feature Business
	18 Feature Implementing the Critical Medication Shortage Prediction ...	11 Feature Resolved	11 Feature Business
	19 Feature Deploy machine learning model in render server	11 Feature Resolved	11 Feature Business
	20 Feature Creating test data set to feed the model	11 Feature Resolved	11 Feature Business
	21 Feature Pre-process the test Data set	11 Feature Resolved	11 Feature Business
	22 Feature Implementing the Attendees Prediction model using Rando...	11 Feature Resolved	11 Feature Business
	23 Feature Implement the Prediction Dashboard User Interface.	11 Feature Resolved	11 Feature Business
	24 Feature Implement Donor-Driven Inventory System for Essential Ite...	11 Feature Resolved	11 Feature Business
	25 Feature Implement machine learning model with localize dataset	11 Feature Resolved	11 Feature Business
	26 Feature Implement Staff Dashboard	11 Feature Resolved	11 Feature Business
	27 Feature Create a meaningful bar graph for essential items using a tr...	11 Feature Resolved	11 Feature Business
	28 Feature Implement Machine Learning model using FastAPI	11 Feature Resolved	11 Feature Business
	29 Feature Implementation of current stock of essential list and hospita...	11 Feature Resolved	11 Feature Business
	30 Feature Implement items prediction dashboard	11 Feature Resolved	11 Feature Business

Order	Work Item Type	Title	State	Effort	Business Area	Tags
12	Feature	Model Implemented to identify Color of Hair	Resolved		Business	
13	Feature	Creating test data set to feed the model	Resolved		Business	
14	Feature	Pre-process the Data set	Resolved		Business	
15	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business	
16	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business	
17	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business	
18	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business	
19	Feature	Deploy machine learning model in render server	Resolved		Business	
20	Feature	Creating test data set to feed the model	Resolved		Business	
21	Feature	Pre-process the test Data set	Resolved		Business	
22	Feature	Implementing the Attendees Prediction model using Rando...	Resolved		Business	
23	Feature	Implement the Prediction Dashboard User Interface.	Resolved		Business	
24	Feature	Implement Donor-Driven Inventory System for Essential Ite...	Resolved		Business	
25	Feature	Implement machine learning model with localize dataset	Resolved		Business	
26	Feature	Implement Staff Dashboard	Resolved		Business	
27	Feature	Create a meaningful bar graph for essential items using a tr...	Resolved		Business	
28	Feature	Implement Machine Learning model using FastAPI	Resolved		Business	
29	Feature	Implementation of current stock of essential list and hospita...	Resolved		Business	
30	Feature	Implement items prediction dashboard	Resolved		Business	

Implemented Machine Learning Models

Trained Using Test Data

Critical Medication Identification Model

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.linear_model import LogisticRegression

# Read the CSV file into a DataFrame
data = pd.read_csv('critical_medication.csv')

data['Shelf_Life'] = pd.to_datetime(data['Shelf_Life'])
data['Shelf_Life_Day'] = data['Shelf_Life'].dt.day
data['Shelf_Life_Month'] = data['Shelf_Life'].dt.month
data['Shelf_Life_Year'] = data['Shelf_Life'].dt.year

data.drop(['Shelf_Life'], axis=1, inplace=True)

# Define the features (predictors) and the target variable
X = data[['Patients', 'Usage', 'Emergency_Usage', 'Shelf_Life_Day', 'Shelf_Life_Month', 'Shelf_Life_Year']]
y = data['Importance_Level']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

log_reg=LogisticRegression(random_state=0).fit(X_train,y_train)

log_reg.predict(X_train)
```

Shortage Prediction Model

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression

df=pd.read_csv("shortage_data.csv")

df['Shelf_Life'] = pd.to_datetime(df['Shelf_Life'])
df['Admin_Timestamp'] = pd.to_datetime(df['Admin_Timestamp'])

df['Shelf_Life_Day'] = df['Shelf_Life'].dt.day
df['Shelf_Life_Month'] = df['Shelf_Life'].dt.month
df['Shelf_Life_Year'] = df['Shelf_Life'].dt.year

df['Admin_Day'] = df['Admin_Timestamp'].dt.day
df['Admin_Month'] = df['Admin_Timestamp'].dt.month
df['Admin_Year'] = df['Admin_Timestamp'].dt.year

df.drop(['Shelf_Life', 'Admin_Timestamp'], axis=1, inplace=True)
X=df.drop(columns='Medication Shortage')

y=df['Medication Shortage']
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,random_state=21)
X_train

scaler=StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled=scaler.transform(X_test)
X_train_scaled

log_reg=LogisticRegression(random_state=0).fit(X_train_scaled,y_train)
log_reg.predict(X_train_scaled)
log_reg.score(X_train_scaled, y_train)
```

Trained Using Localized Dataset Collected from Apeksha Hospital

Critical Medication Identification Model

```
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.ensemble import RandomForestClassifier
4 from sklearn.metrics import accuracy_score
5
6 # Load the dataset
7 data = pd.read_csv("hospital_dataset.csv")
8
9 # Drop rows with missing target values
10 data = data.dropna(subset=['Priority'])
11
12 # Separate features (X) and target (y)
13 X = data.drop(columns=['SR No', 'Item Name', 'Group', 'Priority'])
14 y = data['Priority']
15
16 # Split data into train and test sets
17 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
18
19 # Train a Random Forest classifier
20 model = RandomForestClassifier()
21 model.fit(X_train, y_train)
22
23 # Make predictions on the test set
24 predictions = model.predict(X_test)
25
26 # Calculate accuracy
27 accuracy = accuracy_score(y_test, predictions)
28
29 # Create a DataFrame with item names and predicted priority values
30 output_df = pd.DataFrame({'Item Name': data.loc[X_test.index, 'Item Name'], 'Predicted Priority': predictions})
31
32 # Display the output and accuracy
33 print("Predicted values:")
34 print(output_df)
35 print("\nAccuracy:", accuracy)
36
37 print("Train set shape:", X_train.shape)
38 print("Test set shape:", X_test.shape)
39
40
```

Shortage Prediction Model

```
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.preprocessing import StandardScaler, OneHotEncoder
4 from sklearn.compose import ColumnTransformer
5 from sklearn.pipeline import Pipeline
6 from sklearn.ensemble import RandomForestClassifier # Change
7 from sklearn.impute import SimpleImputer
8 from sklearn.metrics import accuracy_score
9
10 # Load the dataset containing all necessary information
11 df = pd.read_csv("medication_shortage.csv")
12
13 # Convert date columns to datetime
14 df['shelf_life'] = pd.to_datetime(df['shelf_life'])
15 df['Admin_Timestamp'] = pd.to_datetime(df['Admin_Timestamp'])
16
17 # Extract date features
18 df['Shelf_Life_Day'] = df['shelf_life'].dt.day
19 df['Shelf_Life_Month'] = df['shelf_life'].dt.month
20 df['Shelf_Life_Year'] = df['shelf_life'].dt.year
21 df['Admin_Day'] = df['Admin_Timestamp'].dt.day
22 df['Admin_Month'] = df['Admin_Timestamp'].dt.month
23 df['Admin_Year'] = df['Admin_Timestamp'].dt.year
24
25 # Drop unnecessary columns
26 df.drop(['shelf_life', 'Admin_Timestamp'], axis=1, inplace=True)
27
28 # Split features and target variable
29 X = df.drop(columns=['shortaged', 'Name', 'ID']) # Exclude non-numeric columns
30 y = df['shortaged']
31
32 # Split data into train and test sets
33 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
34
35 # Define column transformer to handle numeric and categorical features
36 numeric_features = X.select_dtypes(include=['int64', 'float64']).columns
37 numeric_transformer = Pipeline(steps=[
38     ('imputer', SimpleImputer(strategy='median')),
39     ('scaler', StandardScaler())])
40
41 categorical_features = X.select_dtypes(include=['object']).columns
```

Web Implementation

The screenshot shows the homepage of the LEND A HAND website. At the top, there is a navigation bar with links for Home, Contact Us, About Us, FAQ, Sign-Up, and Sign-In. The main content area features a red header with the text "Join hands for a better tomorrow." Below this, a message says "Your generosity powers miracles. Dive in, explore, and see the difference you make. At Apeksha Hospital's Donor Hub, impact is our currency. Join us, spark change, and be a beacon of hope." To the right is a cartoon illustration of a red blood drop character and a medical bottle character holding hands, with a red heart-shaped balloon above them. Below the illustration is a horizontal banner with several small images.

The screenshot shows a table titled "Urgent Medications". The table has columns for "Medication Name" and "Donate". It lists seven items with their respective donation request buttons:

	Medication Name	Donate
瑞士	Carbachol Intraocular Sol. 0.01% ImL Dropper Bot.	Place a donation Request
瑞士	Carboplatin Inj. 450mg/45mLVia	Place a donation Request
瑞士	Carboprost Tromethamine Injection 250mcg/ml	Place a donation Request
瑞士	Cefuroxime Inj. 750mg Vial	Place a donation Request
瑞士	Cefuroxime Tab. 500mg	Place a donation Request
瑞士	Nystatin Tab. 500,000IU	Place a donation Request
瑞士	Vancomycin Inj. 500mg Vial	Place a donation Request

The screenshot shows a "Make Donation" form. The fields include:

- Item Name: Carbachol%20Intraocular%20Sol.%200.01%ImL%20Dropper
- Quantity: Enter quantity
- User Name: Sahan Punchihewa
- Email Address: sahanilupul18@gmail.com
- Hand Over Date: mm/dd/yyyy (dropdown menu)
- Donation Type: ESSENTIAL (dropdown menu)

A large blue "Submit" button is at the bottom of the form.



Predicted Medication Shortages

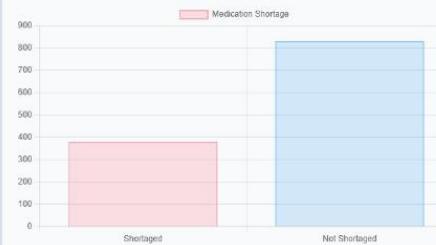
Filter by Shortage: All

Search by Medication Name:

Index	Name	Shortage Prediction
1	30mg Tab. Atazanavir (as sulphate) Cap.300 mg	not shortaged
2	Abacavir Sulfate 60mg +Lamivudine 30mg Tab.	not shortaged
3	Abacavir Tab. 300mg	not shortaged
4	Abciximab IV Infu. 10mg/5mL Vial	not shortaged
5	Abiraterone Acetate Tab.250mg	not shortaged

[Previous](#) [Next](#)

Medication Counts



Medication Priority Level

Search Medications

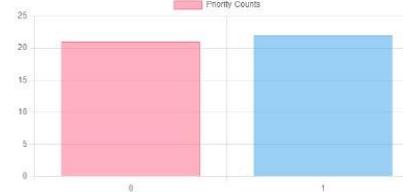
Filter by Priority

Medications

Index	Name	Priority
1	Benzoyl Peroxide Gel 2.5%, 20gTube	Low
2	Nystatin Tab. 50,000IU	High
3	Captopril Tab. 25mg Losartan	Low
4	Meropenem Inj. 1g Vial	Low
5	Cefuroxime Tab. 500mg	Low

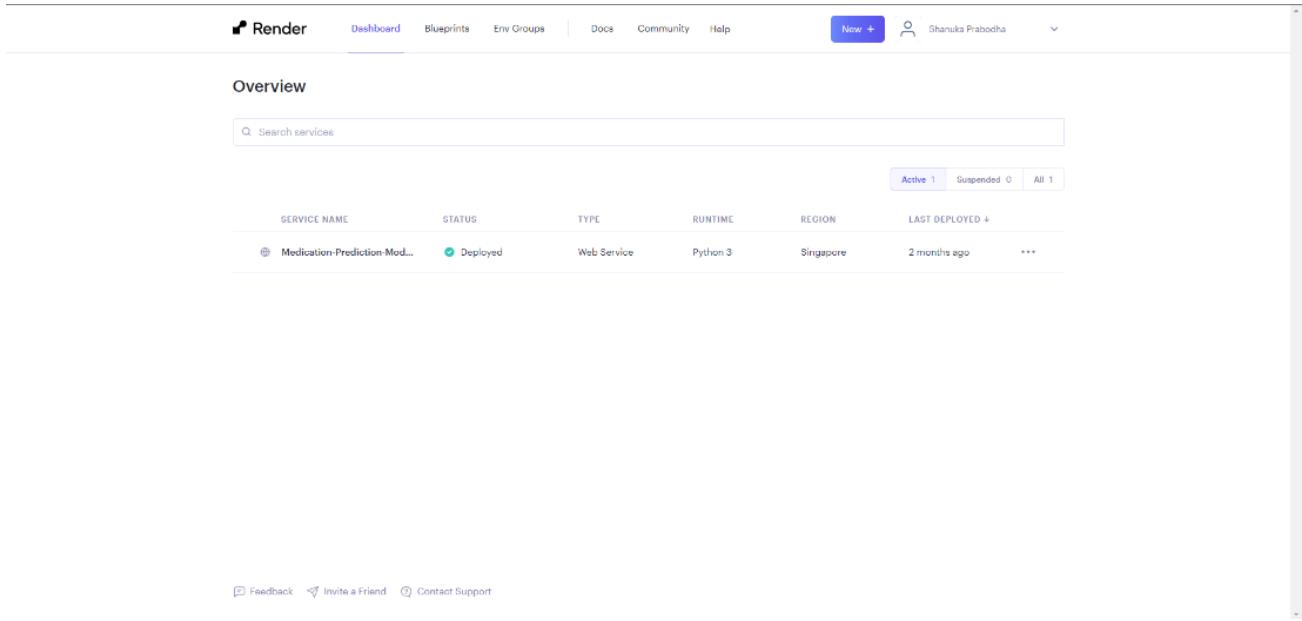
[1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#)

Priority Distribution

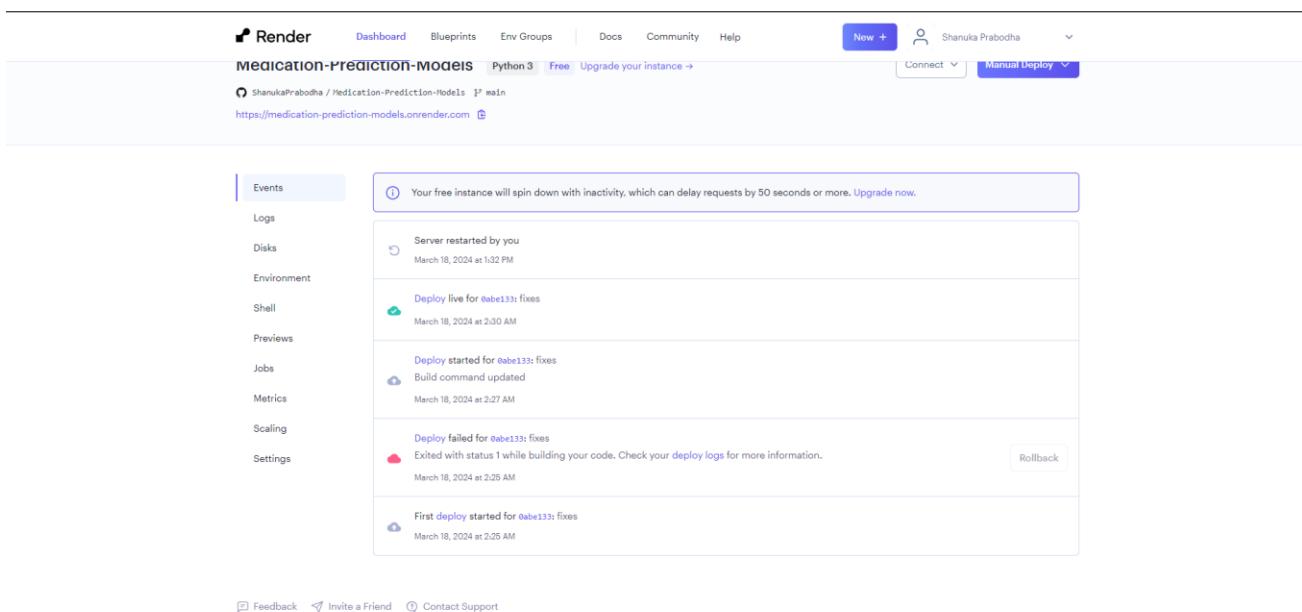


Deployment and Hosting

Python Model-Render



The screenshot shows the Render platform's Overview page. At the top, there is a navigation bar with links for Dashboard, Blueprints, Env Groups, Docs, Community, Help, and a New + button. A user profile for Shanuka Prabodha is shown. Below the navigation is a search bar labeled "Search services". A filter bar indicates "Active 1", "Suspended 0", and "All 1". A table lists one service: Medication-Prediction-Mod..., which is Deployed as a Web Service using Python 3 in the Singapore region, last deployed 2 months ago. At the bottom of the page are links for Feedback, Invite a Friend, and Contact Support.



The screenshot shows the Render platform's Events page for the service "Medication-Prediction-Models". The top navigation bar is identical to the Overview page. The main content area displays a timeline of events:

- Server restarted by you (March 18, 2024 at 1:32 PM)
- Deploy live for `eabe133: fixes` (March 18, 2024 at 2:30 AM)
- Deploy started for `eabe133: fixes` (Build command updated, March 18, 2024 at 2:27 AM)
- Deploy failed for `eabe133: fixes` (Exited with status 1 while building your code. Check your [deploy logs](#) for more information. Rollback, March 18, 2024 at 2:25 AM)
- First deploy started for `eabe133: fixes` (March 18, 2024 at 2:25 AM)

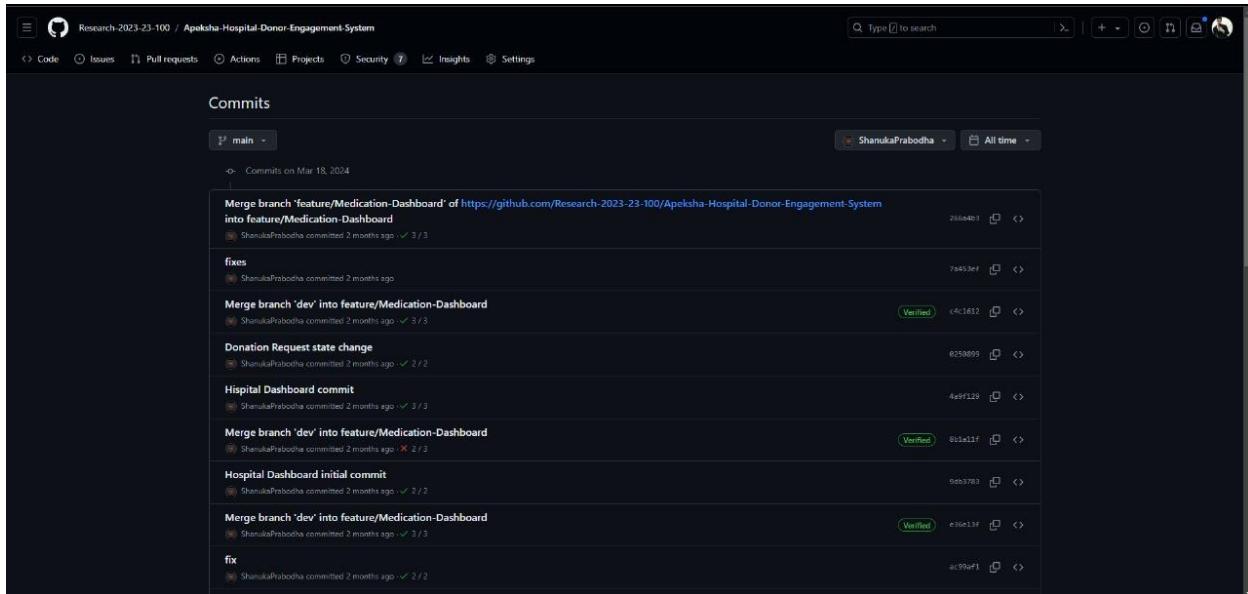
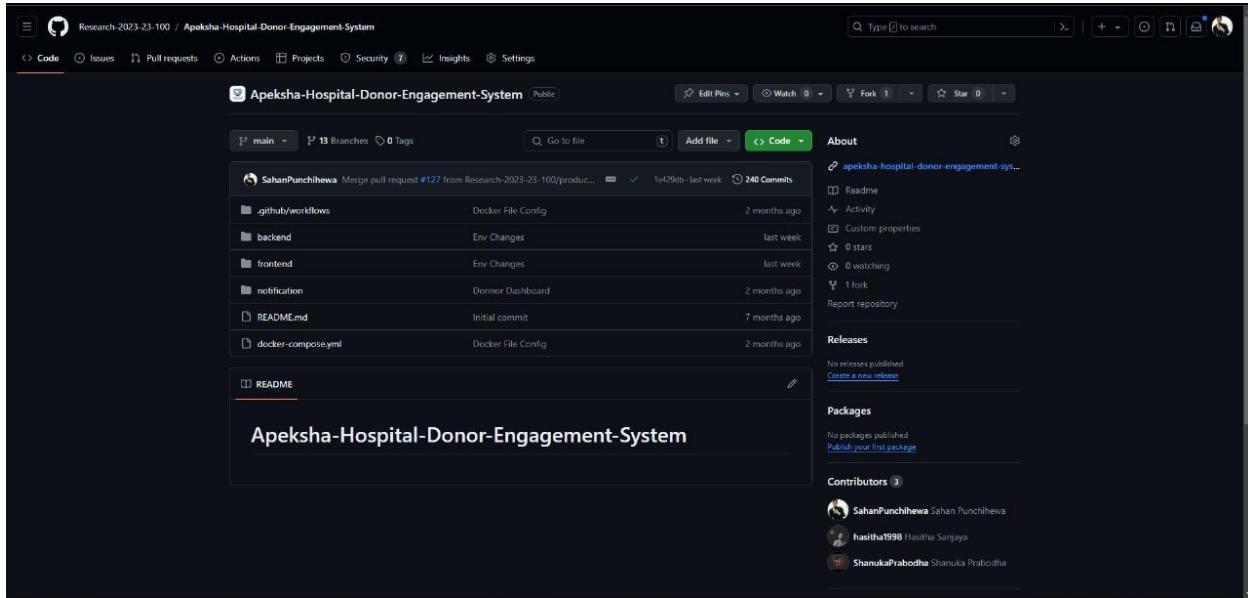
At the bottom of the page are links for Feedback, Invite a Friend, and Contact Support.

