

Apeksha Hospital Donor Engagement System

2023-24-100



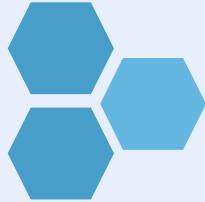


Team Members

Student Name	Student ID
Punchihewa S.N	IT20665166
Prabodha K.W.D.S	IT20665098
Bandara H.R.H.S	IT20662028
Wijesooriya P.L.P.G.D.S	IT20660352

Supervisor : Mrs.Lokesha Weerasinghe

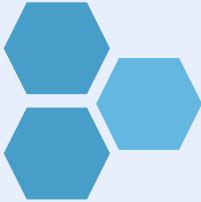
Co-Supervisor : Ms.Chamali Pabasara



Introduction

Our project aims to revolutionize the donation process at Apeksha Hospital, ensuring high-quality donations and patient care.

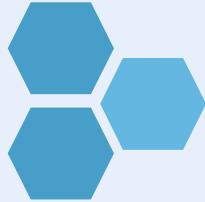




Research Problem

The lack of a reliable and comprehensive platform for donations at Apeksha Hospital poses several challenges, including fraudulent activities and the absence of a specific and trustworthy platform for hair donations. These issues have a significant impact on the overall donation process, leading to inefficiencies, a lack of transparency, and a potential compromise in the quality and suitability of donated items for patient care.





Objectives

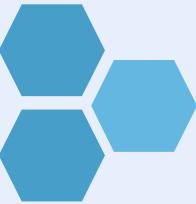


Main Objective

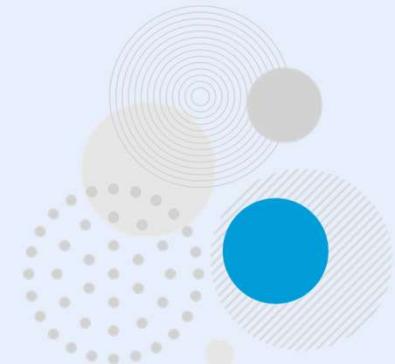
Improve communication, engagement, transparency, and security, while also optimizing donation campaigns and promoting high-quality donations.



Sub Objective



- ❖ Seamless Integration of Components
- ❖ User-Friendly Interface and Experience
- ❖ Scalability and Performance Optimization
- ❖ Continuous Monitoring and Improvement



Components

0
1

Intelligent Donor-Driven Inventory System for Essential Items

02

Critical Medication Priority Recommender System

03

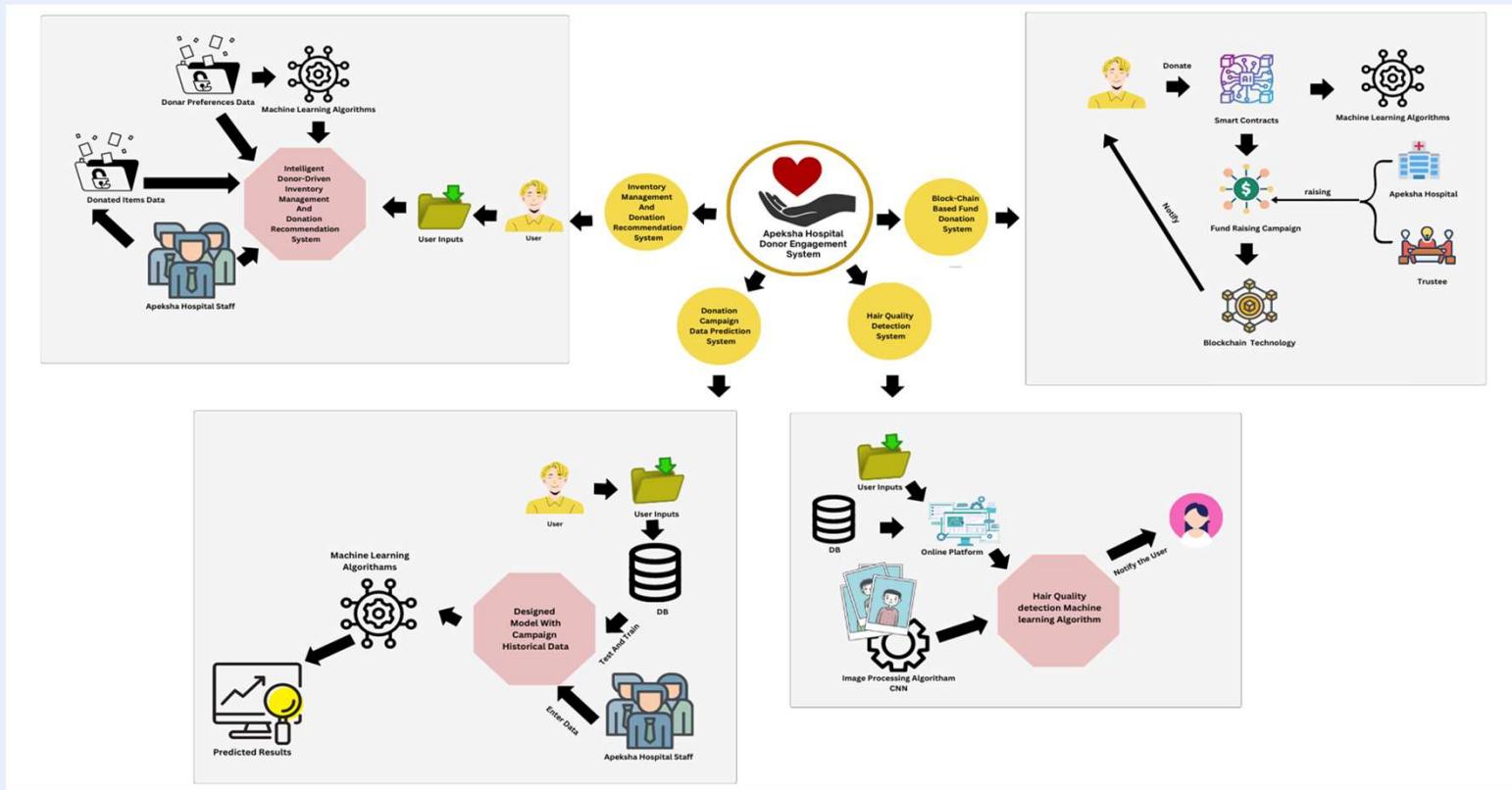
Predictive Analytics for Donation Campaign Success