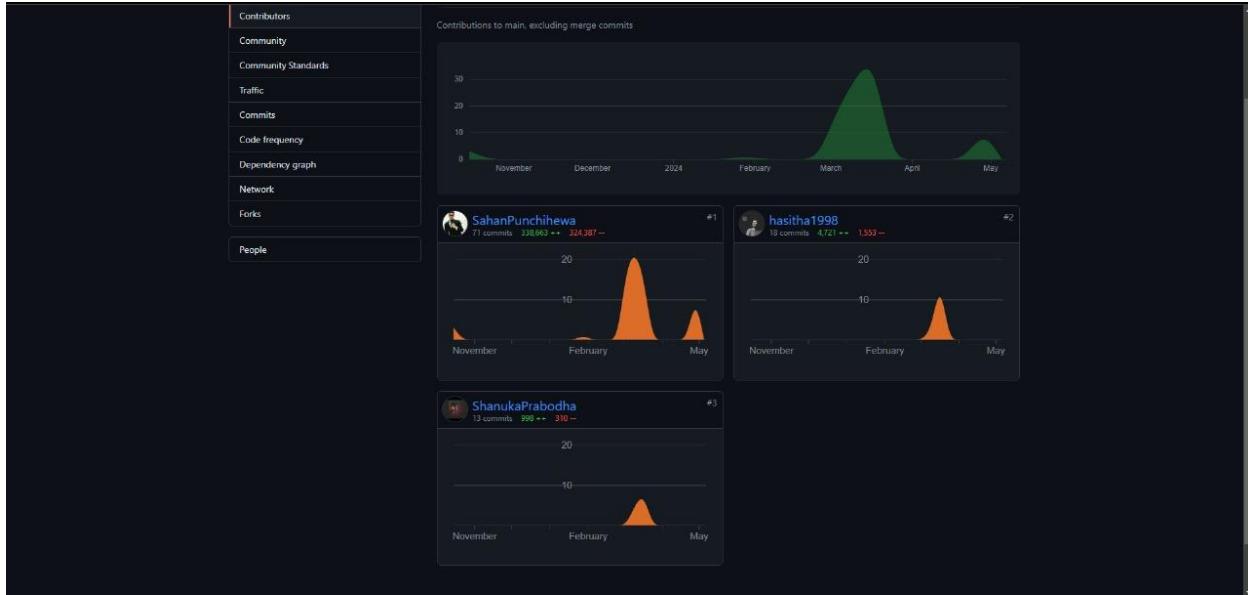
The screenshot shows the Render platform interface for the 'Medication-Prediction-Models' application. At the top, there are navigation links for Dashboard, Blueprints, Env Groups, Docs, Community, Help, and a 'New +' button. The user is Shanuka Prabodha. Below the header, it says '@ WEB SERVICE Medication-Prediction-Models Python 3 Free Upgrade your instance →'. There are 'Connect' and 'Manual Deploy' buttons. On the left, a sidebar lists Events, Logs (which is selected), Disks, Environment, Shell, Previews, Jobs, Metrics, Scaling, and Settings. The main area shows a log viewer with a search bar, a 'Live tail' button, and a timestamp of GMT+5:30. A message at the top of the log viewer says, 'Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more. Upgrade now.' The log entries are as follows:

```
May 7 08:41:42 PM INFO: 159.223.80.114:0 - "GET /predict_all_shortages HTTP/1.1" 200 OK
May 7 08:41:43 PM INFO: 159.223.80.114:0 - "GET /critical_pred HTTP/1.1" 200 OK
May 7 08:41:43 PM INFO: 159.223.80.114:0 - "GET /predict_all_shortages HTTP/1.1" 200 OK
May 7 08:41:44 PM INFO: 159.223.80.114:0 - "GET /critical_pred HTTP/1.1" 200 OK
May 7 08:41:44 PM INFO: 159.223.80.114:0 - "GET /critical_pred HTTP/1.1" 200 OK
May 7 08:44:05 PM INFO: 159.223.80.114:0 - "GET /predict_all_shortages HTTP/1.1" 200 OK
May 7 08:44:05 PM INFO: 159.223.80.114:0 - "GET /critical_pred HTTP/1.1" 200 OK
May 7 08:44:05 PM INFO: 159.223.80.114:0 - "GET /predict_all_shortages HTTP/1.1" 200 OK
May 7 09:00:13 PM INFO: Shutting down
May 7 09:00:14 PM INFO: Waiting for application shutdown.
May 7 09:00:14 PM INFO: Application shutdown complete.
May 7 09:00:14 PM INFO: Finished server process [41]
```

[Feedback](#) [Invite a Friend](#) [Contact Support](#)

Version Control and Contribution





Research-2023-23-100 / Apeksa-Hospital-Donor-Engagement-System

Type ⌘ to search

Code Issues Pull requests Actions Projects Security Insights Settings

Insights / People

Find a user Everyone Outside collaborators

4 people have access to this repository

	Hasitha Sanjaya hasitha1998	Admin
	Dasunika Sewwandi IT20660352-dasu	Write
	Sahan Punchihewa SahanPunchihewa	Admin
	Shanuka Prabodha ShanukaPrabodha	Admin

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

Apeksha Hospital Donor Engagement System

2023-24-100

Status Document 2

Bandara H.R.H.S

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Predictive Analytics for Donation Campaign Success

2023-24-100

Status Document 2

Bandara H.R.H.S

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Declaration of The Candidate & Supervisor

I declare that this is my work, and this proposal does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Student ID	Signature
Bandara H.R.H.S	IT20662028	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor:

Date:

Signature of the Co-supervisor:

Date:

Table of Contents

1.Introduction	IV
2.Gantt chart.....	V
3.Work Break down Chart.....	VI
4.Research Paper	VII
5.Screenshots of meetings and conversations with supervisor	VIII
6.Version Controller and Contribution	XII
7. Project Management Tools & Screenshot	XIII
7.1Personal Backlog	XIII
7.2Overall backlog.....	XIV
8.Trained New Model Screenshot	XIV
8.1Trained model python code.....	XIV
8.2Trained model accuracy.....	XV
8.3Hosted model in Render	XV
9.Web Implementation.....	XVI

1. Introduction

Blood holds paramount significance for human well-being, particularly among individuals afflicted with blood-related disorders such as leukemia and blood cancer. The pivotal role of blood in medical treatments, especially within Apeksha hospitals, underscores the constant and heightened need for blood donations. However, blood donation campaigns encounter multifaceted challenges impeding their success. Issues like low donor turnout, resource wastage including food, and insufficient medical equipment such as needles, blood collection bags, blood pressure monitors, and donor beds or chairs, along with inadequate medical staff, contribute to the inefficacy of these campaigns.

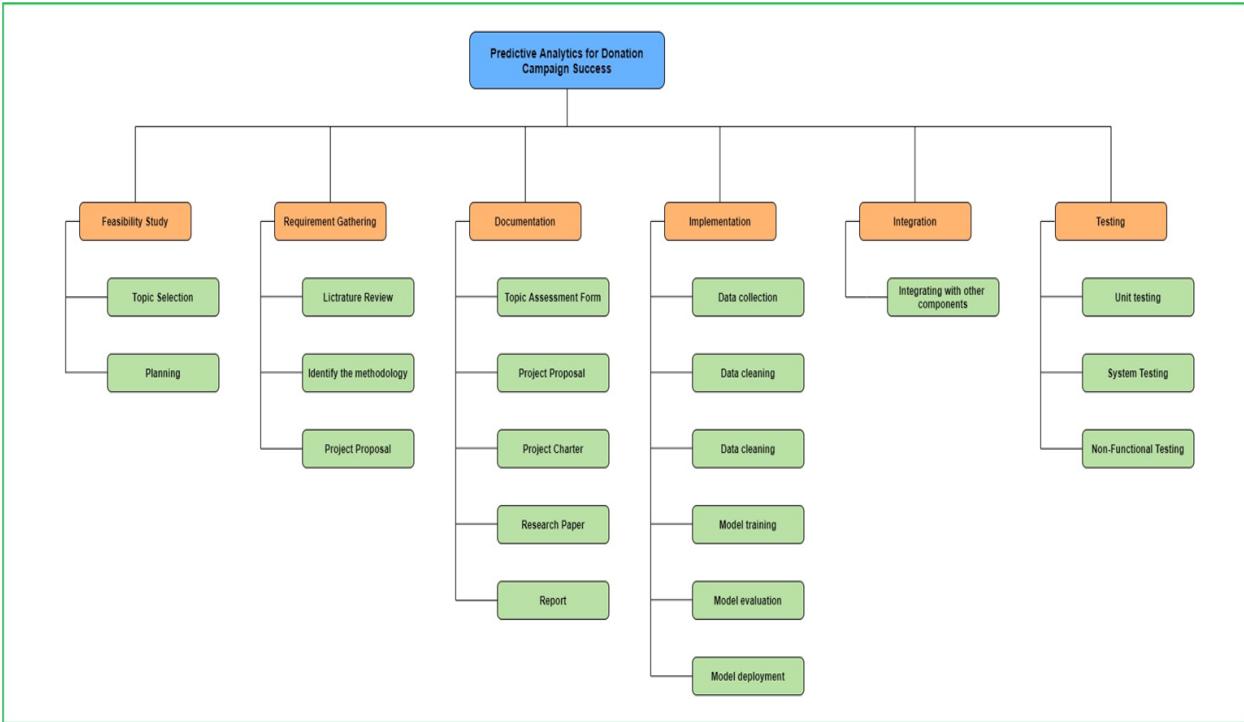
To surmount these obstacles and optimize the allocation of resources, this research advocates for the integration of predictive analytics empowered by machine learning. By harnessing historical data, predictive models can be devised to anticipate the likelihood of campaign success. These models would take into account an array of variables, including donor engagement patterns, external factors, and resource availability. By providing insights into donor behavior trends and optimal campaign conditions, machine learning could profoundly refine the strategic planning and execution of blood donation initiatives, leading to resource optimization. This impact is particularly crucial for Apeksha hospitals, where a constant and heightened supply of blood is indispensable.

Keywords: Machine learning, blood donation campaigns, donor turnout, resource optimization, campaign success, predictive analytics.

2. Gantt chart

Task Name	Semester 1						Semester 2					
	June	July	August	September	October	November	December	January	February	March	April	May
Feasibility Study	Red											
Topic Selection	Yellow											
Topic Evaluation	Yellow											
Background Study	Red	Red										
Background Study and Literature Survey		Yellow	Yellow									
Proposal Draft		Yellow	Yellow									
Project Proposal		Yellow	Yellow									
Proposal Presentation												
Proposal Report												
Project Initiation		Red	Red	Red								
Data gathering		Yellow	Yellow	Yellow								
Selecting technologies		Yellow	Yellow	Yellow								
Project Implementation												
Set up Blockchain Infrastructure		Yellow	Yellow	Yellow								
Develop smart Contracts		Yellow	Yellow	Yellow								
Design and Develop a user friendly interface		Yellow	Yellow	Yellow								
Machine Learning Integration		Yellow	Yellow	Yellow								
Research Paper		Yellow	Yellow	Yellow								
Final Implementation		Yellow	Yellow	Yellow								
Unit Testing		Yellow	Yellow	Yellow								
Evaluating and error fixing		Yellow	Yellow	Yellow								
Integrating with other components		Yellow	Yellow	Yellow								
Final Stages												
System Integration							Yellow	Yellow				
System Testing							Yellow	Yellow				
Evaluating and error fixing							Yellow	Yellow				
Final Report							Yellow	Yellow	Yellow	Yellow		

3. Work Breakdown Chart



4. Research Paper

Donor Management and Scarce Medical Resource Allocation in a Cancer Hospital in Sri Lanka

S.N.Punithewa
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
sahaniinp15@gmail.com

K.W.D.S.Prabodha
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
shamukakaryawana3@gmail.com

H.R.H.S.Bandara
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
hasthsanjanayala13@gmail.com

P.L.P.G.D.S.Wijenoriva
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
dawukawijenoriva@gmail.com

K.M.I.P.Weerasinghe
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
lokasha.w@slit.lk

W.A.C.Pabasara
Faculty of Computing
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
chamali.p@slit.lk

Abstract— In the realm of healthcare management, optimizing donor engagement, inventory control, and resource allocation are paramount for ensuring efficient patient care. This research proposes a comprehensive approach to address these challenges by integrating predictive analytics and machine learning algorithms into the management of essential item inventory, blood donation campaigns, and medication resource allocation at Apeksha Hospital. The first component focuses on enhancing donor engagement through a novel system named Intelligent Donor-Driven Inventory System (IDDIS). By leveraging machine learning algorithms, this system aims to understand inventory preferences and giving patterns to suggest essential products effectively. By integrating donor engagement with inventory control, the system seeks to increase donor satisfaction and establish a positive cycle of engagement and donation. The second component targets the optimization of blood donation campaigns through predictive analytics. By analyzing historical data and donor behavior patterns, predictive models are developed to anticipate campaign success factors and optimize resource allocation. This approach also addresses challenges such as low donor turnout and resource wastage, ultimately ensuring a constant and heightened supply of blood, crucial for Apeksha Hospital's medical treatments. The third component focuses on medication resource management by developing a shortage prediction model and a critical medication identification model. These models help predict potential shortages and propose solutions to mitigate potential shortages, as well as rank medications essential for the hospital based on medical conditions and urgency levels, thus contributing to enhancing patient care through efficient medication resource management.

data-driven observations. This will improve both efficiency and patient care [1].

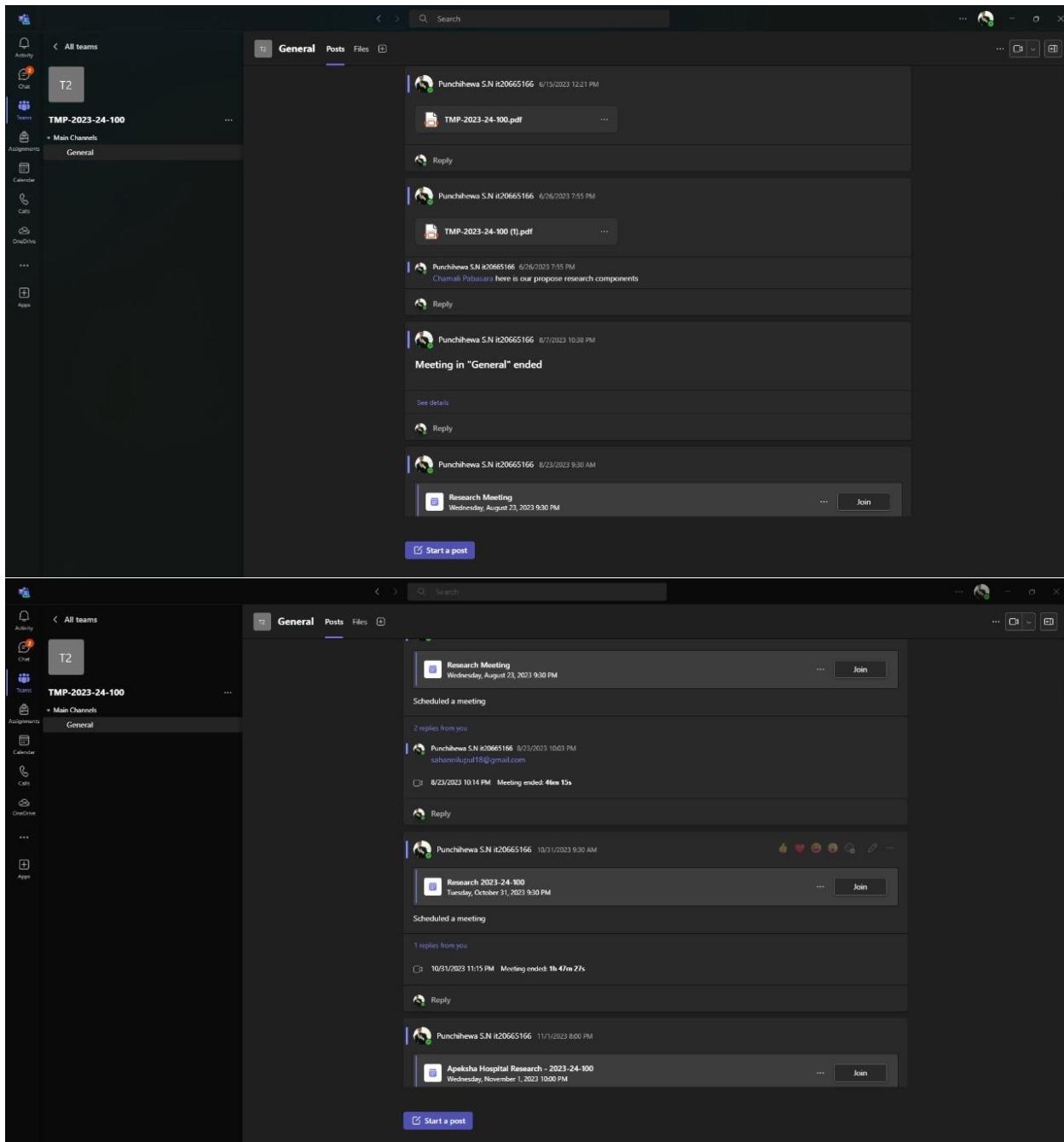
The main goal is to make a system that is easy for anyone to use and that takes into account both the needs of the hospital and the interests of doctors. The purpose of the research is to set an example for other hospitals to follow. The system will help to track all important issues like getting donors involved and keeping track of them. The study's results will eventually help us figure out how to get more donors, make better use of resources, and make critical item donations in healthcare settings more efficient overall [2].

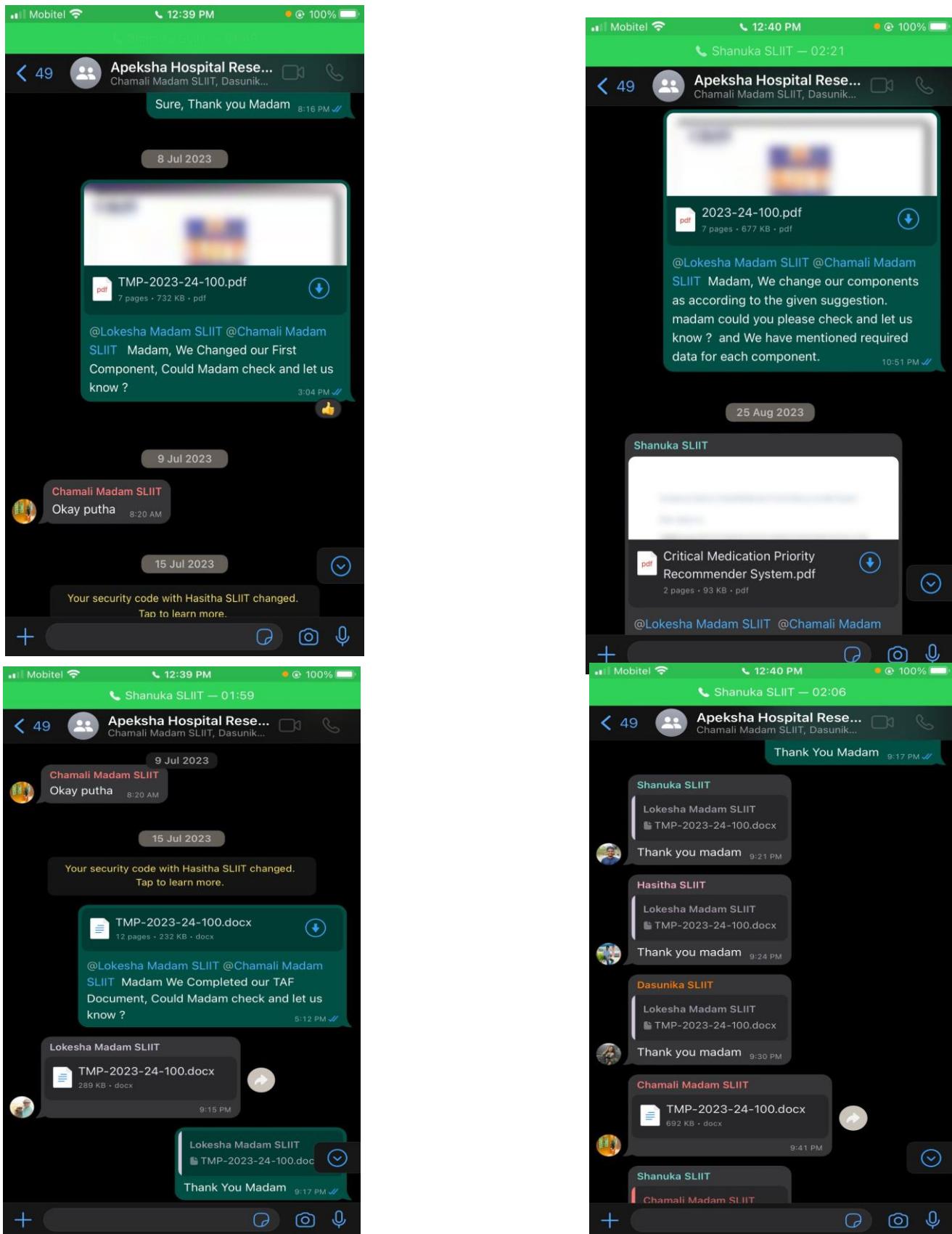
In the dynamic landscape of healthcare, the availability of essential medications stands as a cornerstone for effective patient care and optimal outcomes. However, the procurement and delivery of critical medications presents a formidable challenge, risking compromised treatments and adverse patient outcomes. Apeksha Hospital, among others, grapples with the task of identifying and prioritizing medications prone to shortages.

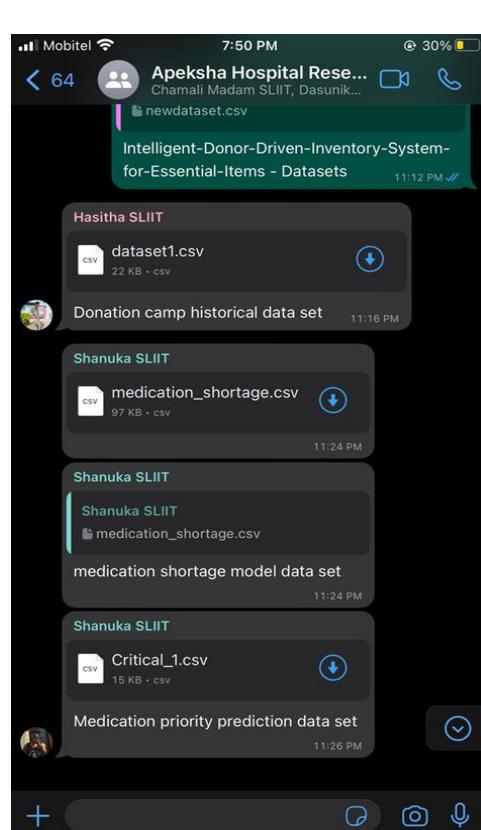
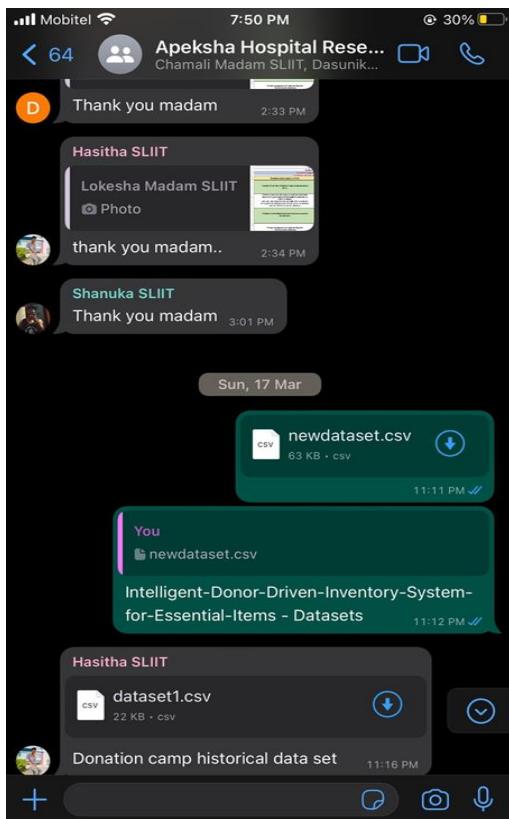
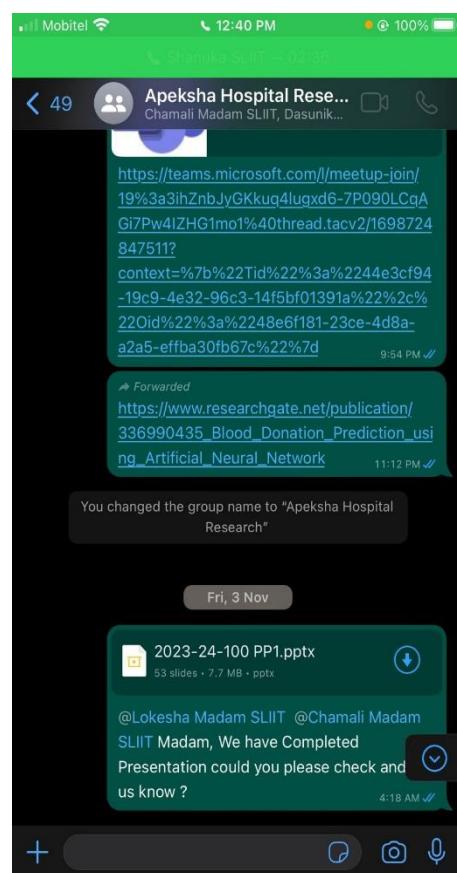
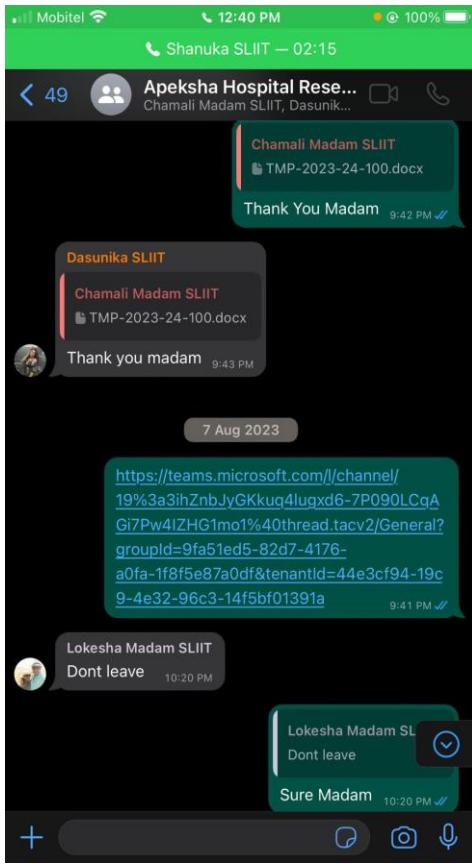
This research introduces a pioneering solution—the 'Critical Medication Prioritization and Resource System' (CMPRS). By integrating advanced machine learning techniques and predictive analytics, this system aims to proactively address medication shortages and optimize allocation strategies. Building upon previous studies such as Rhodes(2016) [3] and Vest(2021) [4], which emphasize the significance of predictive models in identifying essential medications for patient care, this research endeavor to develop an intelligent solution specific for Apeksha Hospital.

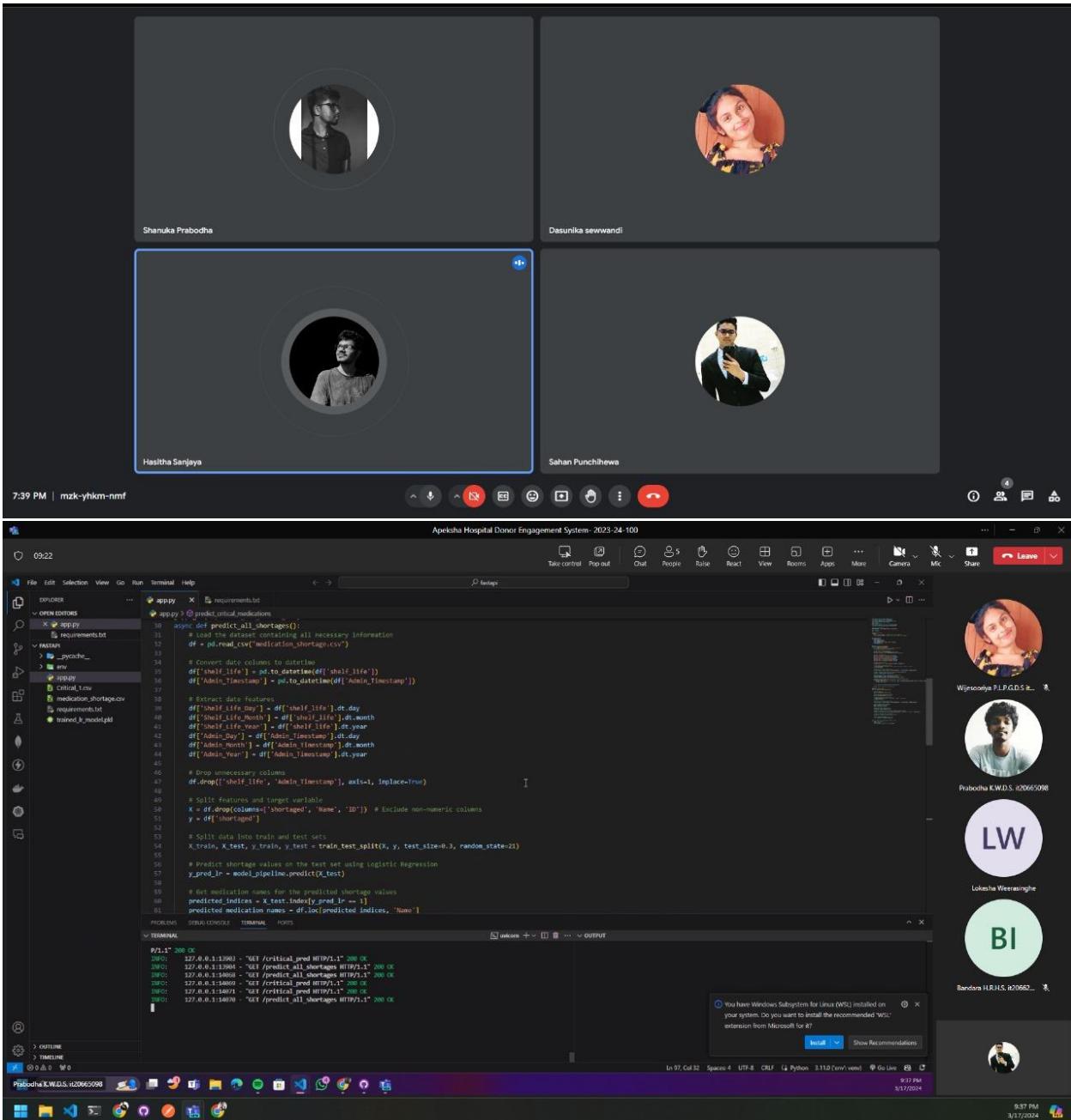
The aim is to create a system predicting shortages, identifying vital medications as per patient needs, necessitating compatibility with donations, and optimizing recommendations based on factors like importance, availability, and expiration. Leveraging historical data, the system empowers Apeksha Hospital to efficiently manage medication resources for timely and effective patient care.

5. Screenshots of meetings and conversations with supervisor

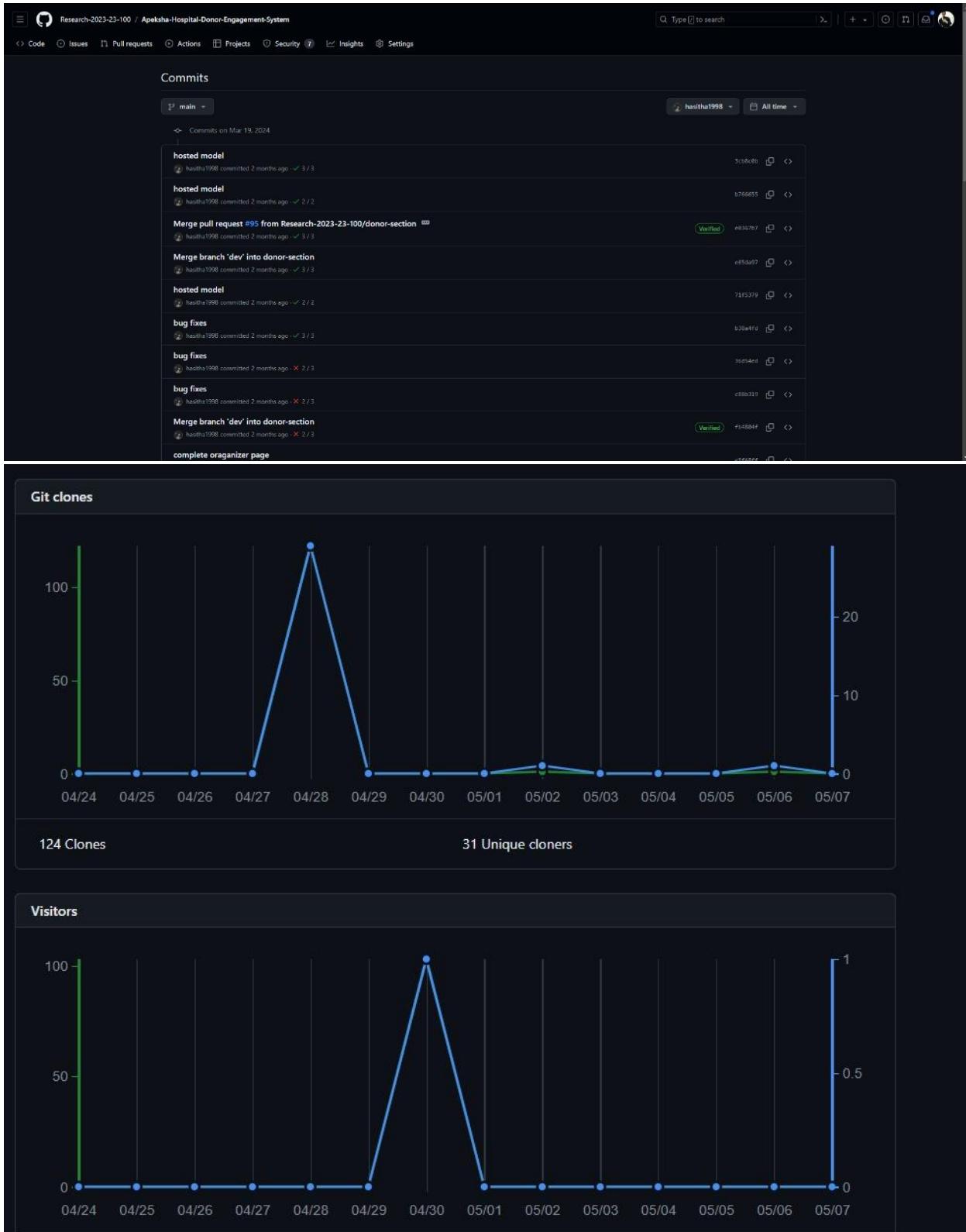


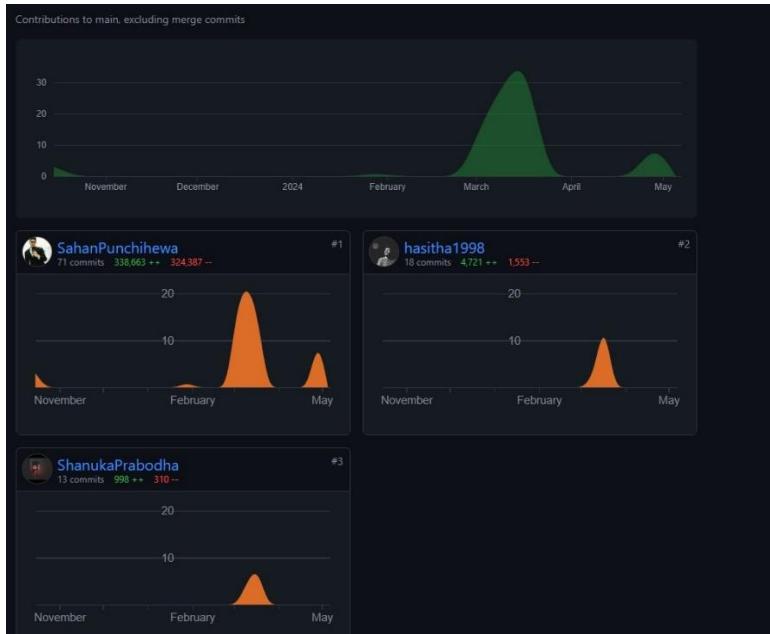






6.Version Controller and Contribution





7. Project Management Tools & Screenshot

7.1 Personal Backlog

New

+ New item

Active

6/5

Resolved

23/5

Closed

Category	Item ID	Description	State
Active	76	Implementing the system integration with other members components	Hasitha Sanjaya State: Active
Active	77	Fixing bugs frontend and backend connection	Hasitha Sanjaya State: Active
Resolved	74	Deploy the new model in Render platform	Hasitha Sanjaya State: Resolved
Resolved	75	Implementing out-side Organizer Management part	Hasitha Sanjaya State: Resolved
Resolved	71	Implementing the new Attendees prediction model using Random Forest Algorithm.	Hasitha Sanjaya State: Resolved
Closed	32	Implement the Prediction Dashboard User Interface.	Hasitha Sanjaya State: Closed
Closed	31	Implementing the Attendees Prediction model using Random Forest Algorithm.	Hasitha Sanjaya State: Closed
Closed	23	Pre-process the test Data set	Hasitha Sanjaya State: Closed

New

Active

Resolved

Closed

Category	Item ID	Description	State
Resolved	73	Implementing Attendees Prediction amount display UI	Hasitha Sanjaya State: Resolved
Resolved	66	Creating model using dataset from apeksha hospital to feed the model	Hasitha Sanjaya State: Resolved
Resolved	69	Collecting Real Donation Campaign historical data from Apeksa Hospital	Hasitha Sanjaya State: Resolved
Resolved	15	Collecting Test Data - Previous Donation campaign attendees count and date	Hasitha Sanjaya State: Resolved
Closed	26	Creating test data set to feed the model	Hasitha Sanjaya State: Closed
Closed	18	Pre-Process the new Dataset	Hasitha Sanjaya State: Closed
Closed	15	Collecting Test Data - Previous Donation campaign attendees count and date	Hasitha Sanjaya State: Closed

7.2Overall backlog

New	Active	Resolved	Closed
	58. System Deployment(hair Donation) IT20660352 Wijesooriya PLP.GDS	74. Deploy the new model in Render platform IT20660352 Wijesooriya PLP.GDS	32. Implement the Prediction Dashboard User Interface Hasitha Sanjaya
	59. Hair Donation Integration with the system IT20660352 Wijesooriya PLP.GDS	75. Implementing out-side Organizer Management part Hasitha Sanjaya	23. Pre-process the test Data set Hasitha Sanjaya
	67. UI Improvement in staff dashboard Sahan Punchihewa	71. Implementing the new Attendees prediction model using Random Forest Algorithm. Hasitha Sanjaya	26. Creating test data set to feed the model Hasitha Sanjaya
	70. create meaningful report for essential items usage Sahan Punchihewa	73. Implementing Attendees Prediction amount display UI Hasitha Sanjaya	60. Creating data sets using localized data set from Apeksha Hospital to feed the model Shanuka Prabodha
		66. Creating model using dataset from apeskha hospital to feed the model Hasitha Sanjaya	61. Pre-processing new data sets Shanuka Prabodha
		69. Pre-Process the new Dataset Hasitha Sanjaya	14. Collecting Real Medication Data from Apeksha Hospital Shanuka Prabodha

8.Trained New Model Screenshot

8.1Trained model python code

```
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import joblib

# Load the data
df = pd.read_csv('Final_encoded_data.csv')

# Split the data into features (X) and the target (y)
X = df[['Month', "Day_Type_poyaday", "Day_Type_public", "Day_Type_weekday", "Day_Type_weekend']]
y = df['Number_of_Attendees']

# Create a random forest regressor
rf_regressor = RandomForestRegressor(n_estimators=100)

# Train the random forest regressor on the entire dataset
rf_regressor.fit(X, y)

# Save the trained model to a joblib file
joblib.dump(rf_regressor, 'random_forest_model.joblib')

# Get user input for prediction
user_month = int(input("Enter the month (1-12): "))
user_day_type_poyaday = int(input("Is it a Poyaday? (0 or 1): "))
user_day_type_public = int(input("Is it a public holiday? (0 or 1): "))
user_day_type_weekday = int(input("Is it a weekday? (0 or 1): "))
user_day_type_weekend = int(input("Is it a weekend? (0 or 1): "))

user_input = pd.DataFrame({
    "Month": [user_month],
    "Day_Type_poyaday": [user_day_type_poyaday],
    "Day_Type_public": [user_day_type_public],
    "Day_Type_weekday": [user_day_type_weekday],
    "Day_Type_weekend": [user_day_type_weekend]
})
```