04

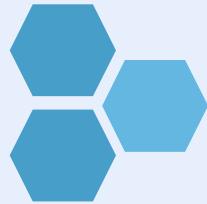
Promoting quality hair donation for cancer patients



System Overview Diagram



Stakeholders Evidence



SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY
1st Floor, Bot. Merchant Street, No. 28, St. Michael's Road, Colombo 03
Sri Lanka
Date: 25/10/2023 Your Ref: My Ref: 2023-24-100

Dear Madam,
I kindly request your organization to provide "Apoksha Hospital Donor Engagement System" as conducted as in
the following project.

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 1990 to educate and train Information Technology (IT) Professionals required by the fast-growing IT Industry in Sri Lanka.

They are final year undergraduate students who conduct research entitled "Apoksha Hospital Donor Engagement System" as partial fulfillment of the B.Sc. in Information Technology degree at Sri Lanka Institute of Information Technology (SLIIT), Colombo 03. The students are conducting the research under the supervision of Ms. K.M.Lokuhena Prasanna.

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.

Thank you for your cooperation.

Yours sincerely,
[Signature]
Dr. Jayantha Amarachchi
Assistant Professor/
Research Project Coordinator,
jayantha@sliit.lk
+94 11 754 4103

Received requested lots
to Nidhi Hospital unit
on 20th of Novem per month.
(2023, 2024)
M.C. (2013, 2018, 2019,
Officer in Charge 2020, 2021)
The National Cancer Institute
Maharagama

Tel: +94(011)23017904 - 5 Fax: +94(011)23017906 E-mail: info@sliit.lk URL: www.sliit.lk

[Signature]
Lanka Trustworthy Data Collected and Tested
[Signature]

SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY
1st Floor, Bot. Merchant Street, No. 28, St. Michael's Road, Colombo 03
Sri Lanka
Date: 25/10/2023 Your Ref: My Ref: 2023-24-100

Dear Sir/Madam,

My Ref: 2023-24-100
Your Ref: My Ref: 2023-24-100

Dear Sir/Madam,

Certifying the project titled "Apoksha Hospital Donor Engagement System" as conducted as in
the following research project.

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 1990 to educate and train Information Technology (IT) Professionals required by the fast-growing IT Industry in Sri Lanka.

This letter is to certify the following students.

IT2066516 - Purnithewa S.N
IT2066599 - Priyobodi K.W.O.S
IT2066202 - Bandara H.R.H.S
IT2066052 - Wigesooraya P.P.G.D.S

They are final year undergraduate students who conduct research entitled "Apoksha Hospital Donor Engagement System" as partial fulfillment of the B.Sc. in Information Technology degree at Sri Lanka Institute of Information Technology (SLIIT), Colombo 03. The students are conducting the research under the supervision of Ms. K.M.Lokuhena Prasanna.

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.

Thank you for your cooperation.