8.2 Trained model accuracy

```
Enter the month (1-12): 2
Is it a Poyaday? (0 or 1): 0
Is it a public holiday? (0 or 1): 1
Is it a weekday? (0 or 1): 0
Is it a weekend? (0 or 1): 0
Predicted Number of Attendees: 230.52119047619047
R-squared (R2) on the entire dataset: 0.9294346305592395
```

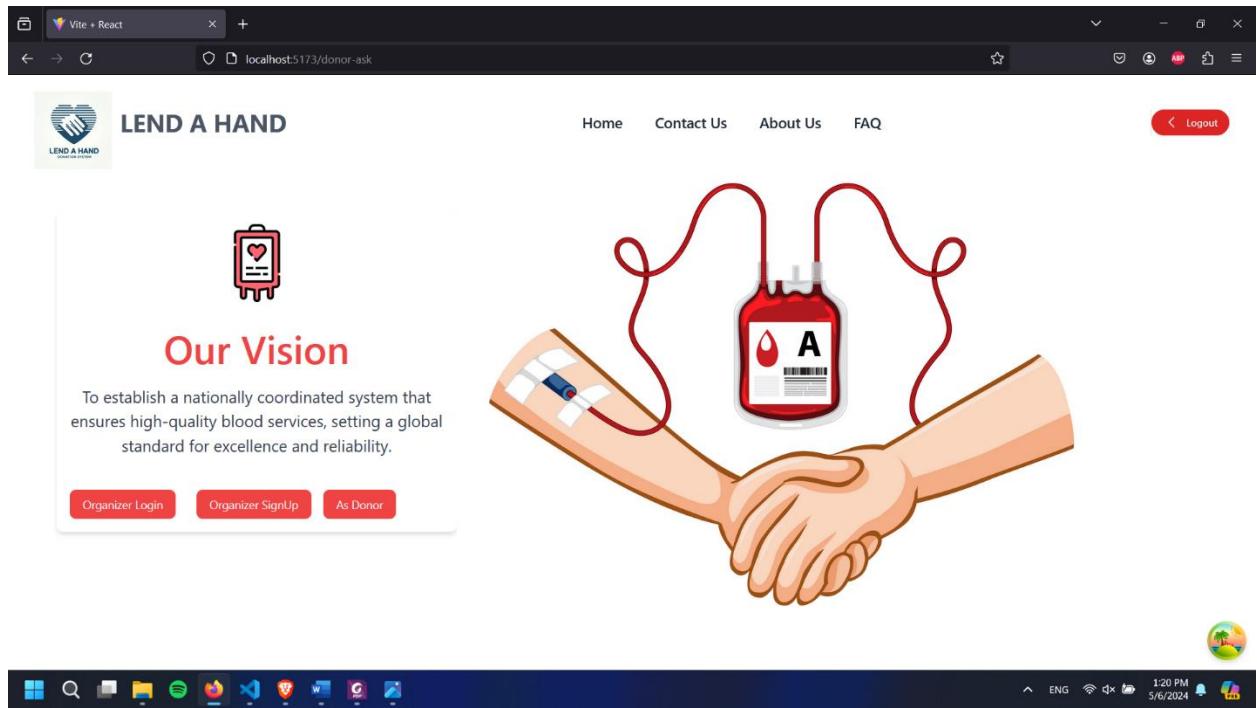
8.3 Hosted model in Render

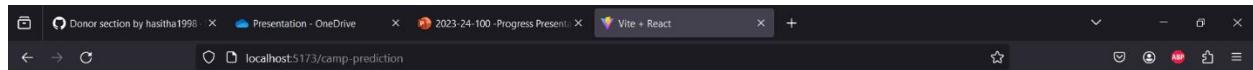
The screenshot shows the Render web interface for a service named 'MLModel-4'. At the top, it displays the service name, Python version (Python 3), and a 'Free' badge. There are buttons for 'Connect' and 'Manual Deploy'. Below this, it shows the repository information ('hasitha1998 / MLModel') and the main file ('main'). A link to the deployed URL ('https://mlmodel-4.onrender.com') is also present.

On the left, a sidebar lists various monitoring and management sections: Events, Logs (which is selected), Disks, Environment, Shell, Previews, Jobs, Metrics, Scaling, and Settings. The main area is a log viewer titled 'All logs'. It includes a search bar, a 'Live tail' button, and a timestamp selector set to 'GMT+5:30'. The log output shows the following entries:

```
May 6 12:16:11 PM 0 ==> Using Node version 20.11.1 (default)
May 6 12:16:11 PM 0 ==> Docs on specifying a Node version: https://render.com/docs/node-version
May 6 12:16:18 PM 0 ==> Running 'unicorn main:app'
May 6 12:16:25 PM 0 [2024-05-06 06:46:25 +0000] [41] [INFO] Starting unicorn 21.2.0
May 6 12:16:25 PM 0 [2024-05-06 06:46:25 +0000] [41] [INFO] Listening at: http://0.0.0.0:10000 (41)
May 6 12:16:25 PM 0 [2024-05-06 06:46:25 +0000] [41] [INFO] Using worker: sync
May 6 12:16:25 PM 0 [2024-05-06 06:46:25 +0000] [57] [INFO] Booting worker with pid: 57
May 6 12:16:28 PM 0 127.0.0.1 - - [06/May/2024:06:46:28 +0000] "GET / HTTP/1.1" 404 207 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:125.0) Gecko/20100101 Firefox/125.0"
```

9. Web Implementation





LEND A HAND

Attendance Prediction

Month: February

Day Type: Public

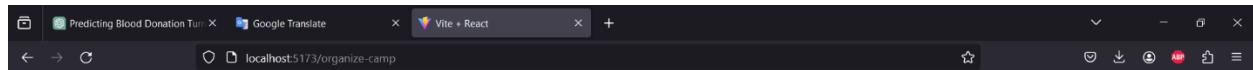
Predict Organize Camp Refresh

Predicted Number of Attendees
202



PLACE	DATE	PEOPLE AMOUNT	PERMISSION LETTER	STAFF	REQUIRED ITEMS	CAMP STATUS
Apeksha Hospital	3/19/2024	125		0 <button>Update</button>	0 <button>Update</button>	Pending <button>Pending</button>
maharagama temple	3/28/2024	351		10 <button>Update</button>	350 <button>Update</button>	Approved <button>Approved</button>
Apeksha hospital	4/30/2024	250		0 <button>Update</button>	0 <button>Update</button>	Pending <button>Pending</button>
Presidents collage Maharagama	5/14/2024	251		0 <button>Update</button>	0 <button>Update</button>	Pending <button>Pending</button> Approved <button>Approved</button> Rejected <button>Rejected</button>

1 2



LEND A HAND
DONATION SYSTEM

Create Donation Camp

Organizer Name: Mobile:

Email: Place:

Date: Expected People Amount:

Upload Permission Letter: No file selected.

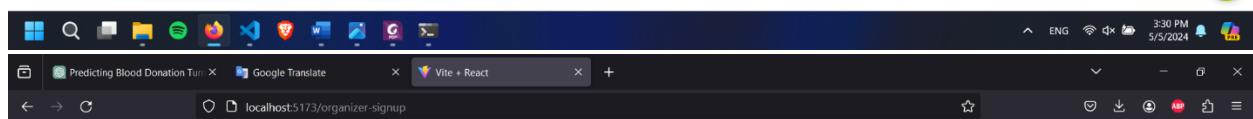


Blood Donation Camp Organizer Notice

If you need to organize a blood donation camp, you may get permission from relevant authority.

submit a permission letter from the relevant authority of the location where you will hold the blood campaign.

As an example you gonna organize donation camp in the school main hall you need to get requested letter from principle.



LEND A HAND

Home Contact Us About Us FAQ

User Sign Up

Organizer Name Mobile

Email Address Password

NIC Front Image



NIC Rear Image



Screenshot of a web application interface for "LEND A HAND" blood camp administration.

The top navigation bar includes links for Home, Contact Us, About Us, FAQ, Sign-Up, and Sign-In.

The main content area displays a large image of a Sri Lankan voter ID card with a portrait of a person and the number 980401146V.

The bottom section shows a dashboard for "All Blood Donation Camps" with three pie charts:

- Account Status Distribution:** Approved (Green), Rejected (Red), Pending (Yellow).
- Medical Staff Distribution:** Available Staff (Teal), Allocated Staff (Blue).
- Medical Items Distribution:** Available Items (Green), Allocated Items (Red). A callout box indicates 500 allocated items.

A table lists details for two camps:

ORGANIZER NAME	MOBILE	EMAIL	PLACE	DATE	PEOPLE AMOUNT	PERMISSION LETTER	STAFF	REQUIRED ITEMS	CAMP STATUS		
kamal	071123456	hasitha@gmail.com	Apeksha Hospital	3/20/2024	248		10	<button>Update</button>	250	<button>Update</button>	Pending
shanuka	0712352582	it20662028@my.sliit.lk	Apeksha Hospital	3/20/2024	250		150	<button>Update</button>	0	<button>Update</button>	Rejected

Apeksha Hospital Donor Engagement System

2023-24-100

Status Document 2

Wijesooriya PLPGDS

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Promoting Quality Hair Donation for Cancer Patients

2023-24-100

Status Document 2

Wijesooriya PLPGDS

BSc (Hons) in Information Technology specializing in Software Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2023

Declaration of The Candidate & Supervisor

I declare that this is my work, and this proposal does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Student ID	Signature
Wijesooriya PLPGDS	IT20660352	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor:

Date:

Signature of the Co-supervisor:

Date:

Contents

1. Introduction	IV
2. Gantt Chart.....	V
3. Work-Break-Down Chart	VI
4. Screenshot of Meetings and Conversations.....	VI
5. Project Management Tools & Screenshot.....	XII
5.1 Overall Backlog.....	XII
5.2 Personal Backlog	XIII
6. Trained Model Screenshot	XIII
6.1 Preprocessing for the image data sets	XIV
6.2 Model for the Hair Type Detection	XIV
6.3 Model for the Hair Color Detection.....	XVI
.....	XVII
6.4. Model for the Dandruff & Lice Detection	XIX
6.4.1. CNN Architecture Model- VGG16	XIX
7. Research Paper.....	XXII
8.Emails on Outlook	XXII
9.Implemented Interfaces of the Web application	XXIII
10.Testing	XXVIII
11.Version Control and Contribution	XXX
12.Snap Shot	XXXIV

1. Introduction

In the realm of cancer care, maintaining a sense of normalcy and confidence is paramount for patients undergoing treatment. One aspect often overlooked yet crucial for their well-being is the provision of high-quality wigs or hairpieces. However, the absence of a structured hair donation program at Apeksha Hospital poses a significant challenge in ensuring the availability of suitable hairpieces for cancer patients. To address this gap, it is imperative to establish a robust framework for assessing and promoting the quality of donated hair.

The research problem at hand revolves around determining whether donated hair meets the stringent standards and qualities mandated by Apeksha Hospital. This entails evaluating various characteristics of donated hair, including length, absence of split ends, and overall condition. Traditional methods of inspection may not suffice in ensuring the requisite quality standards. Therefore, leveraging cutting-edge image processing techniques emerges as a promising avenue to enhance the efficacy and accuracy of hair quality assessment.

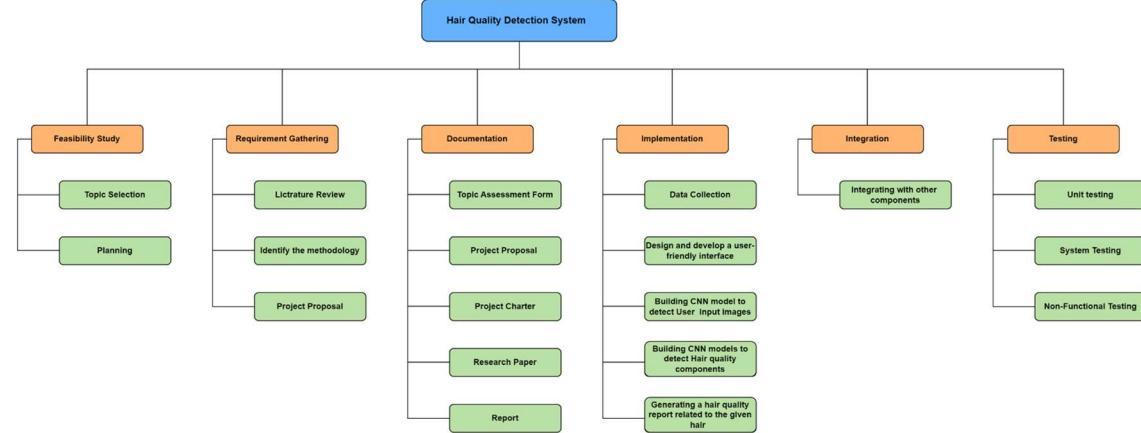
By employing image processing algorithms, donors can meticulously scrutinize donated hair for imperfections such as bleaching, dyeing, dryness, and split ends. This technology-driven approach not only streamlines the screening process but also minimizes the likelihood of substandard hair being utilized for crafting wigs or hairpieces. Moreover, promoting adherence to hospital standards fosters a culture of responsibility and accountability among potential donors, thereby augmenting the overall quality of donated hair.

In essence, this research endeavors to bridge the gap between hair donation and cancer patient care by establishing a comprehensive framework for evaluating and promoting the quality of donated hair. By integrating advanced image processing techniques with stringent quality criteria, Apeksha Hospital can ensure that cancer patients receive superior wigs or hairpieces, thereby enhancing their sense of confidence and well-being throughout the rigors of treatment.

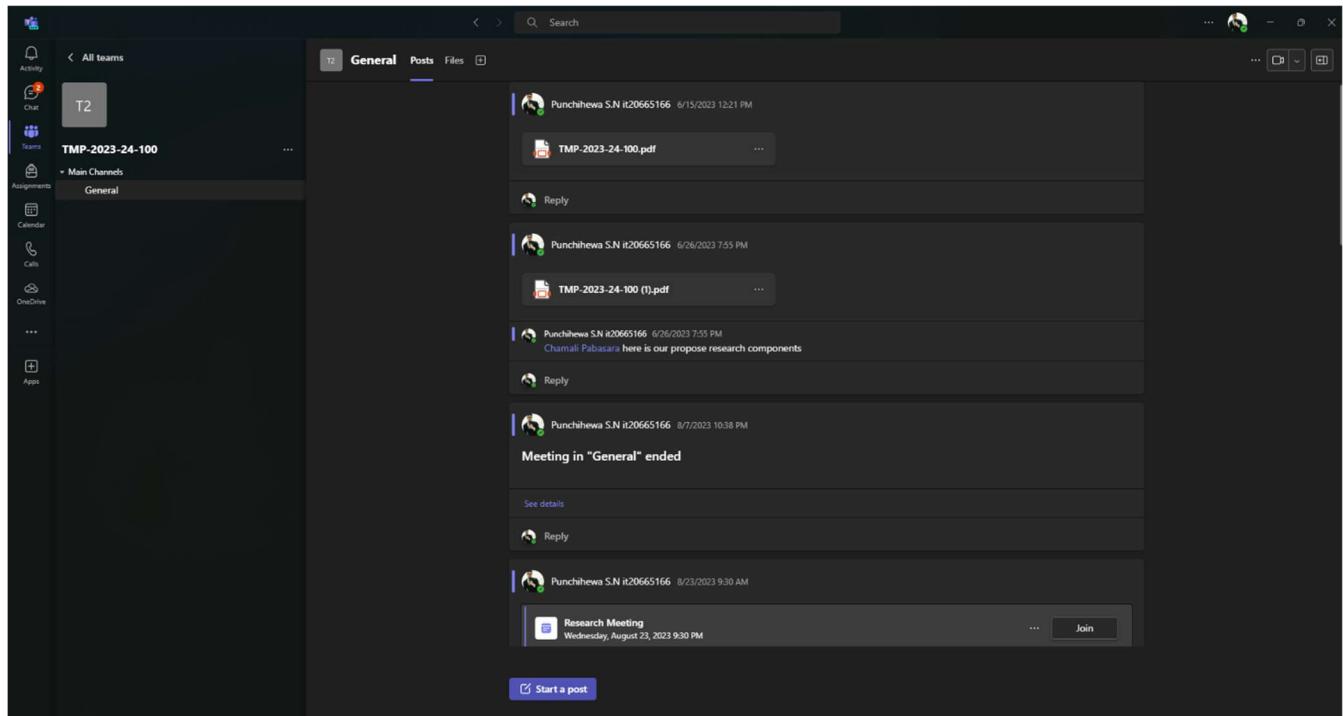
2. Gantt Chart

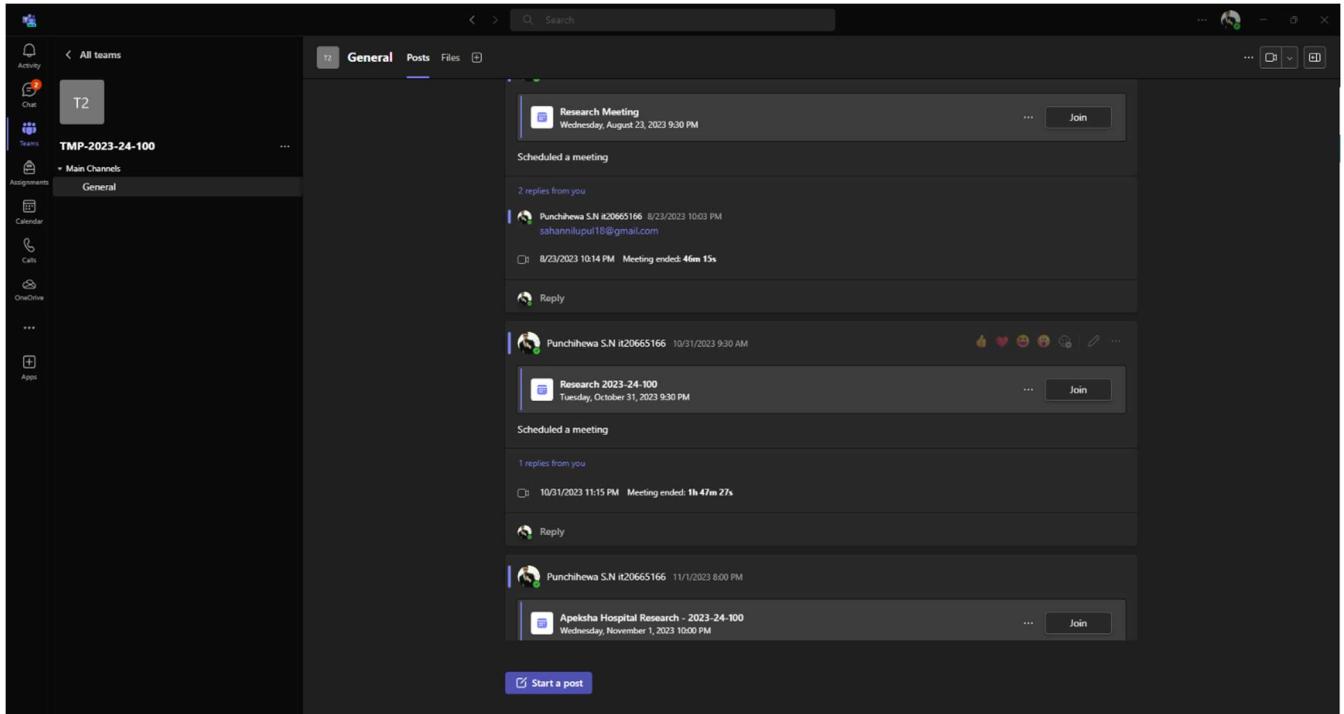
Task Name	Semester 1						Semester 2					
	June	July	August	September	October	November	December	January	February	March	April	May
Feasibility Study												
Topic Selection	■											
Topic Evaluation	■	■										
Background Study	■	■	■									
Background Study and Literature Survey												
Proposal Draft												
Project Proposal		■	■	■								
Proposal Presentation												
Proposal Report			■	■								
Project Initiation		■	■	■	■							
Data gathering			■	■	■							
Selecting technologies				■	■							
Project Implementation				■	■	■	■	■	■	■	■	
Data Collection					■	■	■	■	■	■	■	
Design and Develop a User friendly interface					■	■	■	■	■	■	■	
Building CNN Model to detect user input images					■	■	■	■	■	■	■	
Building CNN models to detect hair quality components					■	■	■	■	■	■	■	
generating a hair quality report related to the given hair					■	■	■	■	■	■	■	
Final Implementation					■	■	■	■	■	■	■	
Unit Testing						■	■	■	■	■	■	
Evaluating and error fixing						■	■	■	■	■	■	
Integrating with other components						■	■	■	■	■	■	
Final Stages						■	■	■	■	■	■	
System Integration						■	■	■	■	■	■	
System Testing						■	■	■	■	■	■	
Evaluating and error fixing						■	■	■	■	■	■	
Final Report						■	■	■	■	■	■	

3. Work-Break-Down Chart



4. Screenshot of Meetings and Conversations





Apexka Hospital Donor Engagement System- 2023-24-100

File Edit Selection View Go Run Terminal Help

EXPLORER app.py requirements.txt

```
app.py > predict_critical_medications
30     async def predict_all_shortages():
31         """A function for predicting all necessary information
32             df = pd.read_csv('medication_shortage.csv')
33
34             # Convert date columns to datetime
35             df['Shelf_Life'] = pd.to_datetime(df['Shelf_Life'])
36             df['Admin_Timestamp'] = pd.to_datetime(df['Admin_Timestamp'])
37
38             # Extract date features
39             df['Shelf_Life_Day'] = df['Shelf_Life'].dt.day
40             df['Shelf_Life_Month'] = df['Shelf_Life'].dt.month
41             df['Shelf_Life_Year'] = df['Shelf_Life'].dt.year
42             df['Admin_Day'] = df['Admin_Timestamp'].dt.day
43             df['Admin_Month'] = df['Admin_Timestamp'].dt.month
44             df['Admin_Year'] = df['Admin_Timestamp'].dt.year
45
46             # Drop unnecessary columns
47             df.drop(['shelf_life', 'Admin_Timestamp'], axis=1, inplace=True)
48
49             # Predict features and target variable
50             x = df.drop(columns=['shortaged', 'Name', 'ID']) # Exclude non-numeric columns
51             y = df['shortaged']
52
53             # Split data into train and test sets
54             X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=21)
55
56             # Predict shortage values on the test set using Logistic Regression
57             y_pred_lr = model_pipeline.predict(X_test)
58
59             # Get medication names for the predicted shortage values
60             predicted_indices = X_test[y_pred_lr == 1]
61             predicted_medication_names = df.loc[predicted_indices, 'Name']
```

TERMINAL

```
py3.10 202 OK
INFO: 127.0.0.1:13983 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13984 - "GET /predict_all_shortages HTTP/1.1" 200 OK
INFO: 127.0.0.1:13985 - "GET /predict_all_shortages HTTP/1.1" 200 OK
INFO: 127.0.0.1:13986 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13987 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13988 - "GET /predict_all_shortages HTTP/1.1" 200 OK
INFO: 127.0.0.1:13989 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13990 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13991 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13992 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13993 - "GET /critical_pred HTTP/1.1" 200 OK
INFO: 127.0.0.1:13994 - "GET /predict_all_shortages HTTP/1.1" 200 OK
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS

Chat People Raise React View Rooms Apps More Camera Mic Share

Leave

Wijesooriya P.J.P.G.D.S.i20665098

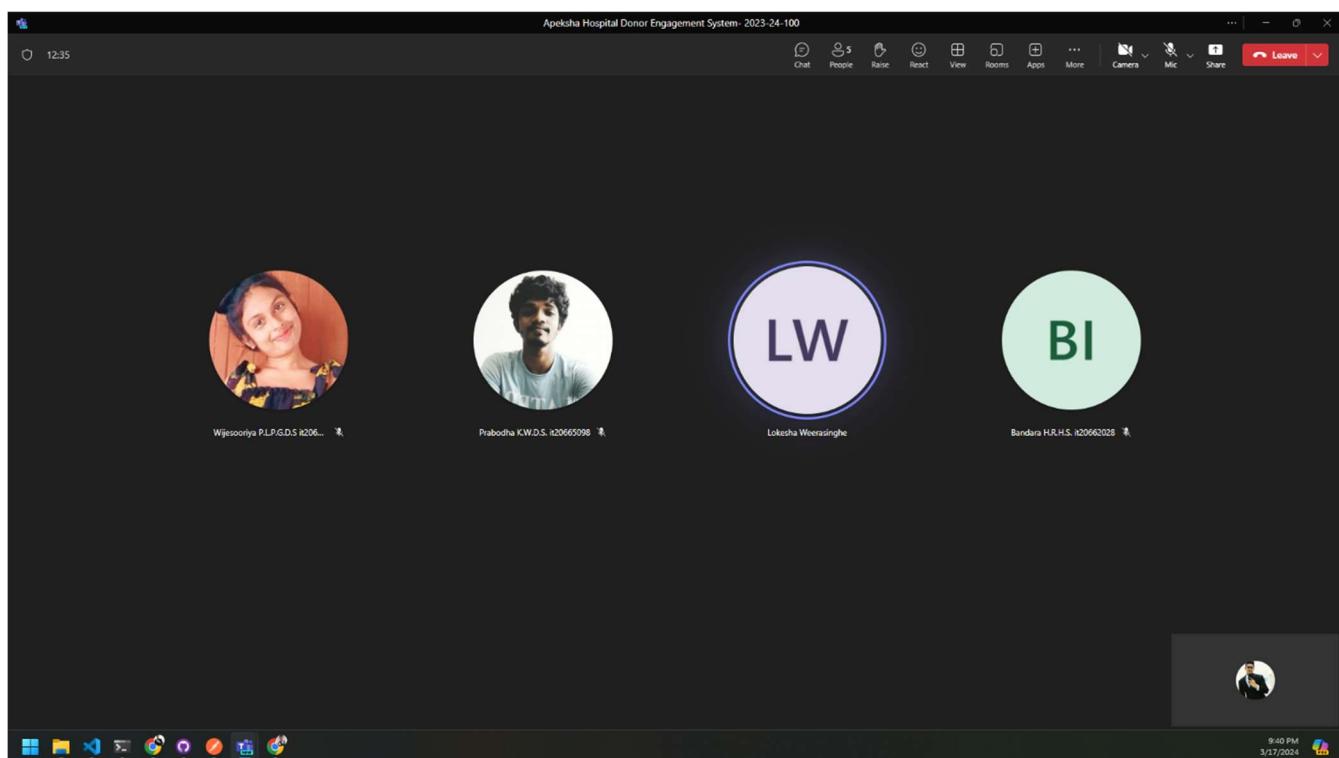
Praboda K.W.D.S.i20665098

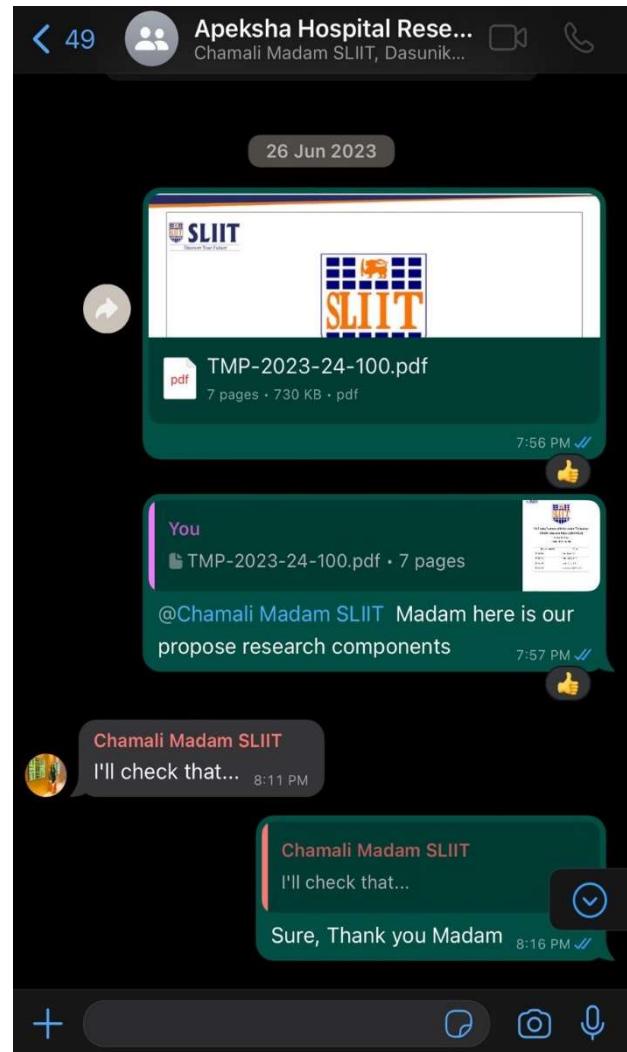
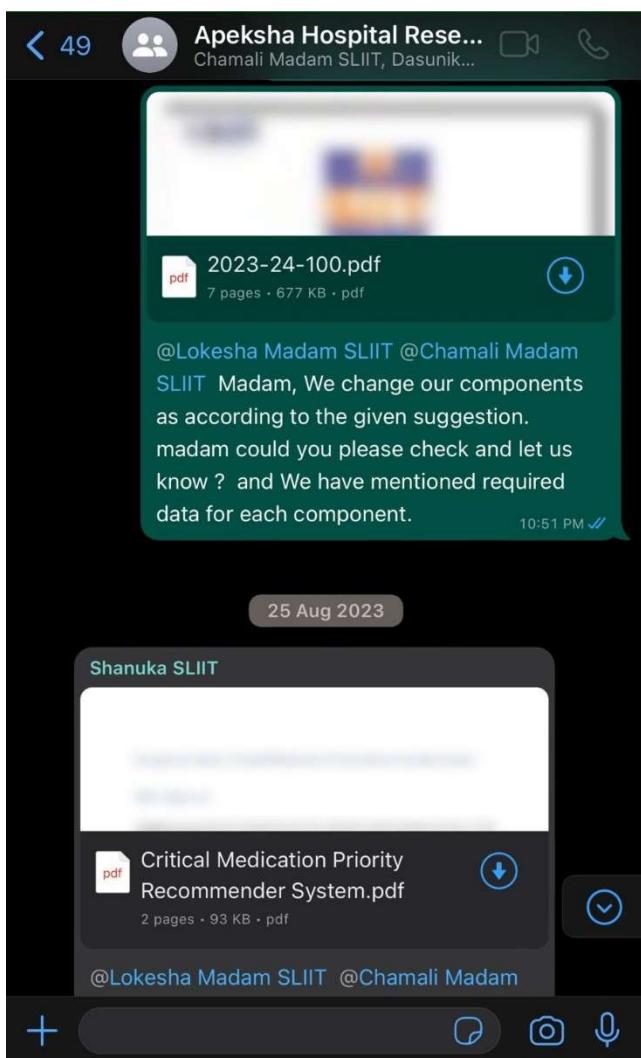
Lokesha Weerasinghe

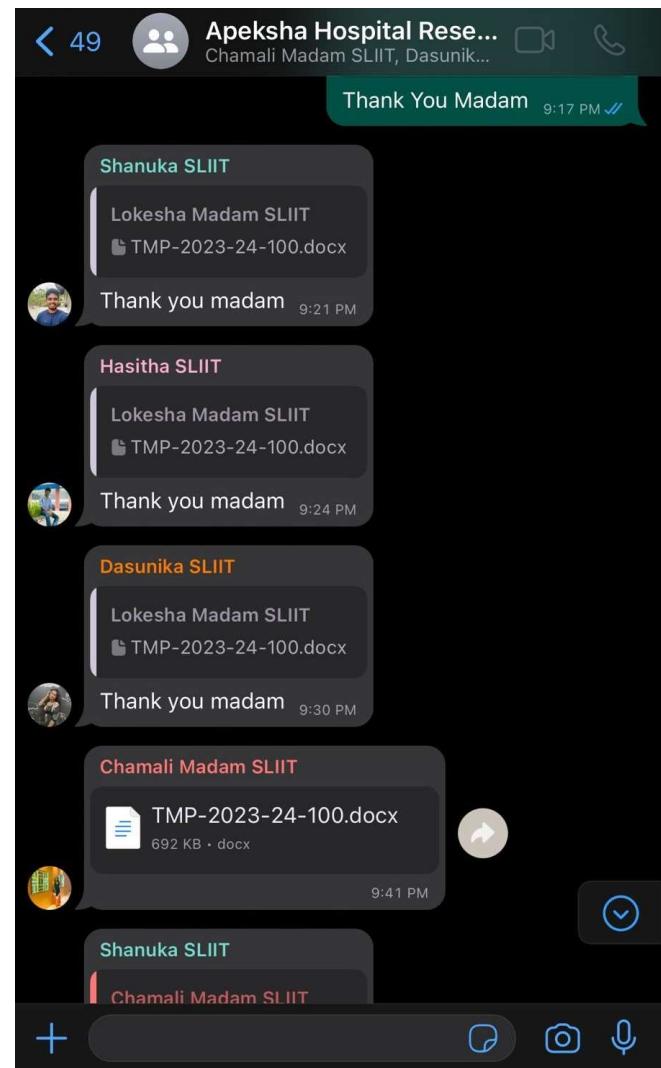
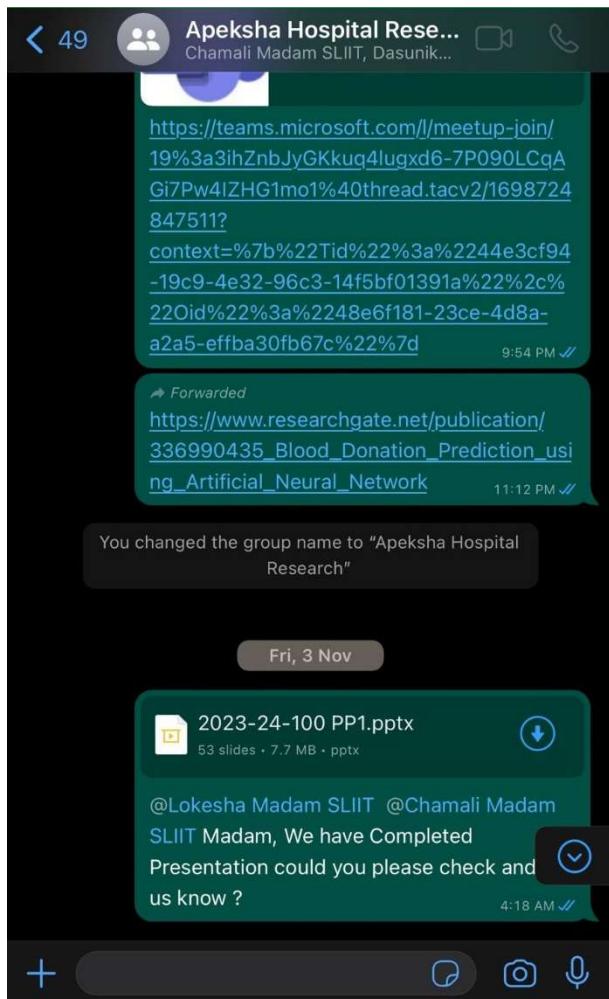
Bandara H.R.H.S.i206662028

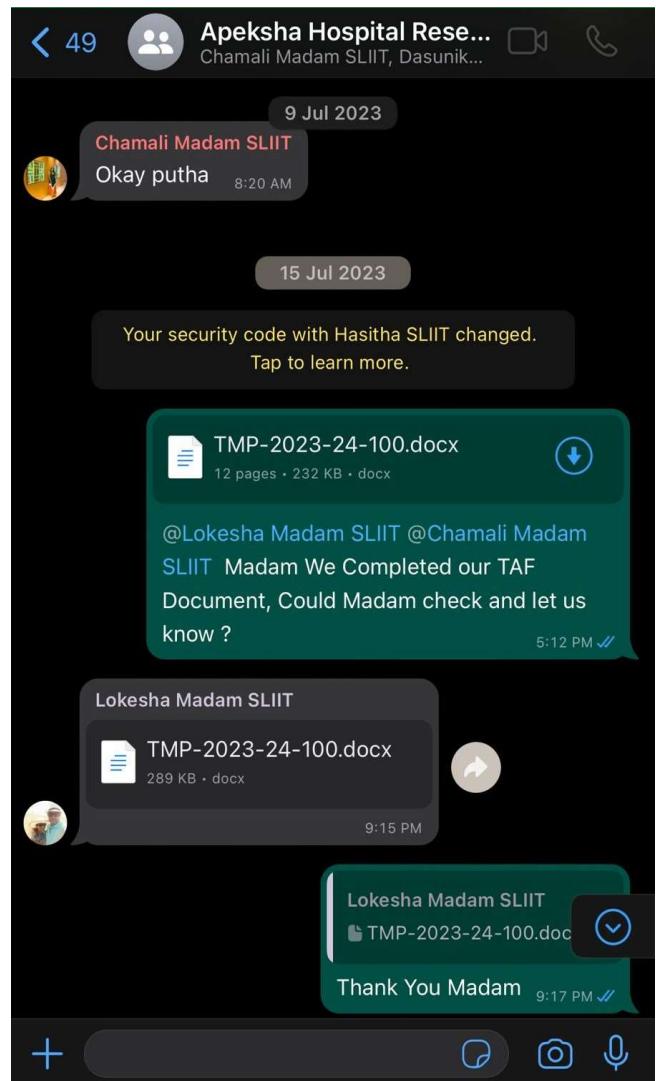
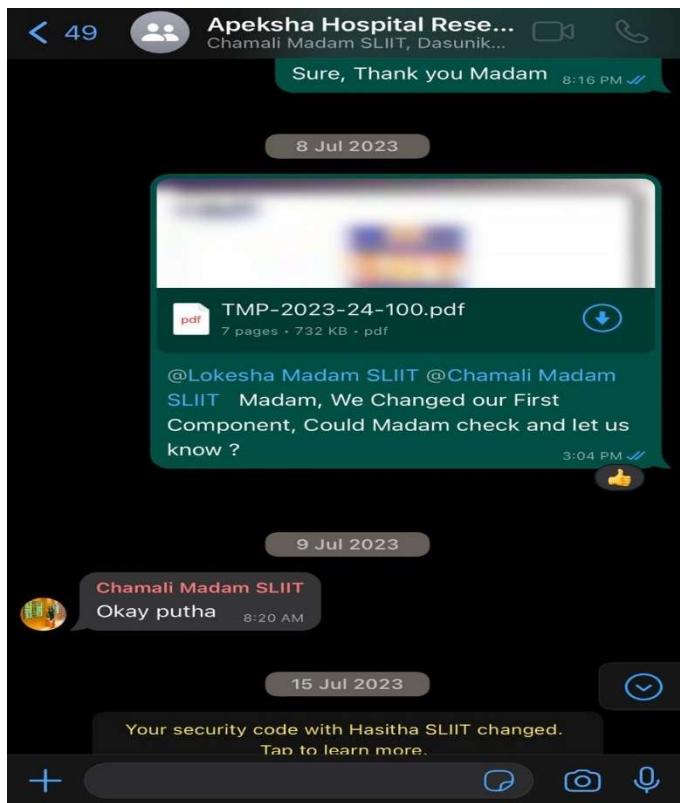
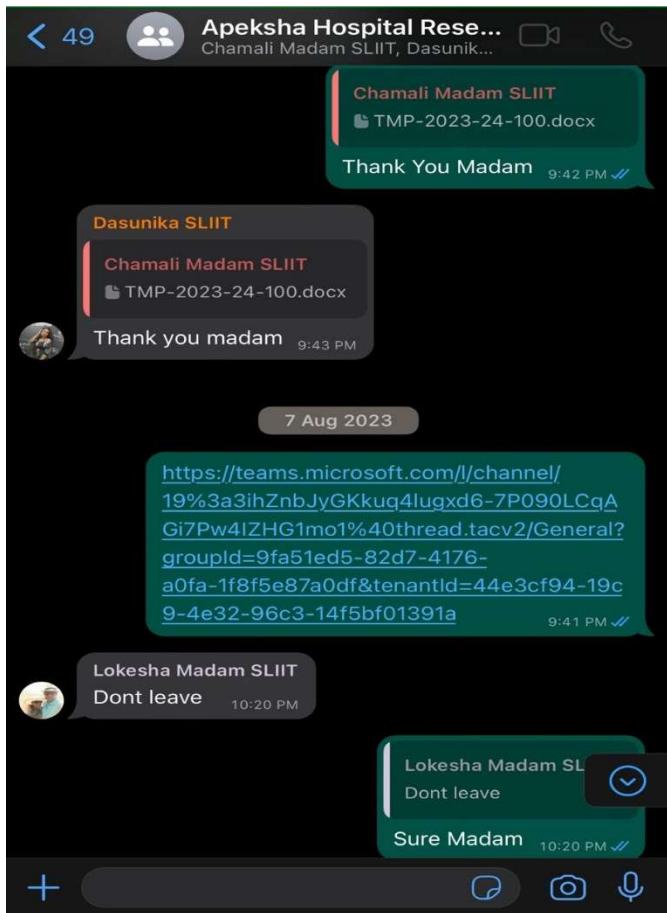
9:37 PM 3/17/2024

9:37 PM 3/17/2024









5. Project Management Tools & Screenshot

5.1 Overall Backlog

The screenshot shows the Azure DevOps interface for the 'Apeksha Hospital Donor Engagement System-2023-24-100 Team'. The left sidebar is collapsed, and the main area displays a backlog of work items. The backlog table has columns for Order, Work Item Type, Title, State, Effort, Business Area, and Tags. There are 19 items listed, all of which are Feature type work items. The titles describe various tasks related to system implementation and data processing.

Order	Work Item Type	Title	State	Effort	Business Area	Tags
1	Feature	Implementing the Critical Medication Identification model using R...	New		Business	
2	Feature	Deploy the new model in Render platform	New		Business	
3	Feature	Implementing Attendees Prediction amount display UI	New		Business	
4	Feature	Implementing Donor Interaction UI-Medication	New		Business	
5	Feature	Implementing the new Attendees prediction model using R...	New		Business	
6	Feature	Pre-Process the new dataset	New		Business	
7	Feature	Creating model using dataset from apexha hospital to fee...	New		Business	
8	Feature	System Deployment(hair Donation)	Active		Business	
9	Feature	Hair Donation Integration with the system	Active		Business	
10	Feature	UI Improvement in staff dashboard	Active		Business	
11	Feature	Create meaningful report for essential items usage	Active		Business	
12	Feature	Model Implemented to identify Color of Hair	Resolved		Business	
13	Feature	Creating test data set to feed the model	Resolved		Business	
14	Feature	Pre-process the Data set	Resolved		Business	
15	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business	
16	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business	
17	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business	
18	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business	
19	Feature	Deploy machine learning model in render server	Resolved		Business	

This screenshot shows the same Azure DevOps backlog as the previous one, but with a different set of 30 items. All items are Feature type work items, and they appear to be the same tasks as the first backlog, just in a different order or with some additional items. The interface and layout are identical to the first screenshot.

Order	Work Item Type	Title	State	Effort	Business Area	Tags
12	Feature	Model Implemented to identify Color of Hair	Resolved		Business	
13	Feature	Creating test data set to feed the model	Resolved		Business	
14	Feature	Pre-process the Data set	Resolved		Business	
15	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business	
16	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business	
17	Feature	Implementing the Critical Medication Identification model u...	Resolved		Business	
18	Feature	Implementing the Critical Medication Shortage Prediction ...	Resolved		Business	
19	Feature	Deploy machine learning model in render server	Resolved		Business	
20	Feature	Creating test data set to feed the model	Resolved		Business	
21	Feature	Pre-process the test Data set	Resolved		Business	
22	Feature	Implementing the Attendees Prediction model using Rando...	Resolved		Business	
23	Feature	Implement the Prediction Dashboard User Interface.	Resolved		Business	
24	Feature	Implement Donor-Driven Inventory System for Essential It...	Resolved		Business	
25	Feature	Implement machine learning model with localize dataset	Resolved		Business	
26	Feature	Implement Staff Dashboard	Resolved		Business	
27	Feature	Create a meaningful bar graph for essential items using a tr...	Resolved		Business	
28	Feature	Implement Machine Learning model using FastAPI	Resolved		Business	
29	Feature	Implementation of current stock of essential list and hospita...	Resolved		Business	
30	Feature	Implement items prediction dashboard	Resolved		Business	

Azure DevOps Boards view for the Apeksha Hospital Donor Engagement System. The backlog shows the following work items:

- New:**
 - 58 System Deployment(hair Donation) - Active
 - IT20660352 Wijesooriya P.L.P.G.D.S
- Active:**
 - 59 Hair Donation Integration with the system - Active
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - 67 UI Improvement in staff dashboard - Active
 - Sahan Punchihewa
 - 70 create meaningful report for essential items usage - Active
 - Sahan Punchihewa
- Resolved:**
 - 74 Deploy the new model in Render platform - Resolved
 - Hasitha Sanjaya
 - 75 Implementing out-side Organizer Management part - Resolved
 - Hasitha Sanjaya
 - 71 Implementing the new Attendees prediction model using Random Forest Algorithm. - Resolved
 - Hasitha Sanjaya
 - 73 Implementing Attendees Prediction amount display UI - Resolved
 - Hasitha Sanjaya
 - 66 Creating model using dataset from apeksha hospital to feed the model - Resolved
 - Hasitha Sanjaya
 - 69 Pre-Process the new Dataset - Resolved
- 24/s:**
 - 32 Implement the Prediction Dashboard User Interface. - Closed
 - Hasitha Sanjaya
 - 23 Pre-process the test Data set - Closed
 - Hasitha Sanjaya
 - 26 Creating test data set to feed the model - Closed
 - Hasitha Sanjaya
 - 60 Creating data sets using localized data set from Apeksha Hospital to feed the model - Closed
 - Shanuka Praboda
 - 61 Pre-processing new data sets - Closed
 - Shanuka Praboda
 - 14 Collecting Real Medication Data from Apeksha Hospital - Closed
 - Shanuka Praboda
- Closed:**
 - 32 Implement the Prediction Dashboard User Interface. - Closed
 - Hasitha Sanjaya
 - 23 Pre-process the test Data set - Closed
 - Hasitha Sanjaya
 - 26 Creating test data set to feed the model - Closed
 - Hasitha Sanjaya
 - 60 Creating data sets using localized data set from Apeksha Hospital to feed the model - Closed
 - Shanuka Praboda
 - 61 Pre-processing new data sets - Closed
 - Shanuka Praboda
 - 14 Collecting Real Medication Data from Apeksha Hospital - Closed
 - Shanuka Praboda

5.2 Personal Backlog

Azure DevOps Boards view for the Apeksha Hospital Donor Engagement System. The backlog shows the following work items:

- New:**
 - + New item
- Active:**
 - 58 System Deployment(hair Donation) - Active
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
 - 59 Hair Donation Integration with the system - Active
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
- Resolved:**
 - 44 Combine all the models to implement the final model - Resolved
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
 - 49 Identification of best architecture for transfer learning - Resolved
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
- 23/s:**
 - 50 Data Pre-Processing and Create Data Set With Data Augmentation - Resolved
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
 - 52 Model Implemented to identify length type of Hair - Resolved
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
- Closed:**
 - 49 Identification of best architecture for transfer learning - Closed
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
 - 50 Data Pre-Processing and Create Data Set With Data Augmentation - Closed
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State
 - 52 Model Implemented to identify length type of Hair - Closed
 - IT20660352 Wijesooriya P.L.P.G.D.S
 - State

6. Trained Model Screenshot

6.1 Preprocessing for the image data sets

```
In [4]: import cv2
import imghdr

In [6]: image_exts = ['jpeg', 'jpg', 'bmp', 'png']

In [7]: data_dir='C:/Users/Ridma/Downloads/Hairs'
for image_class in os.listdir(data_dir):
    for image in os.listdir(os.path.join(data_dir, image_class)):
        image_path = os.path.join(data_dir, image_class, image)
        try:
            img = cv2.imread(image_path)
            tip = imghdr.what(image_path)
            if tip not in image_exts:
                print('Image not in ext list {}'.format(image_path))
                os.remove(image_path)
        except Exception as e:
            print('Issue with image {}'.format(image_path))
            # os.remove(image_path)

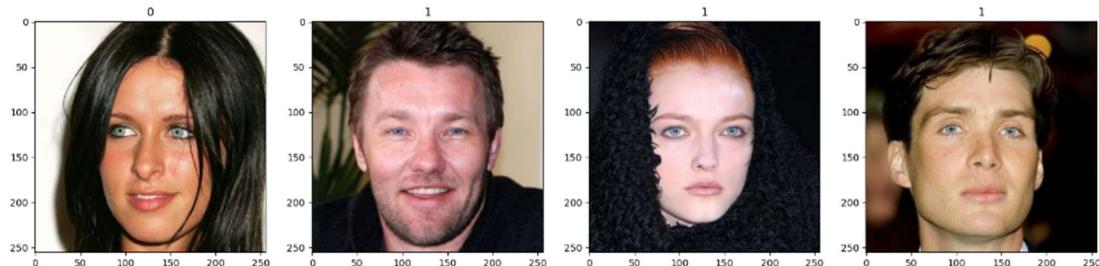
In [ ]: datagen = preprocessing.image.ImageDataGenerator(
    rescale=1./255,
    validation_split=0.2
```

6.2 Model for the Hair Type Detection

```
In [16]: data_iterator = data.as_numpy_iterator()

In [17]: batch = data_iterator.next()

In [18]: fig, ax = plt.subplots(ncols=4, figsize=(20,20))
for idx, img in enumerate(batch[0][:4]):
    ax[idx].imshow(img.astype(int))
    ax[idx].title.set_text(batch[1][idx])
```



Proposed Model

6.2.1. CNN Architecture Model- VGG1

```

In [4]: train_generator = datagen.flow_from_directory(
    'C:/Users/Ridma/Downloads/Hairs',
    target_size=img_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)
Found 858 images belonging to 2 classes.

In [5]: val_generator = datagen.flow_from_directory(
    'C:/Users/Ridma/Downloads/Hairs',
    target_size=img_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)
Found 213 images belonging to 2 classes.

In [6]: from tensorflow.keras.callbacks import EarlyStopping
early_stop = EarlyStopping(monitor='val_loss', patience=10, verbose=1, restore_best_weights=True)

In [7]: base_model = VGG16(input_shape=(160, 160, 3), include_top=False, weights='imagenet')
base_model.trainable = False

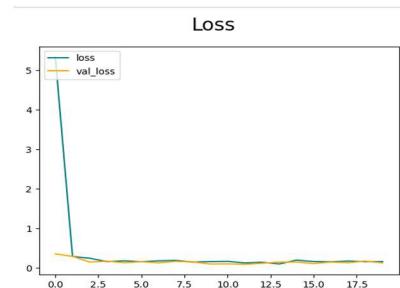
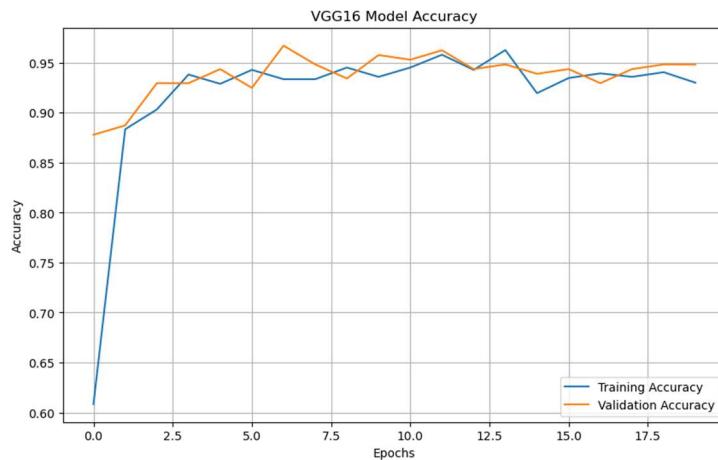
vgg16_model = models.Sequential([
    base_model,
    layers.Flatten(),
    layers.Dense(4096, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(4096, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(2, activation='softmax')
])
vgg16_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

history=vgg16_model.fit(train_generator, validation_data=val_generator, epochs=20 , callbacks=[early_stop])

Epoch 1/20
27/27 [=====] - 163s/step - loss: 5.3365 - accuracy: 0.6084 - val_loss: 0.3546 - val_accuracy: 0.87

```

Evidence of Completion



Loss: 0.2021

Accuracy: 0.9345

6.2.2. CNN Architecture Model- MobileNetV2

```

In [38]: train_generator = datagen.flow_from_directory(
    'C:/Users/Ridma/Downloads/Hairs',
    target_size=img_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)
Found 858 images belonging to 2 classes.

In [39]: val_generator = datagen.flow_from_directory(
    'C:/Users/Ridma/Downloads/Hairs',
    target_size=img_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)
Found 213 images belonging to 2 classes.

In [40]: from tensorflow.keras.callbacks import EarlyStopping
early_stop = EarlyStopping(monitor='val_loss', patience=10, verbose=1, restore_best_weights=True)

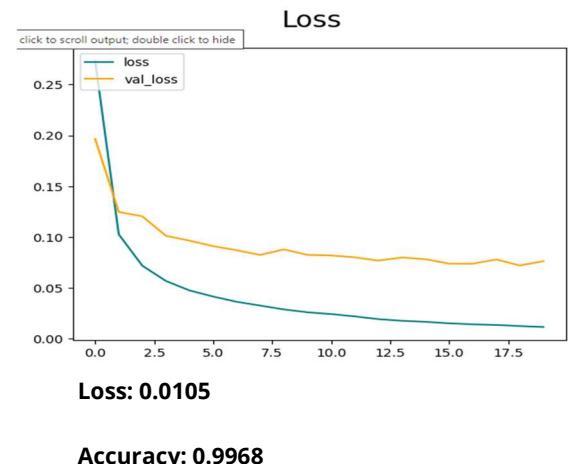
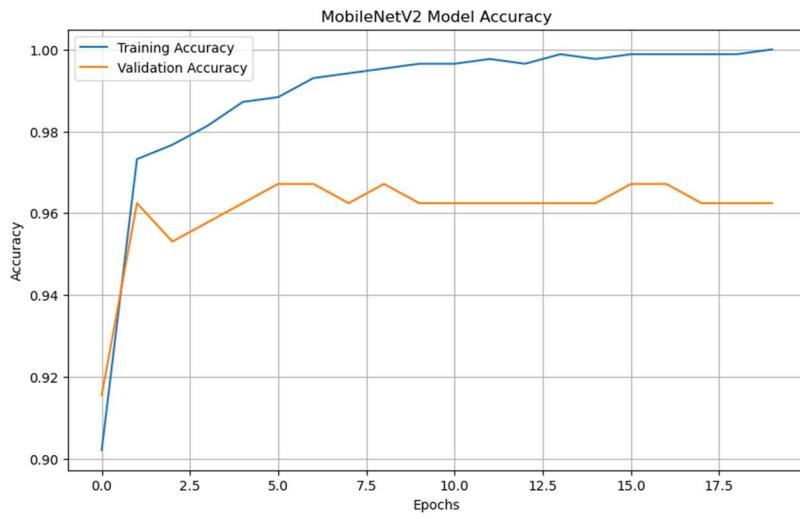
In [10]: base_model = MobileNetV2(input_shape=(160, 160, 3), include_top=False, weights='imagenet')
base_model.trainable = False

mobilenet_model = models.Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),
    layers.Dense(2, activation='softmax')
])

mobilenet_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
history=mobilenet_model.fit(train_generator, validation_data=val_generator, epochs=20 ,callbacks=[early_stop])
Epoch 1/20
14/14 [=====] - 26s 1s/step - loss: 0.2733 - accuracy: 0.9021 - val_loss: 0.1966 - val_accuracy: 0.915
5
Epoch 2/20
14/14 [=====] - 16s 1s/step - loss: 0.1027 - accuracy: 0.9732 - val_loss: 0.1247 - val_accuracy: 0.962
4

```

Evidence of Completion



6.3 Model for the Hair Color Detection

6.3.1. CNN Architecture Model- VGG16

```

In [6]: train_generator = datagen.flow_from_directory(
    'C:/Users/Ridma/Downloads/newimages',
    target_size=img_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)
Found 792 images belonging to 2 classes.

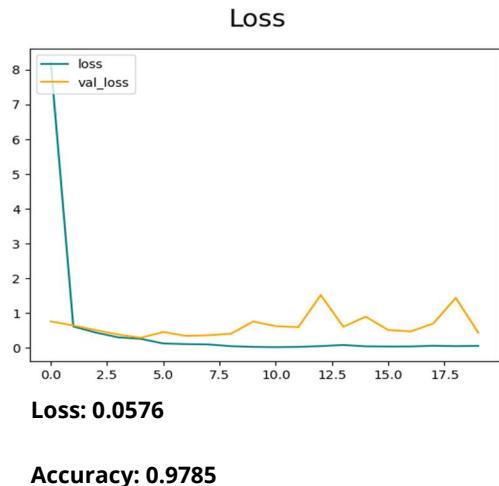
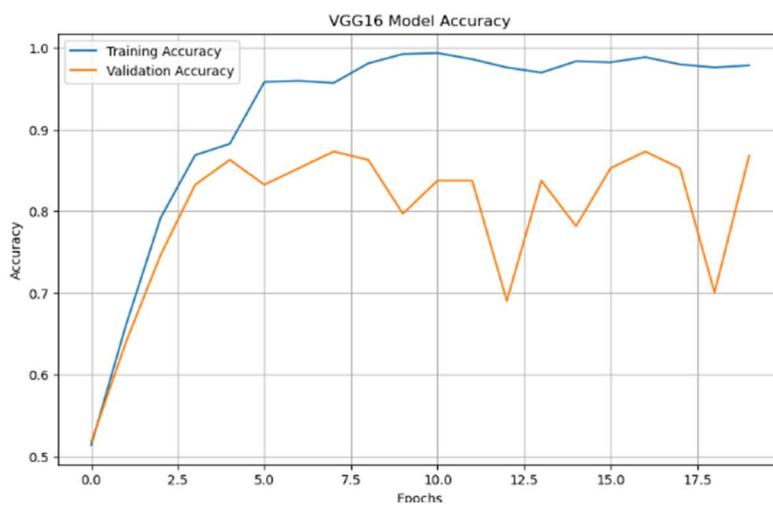
In [7]: val_generator = datagen.flow_from_directory(
    'C:/Users/Ridma/Downloads/newimages',
    target_size=img_size,
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)
Found 197 images belonging to 2 classes.

In [8]: base_model = VGG16(input_shape=(160, 160, 3), include_top=False, weights='imagenet')
base_model.trainable = False

vgg16_model = models.Sequential([
    base_model,
    layers.Flatten(),
    layers.Dense(4096, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(4096, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(2, activation='softmax') # Assuming 2 hair colors
])
vgg16_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
history=vgg16_model.fit(train_generator, validation_data=val_generator, epochs=20)

Epoch 1/20
13/13 [=====] - 209s 16s/step - loss: 8.1964 - accuracy: 0.5139 - val_loss: 0.7592 - val_accuracy: 0.5
...

```



6.3.2. CNN Architecture Model- MobileNetV2

```
In [13]: train_generator = datagen.flow_from_directory(  
    'C:/Users/Ridma/Downloads/newimages',  
    target_size=img_size,  
    batch_size=batch_size,  
    class_mode='categorical',  
    subset='training'  
)  
Found 792 images belonging to 2 classes.
```

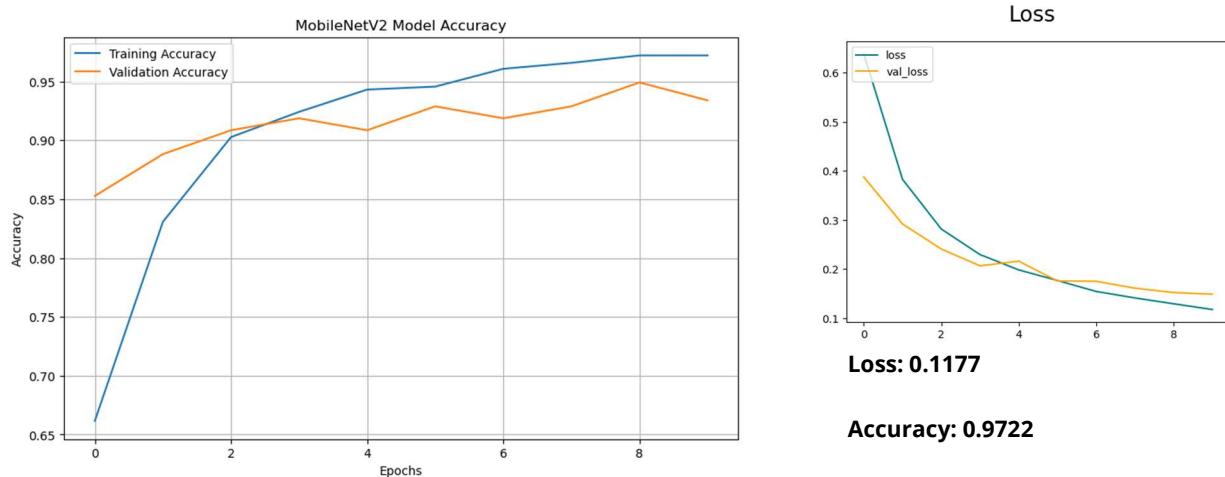


```
In [14]: val_generator = datagen.flow_from_directory(  
    'C:/Users/Ridma/Downloads/newimages',  
    target_size=img_size,  
    batch_size=batch_size,  
    class_mode='categorical',  
    subset='validation'  
)  
Found 197 images belonging to 2 classes.
```



```
In [15]: base_model = MobileNetV2(input_shape=(160, 160, 3), include_top=False, weights='imagenet')  
base_model.trainable = False  
  
mobilenet_model = models.Sequential([  
    base_model,  
    layers.GlobalAveragePooling2D(),  
    layers.Dense(2, activation='softmax') # Assuming 2 hair colors  
)  
  
mobilenet_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])  
history=mobilenet_model.fit(train_generator, validation_data=val_generator, epochs=10)  
  
Epoch 1/10  
13/13 [=====] - 42s 3s/step - loss: 0.6384 - accuracy: 0.6616 - val_loss: 0.3875 - val_accuracy: 0.852  
8
```

Evidence of Completion

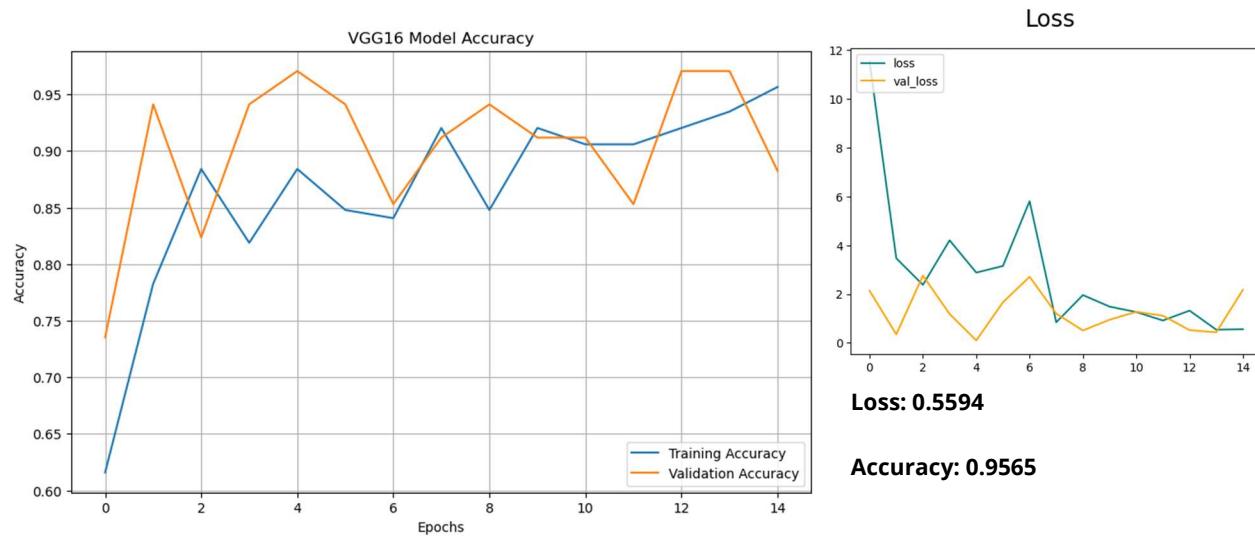


6.4. Model for the Dandruff & Lice Detection

6.4.1. CNN Architecture Model- VGG16

```
In [5]: train_generator = datagen.flow_from_directory(  
    'C:/Users/Ridma/Downloads/Dandruff',  
    target_size=img_size,  
    batch_size=batch_size,  
    class_mode='categorical',  
    subset='training'  
)  
Found 138 images belonging to 2 classes.  
  
In [6]: val_generator = datagen.flow_from_directory(  
    'C:/Users/Ridma/Downloads/Dandruff',  
    target_size=img_size,  
    batch_size=batch_size,  
    class_mode='categorical',  
    subset='validation'  
)  
Found 34 images belonging to 2 classes.  
  
In [7]: from tensorflow.keras.callbacks import EarlyStopping  
early_stop = EarlyStopping(monitor='val_loss', patience=10, verbose=1, restore_best_weights=True)  
  
In [8]: base_model = VGG16(input_shape=(160, 160, 3), include_top=False, weights='imagenet')  
base_model.trainable = False  
  
vgg16_model = models.Sequential([  
    base_model,  
    layers.Flatten(),  
    layers.Dense(4096, activation='relu'),  
    layers.Dropout(0.5),  
    layers.Dense(4096, activation='relu'),  
    layers.Dropout(0.5),  
    layers.Dense(2, activation='softmax') |  
])  
  
vgg16_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])  
history=vgg16_model.fit(train_generator, validation_data=val_generator, epochs=20 , callbacks=[early_stop])  
Epoch 1/20
```

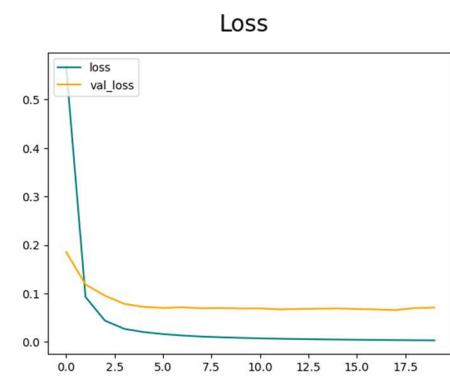
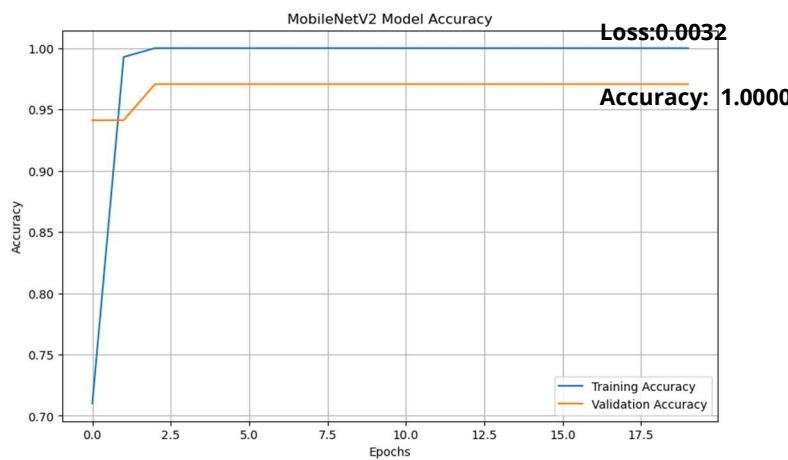
Evidence of Completion



6.4.2. CNN Architecture Model- MobileNetV2

```
In [25]: train_generator = datagen.flow_from_directory(  
    'C:/Users/Ridma/Downloads/Dandruff',  
    target_size=img_size,  
    batch_size=batch_size,  
    class_mode='categorical',  
    subset='training'  
)  
Found 138 images belonging to 2 classes.  
  
In [26]: val_generator = datagen.flow_from_directory(  
    'C:/Users/Ridma/Downloads/Dandruff',  
    target_size=img_size,  
    batch_size=batch_size,  
    class_mode='categorical',  
    subset='validation'  
)  
Found 34 images belonging to 2 classes.  
  
In [27]: from tensorflow.keras.callbacks import EarlyStopping  
early_stop = EarlyStopping(monitor='val_loss', patience=10, verbose=1, restore_best_weights=True)  
  
In [28]: base_model = MobileNetV2(input_shape=(160, 160, 3), include_top=False, weights='imagenet')  
base_model.trainable = False  
  
mobilenet_model = models.Sequential([  
    base_model,  
    layers.GlobalAveragePooling2D(),  
    layers.Dense(2, activation='softmax')  
)  
  
mobilenet_model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])  
history=mobilenet_model.fit(train_generator, validation_data=val_generator, epochs=20, callbacks=[early_stop])  
Epoch 1/20  
18/18 [=====] - 18s 657ms/step - loss: 0.5687 - accuracy: 0.7101 - val_loss: 0.1855 - val_accuracy: 0.  
9412  
Epoch 2/20
```

Evidence of Completion



7. Research Paper

The screenshot shows a Microsoft Word document titled "Research paper last update" with the status bar indicating "Saved". The document contains sections on "TRANSFER LEARNING", "IV. RESULTS AND DISCUSSION", and "A. Detection of hair color". It includes figures related to hair lice and dandruff detection and accuracy details for VGG16 and MobileNetV2.

TRANSFER LEARNING: Start with a pre-trained CNN model (e.g., VGG, [ResNet](#), or [MobileNet](#)) that has been trained on a large dataset like ImageNet. Fine-tune the model on your hair Quality detection task by retraining the final layers or some of the earlier layers.

IV. RESULTS AND DISCUSSION

A. Detection of hair color

According to the hair donation rules of Apeksa Hospital, currently they only receive black hair. Gray hair or dyed hair is not accepted. This system uses a model to identify

Figure 4 – Proposed Model-hair color

As per hair donation rules at Apeksa Hospital, they require clean, healthy hair free from dandruff and lice infestation. This system checks whether the donor's hair is healthy and free from dandruff and lice.

Figure 6– Proposed Model- hair lice and dandruff

Figure 7- hair lice and dandruff Accuracy details

VGG16 and MobileNetV2 are two powerful and successful architectures of deep learning CNN Architecture. To select the best architecture to identify the hair lice and dandruff, the model was trained by both architectures and the best architecture was selected. Accordingly, VGG16- Accuracy is 0.9565 and MobileNetV2 1.0000 was obtained. Accordingly,

8. Emails on Outlook

The screenshot shows the Microsoft Outlook inbox. An external email from "Lokesha Weerasinghe <lokeshaw@slit.lk>" is highlighted. The email subject is "IT20660352 Research Paper" and it has 2 MB attachments. The message body contains a note about external emails and a request for Dasunika to address comments and send back the research paper.

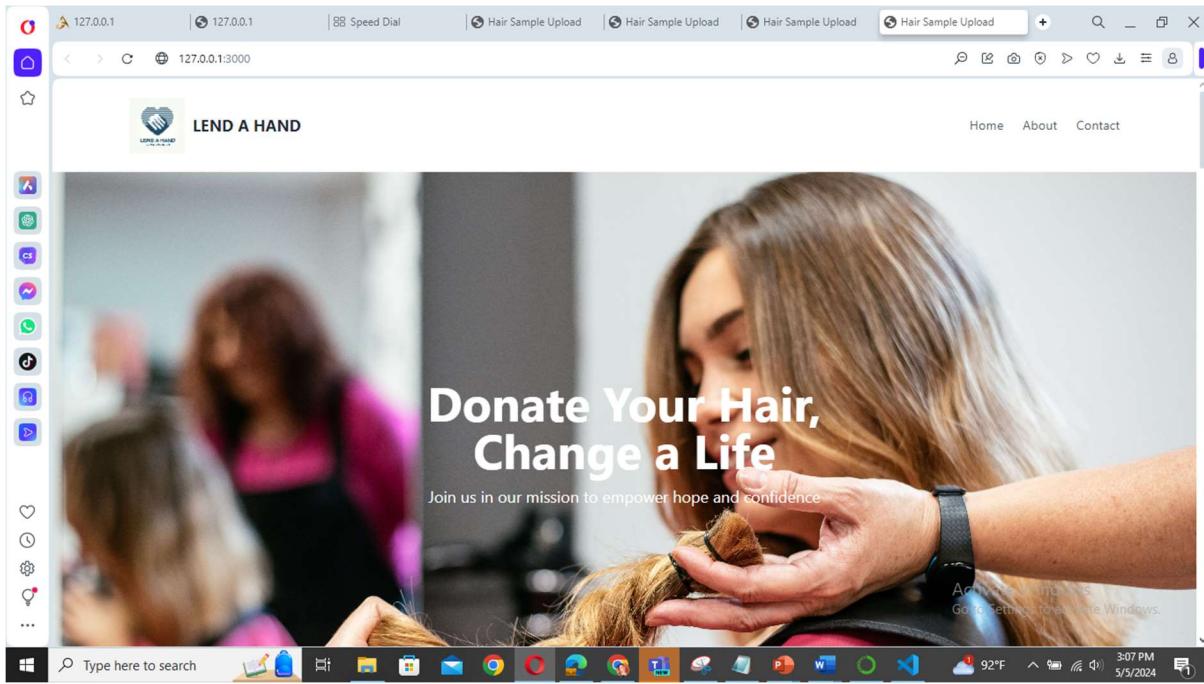
LW Lokesha Weerasinghe <lokeshaw@slit.lk>
To: Wijesooriya P.L.P.G.D.S it20660352
Wed 3/20/2024 9:35 PM

[EXTERNAL EMAIL] This email has been received from an external source - please review before actioning, clicking on links, or opening attachments.

Dear Dasunika,

Please address the comments added to the research paper and send back to me. You can refer to the research papers attached to get an idea about the content.

9. Implemented Interfaces of the Web application



A screenshot of a web browser window showing a guide for hair donation. The title is "Thinking of donating your hair to Apeksha Cancer Hospital - Sri Lanka?". It includes a photo of a woman holding a hair sample. Below the title, there is a "Step-by-Step Guide" section with three steps: "Step 1: Hair Preparation", "Step 2: Capture Detailed Images", and "Step 3: Upload and Receive Evaluation". Each step has a corresponding image and a detailed description. The "Step 3" section includes a "Choose File to Upload" button and a "Or drag and drop from here" placeholder. The browser's address bar shows "127.0.0.1:3000". The taskbar at the bottom includes icons for various Windows applications like File Explorer, Edge, and Mail.

CHECK THE QUALITY OF YOUR HAIR

LEND A HAND
DONATION SYSTEM

Hair Donation Guidelines

Thinking of donating your hair? Thank You!

We truly value your donation so, before your haircut, please read the guidelines below to ensure your hair is the correct length and in the best possible condition to be used in our wigs. And remember it costs £700 to provide one child or young person with a wig so why not order a free fundraising pack and help us turn your hair into a beautiful wig

Activate Windows
Go to Settings to activate Windows.

Upload Your Hair Sample Images - Step 1

Terms and Conditions

- 1. Be honest in uploading photos of the hair of those who wish to donate.
- 2. For this, enter only photos of hair. Do not enter other things or make fun of the system.
- 3. Include clear and unedited photos.

Upload Images
Activate Windows
Upload a file Go to Settings to activate Windows.
or drag and drop

LEND A HAND

Upload Your Hair Sample Images - Step I

Terms and Conditions

- 1. Be honest in uploading photos of the hair of those who wish to donate.
- 2. For this, enter only photos of hair. Do not enter other things or make fun of the system.
- 3. Include clear and unedited photos.

Image is a This is hair Activate Windows Go to Settings to activate Windows.

LEND A HAND

Upload Your Hair Sample Images - Step II

Terms and Conditions

- 1.Upload pictures of your hair that look good (full hair/hair in parts).
- 2.Upload pictures taken from different aspects of the hair.
- 3.The actual condition of your hair must be present in the picture you upload.

Step II

Activate Windows
Go to Settings to activate Windows.

LEND A HAND

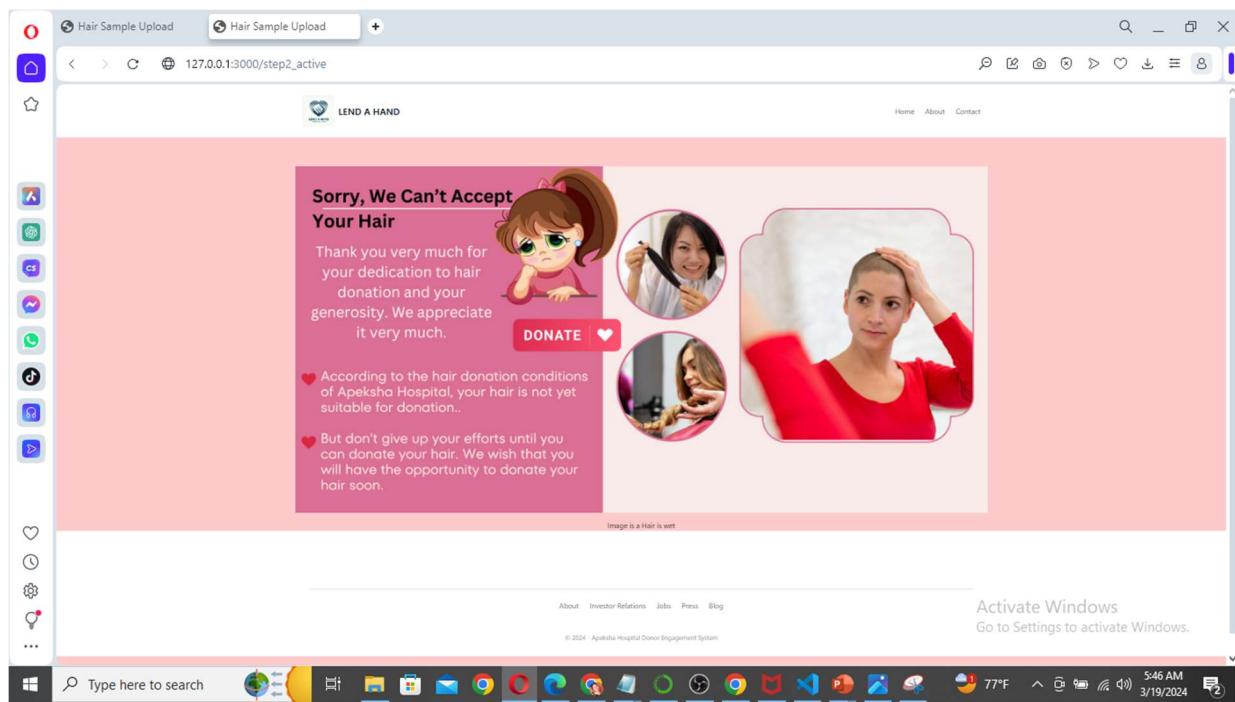
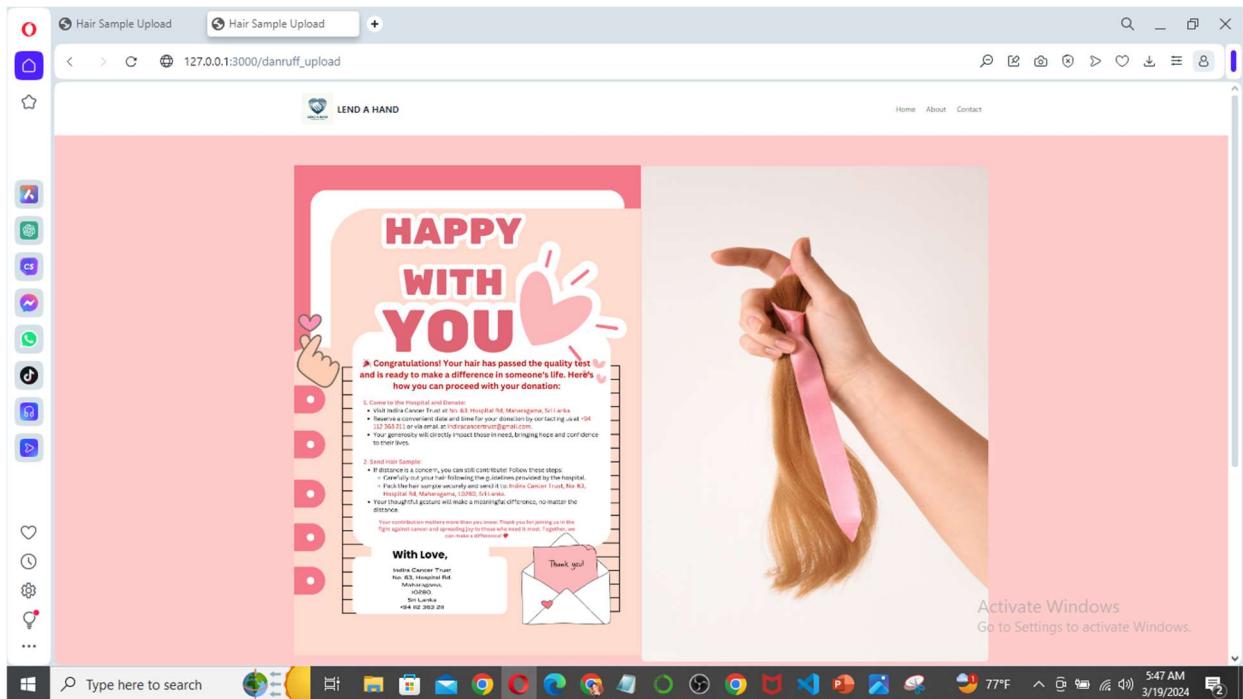
Upload Your Hair Sample Images - Step III

Terms and Conditions

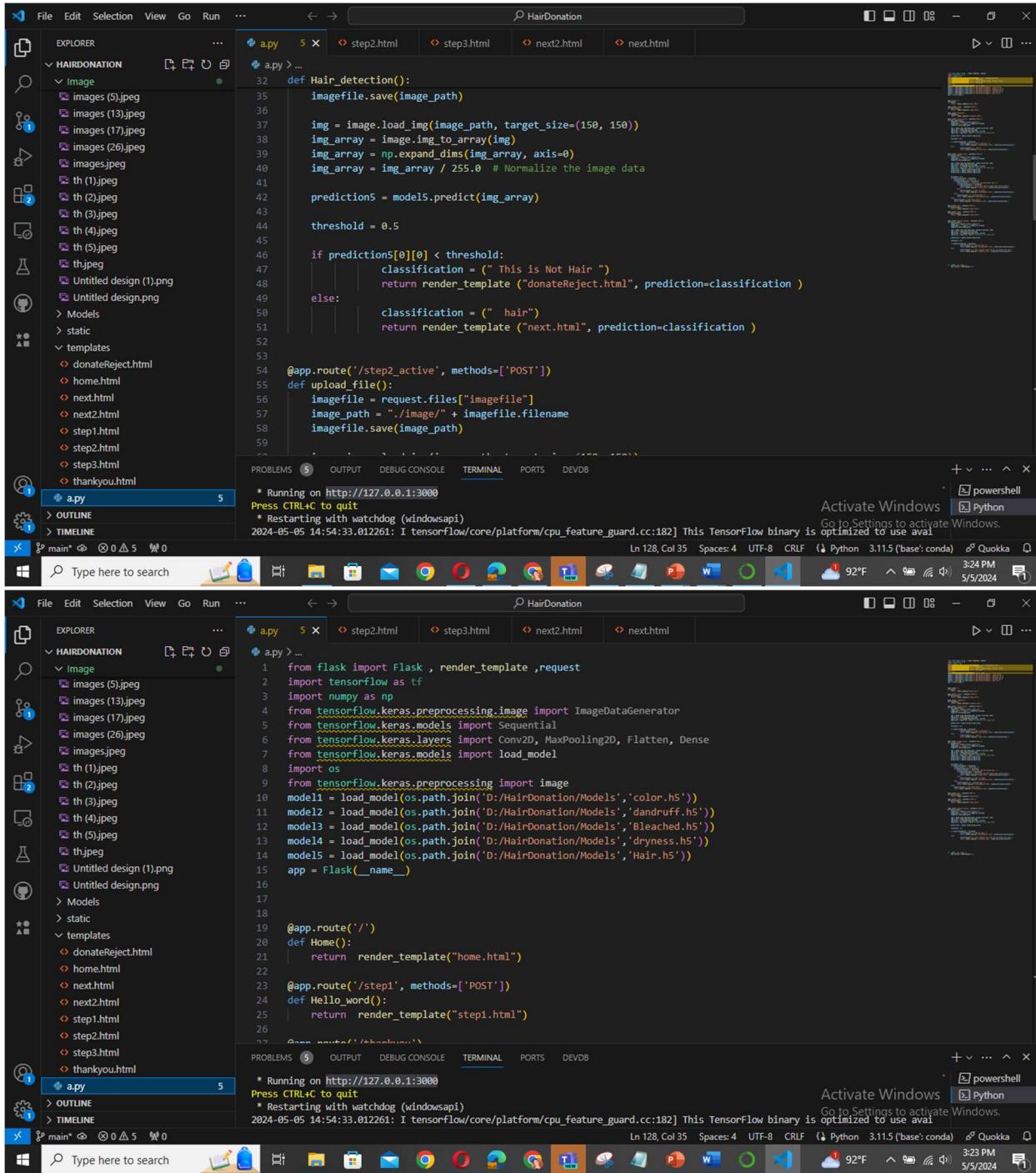
- 1.Take images very close to the scalp so that the scalp and hair roots are clearly visible.
- 2. Healthy hair free from dandruff and lice infestation.
- 3. Include clear and unedited photos.

Step III

Upload Images
Upload a file or drag and drop
Activate Windows
Go to Settings to activate Windows.



Backend Part



```
a.py 5 x step2.html step3.html next2.html nexhtml

def Hair_detection():
    imagefile.save(image_path)

    img = image.load_img(image_path, target_size=(150, 150))
    img_array = image.img_to_array(img)
    img_array = np.expand_dims(img_array, axis=0)
    img_array = img_array / 255.0 # Normalize the image data

    prediction5 = model5.predict(img_array)

    threshold = 0.5

    if prediction5[0][0] < threshold:
        classification = (" This is Not Hair ")
        return render_template("donateReject.html", prediction=classification )
    else:
        classification = (" hair")
        return render_template ("next.html", prediction=classification )

@app.route('/step2_active', methods=['POST'])
def upload_file():
    imagefile = request.files["imagefile"]
    image_path = "./image/" + imagefile.filename
    imagefile.save(image_path)

* Running on http://127.0.0.1:3000
Press CTRL+C to quit
* Restarting with watchdog (windowsapi)
2024-05-05 14:54:33.012261: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use avail

a.py 5 x step2.html step3.html next2.html nexhtml

from flask import Flask , render_template ,request
import tensorflow as tf
import numpy as np
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense
from tensorflow.keras.models import load_model
import os
from tensorflow.keras.preprocessing import image
model1 = load_model(os.path.join('D:/HairDonation/Models','color.h5'))
model2 = load_model(os.path.join('D:/HairDonation/Models','dandruff.h5'))
model3 = load_model(os.path.join('D:/HairDonation/Models','Bleached.h5'))
model4 = load_model(os.path.join('D:/HairDonation/Models','dryness.h5'))
model5 = load_model(os.path.join('D:/HairDonation/Models','Hair.h5'))
app = Flask(__name__)

@app.route('/')
def Home():
    return render_template("home.html")

@app.route('/step1', methods=['POST'])
def Hello_word():
    return render_template("step1.html")

* Running on http://127.0.0.1:3000
Press CTRL+C to quit
* Restarting with watchdog (windowsapi)
2024-05-05 14:54:33.012261: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use avail
```

10.Testing

Table 9.1 Test case to verify whether the captured image is stored cloud storage.

Test Case Id	01
Test Case	Verify image upload
Test Scenario	Verify whether the captured image is stored Google cloud storage
Input	Captured suspicious leaflet images
Expected Output	<ol style="list-style-type: none">1. 200 status code should be displayed2. The images must be stored in the firebase bucket
Actual Result	<ol style="list-style-type: none">1. 200 status code was displayed2. The images were stored in the firebase bucket
Status (Pass/Fail)	Pass

Table 9.2: Test case to classify and select the best model for hair identification

Test Case Id	02
Test Case	Classification of Hair identification using CNN
Test Scenario	Testing images to classify Hair and select the best model.
Precondition	512 labelled training & 134 validation images
Input	Test images (with and without hair)
Expected Output	High accuracy (mAP value)
Actual Result	High model accuracy with 95.60% mAP value
Status (Pass/Fail)	Pass

Table 9.3: Test case to classify and select the best model for hair color Detection

Test Case Id	03
Test Case	Detection of Hair Color using CNN
Test Scenario	Testing images to classify Hair Color and select the best model.
Precondition	660 labelled training & 205 validation images
Input	Test images (black color hair and other colors hair)
Expected Output	High accuracy (mAP value)
Actual Result	High model accuracy with 97.85% mAP value
Status (Pass/Fail)	Pass

Table 9.4: Test case to classify and select the best model for hair Bleached Detection

Test Case Id	04
Test Case	Detection of Hair Bleached using CNN
Test Scenario	Testing images to classify Hair Bleached and select the best model.
Precondition	1025 labelled training & 113 validation images
Input	Test images (Bleached hair and Not Bleached hair)
Expected Output	High accuracy (mAP value)
Actual Result	High model accuracy with 93.45% mAP value
Status (Pass/Fail)	Pass

Table 1Table 9.5: Test case to classify and select the best model for hair Dandruff & Lice Detection

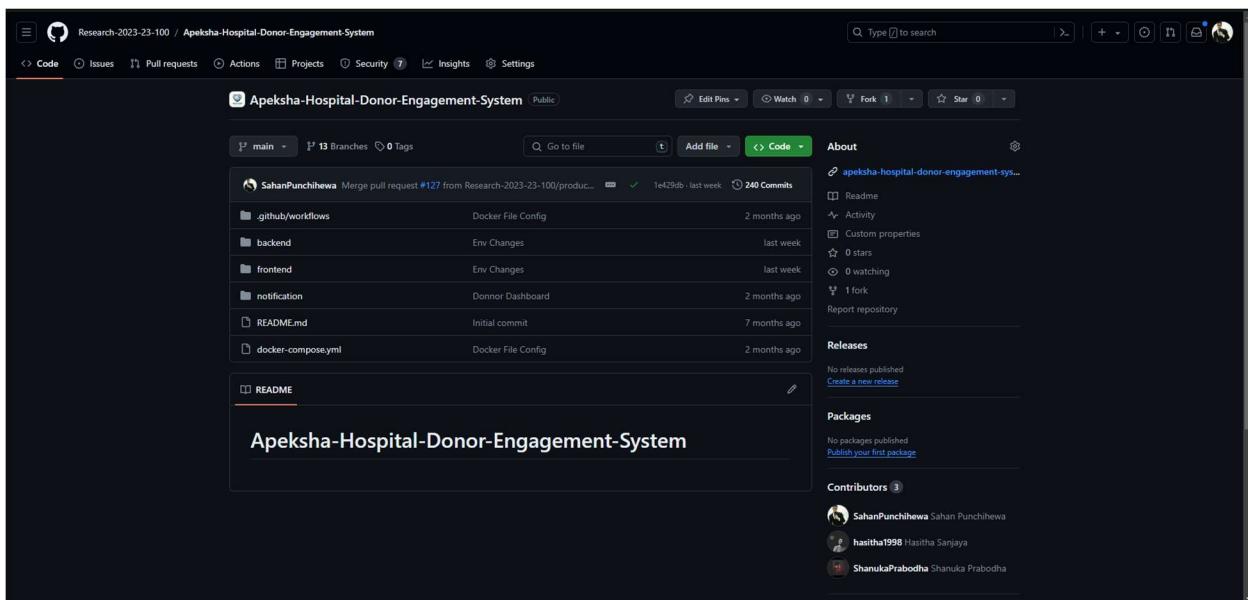
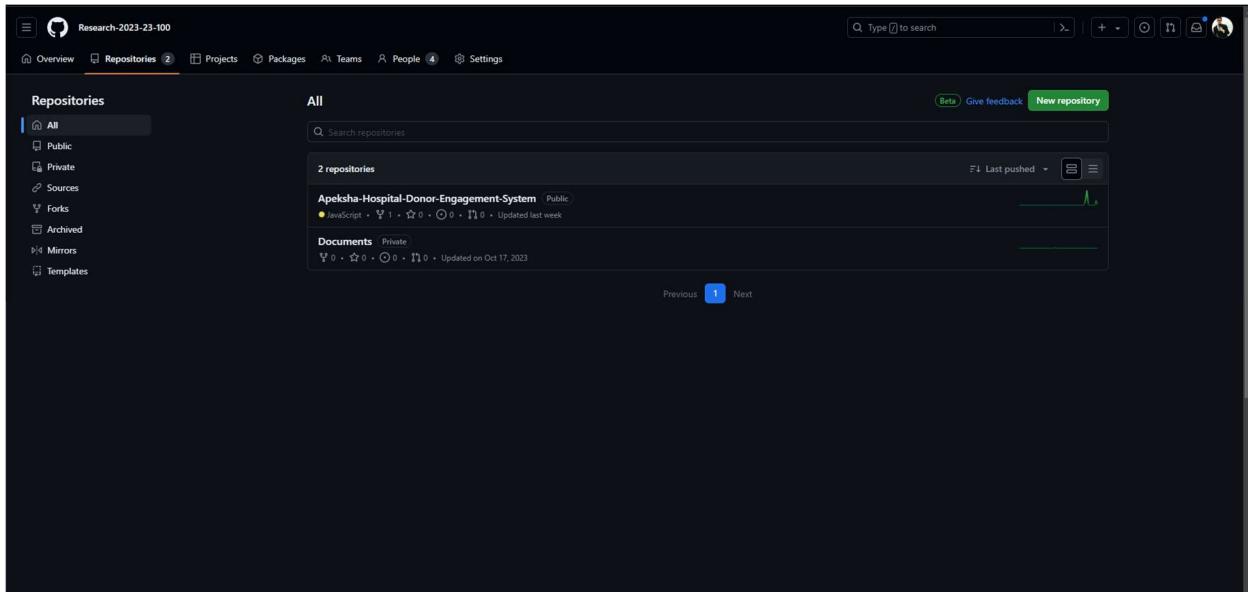
Test Case Id	05
Test Case	Detection of Hair Dandruff & Lice using CNN

Test Scenario	Testing images to classify Hair Dandruff & Lice and select the best model.
Precondition	700 labelled training & 88 validation images
Input	Test images (Dandruff & Lice hair and Not-Dandruff & Lice hair)
Expected Output	High accuracy (mAP value)
Actual Result	High model accuracy with 99.31% mAP value
Status (Pass/Fail)	Pass

Table 9.6: Test case to classify and select the best model for hair Dryness Detection

Test Case Id	06
Test Case	Detection of Hair Dryness using CNN
Test Scenario	Testing images to classify Hair Dryness and select the best model.
Precondition	1100 labelled training & 101 validation images
Input	Test images (Dryness hair and Not-Dryness hair)
Expected Output	High accuracy (mAP value)
Actual Result	High model accuracy with 88.56% mAP value
Status (Pass/Fail)	Pass

11. Version Control and Contribution



Research-2023-23-100 / Apeksha-Hospital-Donor-Engagement-System

Type to search

Code Issues Pull requests Actions Projects Security Insights Settings

Commits

main →

Commits on Apr 29, 2024

Merge pull request #127 from Research-2023-23-100/production →
SahanPunchihewa committed last week · 1 / 1
Verified 1e428db ⌂ ⌂

Merge pull request #126 from Research-2023-23-100/dev →
SahanPunchihewa committed last week · 2 / 2
Verified d39afec ⌂ ⌂

Commits on Apr 28, 2024

Merge pull request #125 from Research-2023-23-100/donor-login →
SahanPunchihewa committed last week · 3 / 3
Verified 9861bad ⌂ ⌂

Env Changes
SahanPunchihewa committed last week · 3 / 3
Verified 071db63 ⌂ ⌂

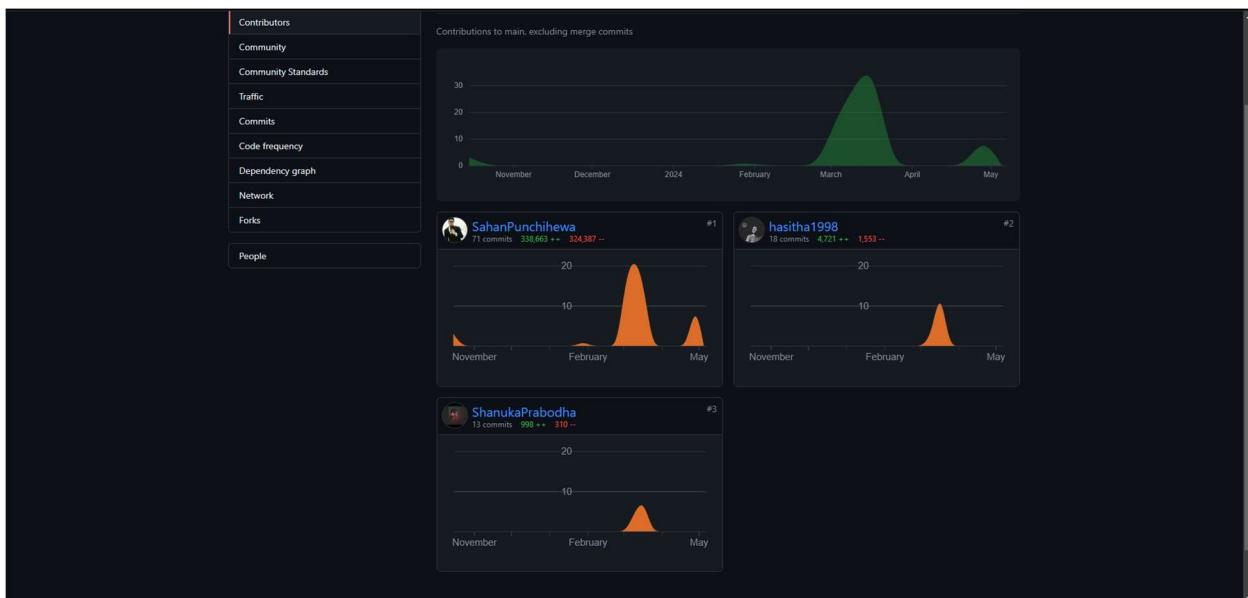
Merge pull request #124 from Research-2023-23-100/production →
SahanPunchihewa committed last week · 1 / 1
Verified 5874317 ⌂ ⌂

Merge pull request #123 from Research-2023-23-100/dev →
SahanPunchihewa committed last week · 2 / 2
Verified 6d41643 ⌂ ⌂

Merge pull request #122 from Research-2023-23-100/donor-login →
SahanPunchihewa committed last week · 3 / 3
Verified e77c663 ⌂ ⌂

Env Changes
SahanPunchihewa committed last week · 3 / 3
Verified 831a7a1 ⌂ ⌂

Merge pull request #121 from Research-2023-23-100/production →



The screenshot shows the GitHub Insights interface for the repository "Research-2023-23-100 / Apeksha-Hospital-Donor-Engagement-System". The "People" tab is selected. It displays a list of 4 people with access to the repository, categorized by their role: "Everyone" (Hasitha Sanjaya, Dassunika Sewwandi, Sahan Punchihewa, Shanuka Prabodha) and "Outside collaborators" (None). Each user entry includes a profile picture, name, GitHub handle, and their level of access (Admin or Write).

User	Access Level
Hasitha Sanjaya hasitha1998	Admin
Dassunika Sewwandi IT20660352-dasu	Write
Sahan Punchihewa SahanPunchihewa	Admin
Shanuka Prabodha ShanukaPrabodha	Admin

At the bottom of the page, there is a footer with links to GitHub's Terms, Privacy, Security, Status, Docs, Contact, Manage cookies, and a link to "Do not share my personal information".

12.Snap Shot

<div style="text-align: center;"> <p>SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY 16th Floor, BOC Merchant Tower, No. 28, St. Michael's Road, Colombo 03</p> <p>Date: 25/10/2023 Your Ref: My Ref: 2023-24-100</p> <p>Dear Sir/Madam,</p> <p>Certifying the project titled "Apelska Hospital Donor Engagement System" is conducted as in IT final year research project.</p> <p>The Sri Lanka Institute of Information Technology (SLIIT) is the largest Degree Awarding Institute in the field of Information Technology recognized by the University Grants Commission under the Universities Act. It was established in the year 1999 to educate and train Information Technology (IT) Professionals required by the fast-growing IT Industry in Sri Lanka.</p> <p>This letter is to certify the following students:</p> <p>IT006651169 - Panchalithra S.N. IT22660238 - Bandara H.R.G.D.S IT22660238 - Bandara H.R.G.D.S IT22660352 - Wijesooriya P.L.P.G.D.S</p> <p>They are final year undergraduate students who conduct research entitled "Apelska Hospital Donor Engagement System" as partial fulfillment of the B.Sc. in Information Technology degree at Sri Lanka Institute of Information Technology (SLIIT). The students are conducting the research under the supervision of Ms. K.M.Lokesha Prasadine.</p> <p>I kindly request your assistance in enabling these students to collect data from your organization to build their dataset for the research project. If you have any questions or require further clarification about the project, please do not hesitate to contact me.</p> <p>Thank you for your cooperation.</p> <p><i>[Signature]</i></p> <p>Dr. Jayantha Amararachchi Assistant Professor/ Research Project Coordinator, jayantha@slit.lk +94 11 754 4103</p> <p>Tel: +94(0)11 2301904 - 5 Fax: +94(0)11 2301906 E-mail: info@slit.lk URL: www.slit.lk</p> </div>	<div style="text-align: center;"> <p>SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY 16th Floor, BOC Merchant Tower, No. 28, St. Michael's Road, Colombo 03</p> <p>Date: 25/10/2023 Your Ref: My Ref: 2023-24-100</p> <p>Dear Sir/Madam,</p> <p>Certifying the project titled "Apelska Hospital Donor Engagement System" is conducted as in IT final year research project.</p> <p>The Sri Lanka Institute of Information Technology (SLIIT) is the largest Degree Awarding Institute in the field of Information Technology recognized by the University Grants Commission under the Universities Act. It was established in the year 1999 to educate and train Information Technology (IT) Professionals required by the fast-growing IT Industry in Sri Lanka.</p> <p>This letter is to certify the following students:</p> <p>IT006651169 - Panchalithra S.N. IT22660238 - Bandara H.R.G.D.S IT22660238 - Bandara H.R.G.D.S IT22660352 - Wijesooriya P.L.P.G.D.S</p> <p>They are final year undergraduate students who conduct research entitled "Apelska Hospital Donor Engagement System" as partial fulfillment of the B.Sc. in Information Technology degree at Sri Lanka Institute of Information Technology (SLIIT). The students are conducting the research under the supervision of Ms. K.M.Lokesha Prasadine.</p> <p>I kindly request your assistance in enabling these students to collect data from your organization to build their dataset for the research project. If you have any questions or require further clarification about the project, please do not hesitate to contact me.</p> <p>Thank you for your cooperation.</p> <p><i>[Signature]</i></p> <p>Dr. Jayantha Amararachchi Assistant Professor/ Research Project Coordinator, jayantha@slit.lk +94 11 754 4103</p> <p>Tel: +94(0)11 2301904 - 5 Fax: +94(0)11 2301906 E-mail: info@slit.lk URL: www.slit.lk</p> </div>	<div style="text-align: center;"> <p>SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY 16th Floor, BOC Merchant Tower, No. 28, St. Michael's Road, Colombo 03</p> <p>Date: 25/10/2023 Your Ref: My Ref: 2023-24-100</p> <p>Dear Sir/Madam,</p> <p>Certifying the project titled "Apelska Hospital Donor Engagement System" is conducted as in IT final year research project.</p> <p>The Sri Lanka Institute of Information Technology (SLIIT) is the largest Degree Awarding Institute in the field of Information Technology recognized by the University Grants Commission under the Universities Act. It was established in the year 1999 to educate and train Information Technology (IT) Professionals required by the fast-growing IT Industry in Sri Lanka.</p> <p>This letter is to certify the following students:</p> <p>IT006651169 - Panchalithra S.N. IT22660238 - Bandara H.R.G.D.S IT22660238 - Bandara H.R.G.D.S IT22660352 - Wijesooriya P.L.P.G.D.S</p> <p>They are final year undergraduate students who conduct research entitled "Apelska Hospital Donor Engagement System" as partial fulfillment of the B.Sc. in Information Technology degree at Sri Lanka Institute of Information Technology (SLIIT). The students are conducting the research under the supervision of Ms. K.M.Lokesha Prasadine.</p> <p>I kindly request your assistance in enabling these students to collect data from your organization to build their dataset for the research project. If you have any questions or require further clarification about the project, please do not hesitate to contact me.</p> <p>Thank you for your cooperation.</p> <p><i>[Signature]</i></p> <p>Dr. Jayantha Amararachchi Assistant Professor/ Research Project Coordinator, jayantha@slit.lk +94 11 754 4103</p> <p>Tel: +94(0)11 2301904 - 5 Fax: +94(0)11 2301906 E-mail: info@slit.lk URL: www.slit.lk</p> </div>
---	---	---