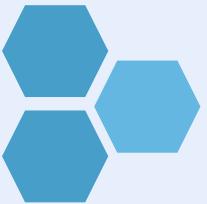
Yours sincerely,
[Signature]
Dr. Jayantha Amarachchi
Assistant Professor/
Research Project Coordinator,
jayantha@sliit.lk
+94 11 754 4103

Received requested lots
to Nidhi Hospital unit
on 20th of Novem per month.
(2023, 2024)
M.C. (2013, 2018, 2019,
Officer in Charge 2020, 2021)
The National Cancer Institute
Maharagama

Tel: +94(011)23017904 - 5 Fax: +94(011)23017906 E-mail: info@sliit.lk URL: www.sliit.lk

[Signature]
Lanka Trustworthy Data Collected and Tested
[Signature]

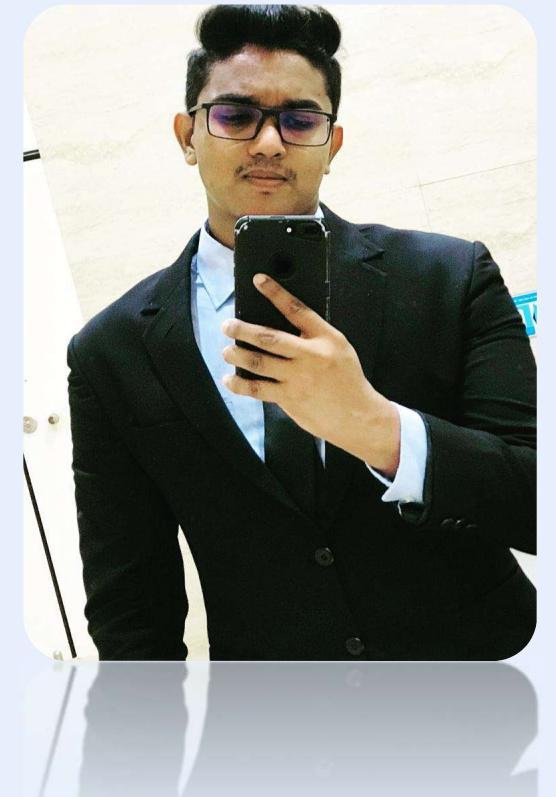




IT20665166

Punchihewa S.N

Bachelor of Science (Hons) in Information Technology
Specializing in Software Engineering

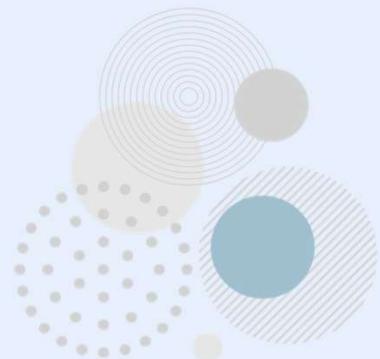




Introduction



Intelligent Donor-Driven Inventory System for Essential Items

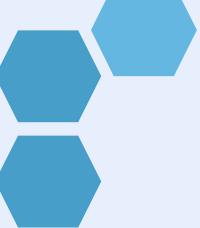


Background



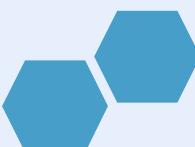
WHAT IS INTELLIGENT DONOR-DRIVEN INVENTORY SYSTEM?

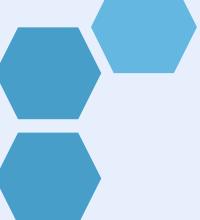
USING MACHINE LEARNING TO UNDERSTAND ESSENTIAL ITEMS IN INVENTORY SYSTEM?



Problem

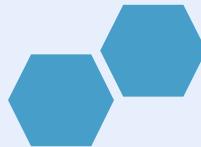
Intelligent donor-driven inventory management system. This solution utilizes machine learning algorithms to analyze Inventory behavior and historical Inventory usage patterns, recommending specific essential items required by the hospital. By providing an interactive interface for donors to select and track their impact, the system aims to enhance donor engagement and resource allocation efficiency.





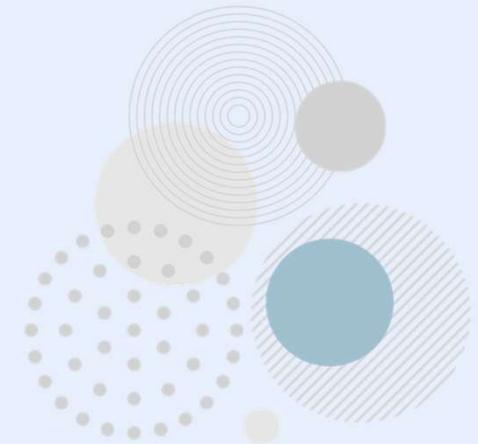
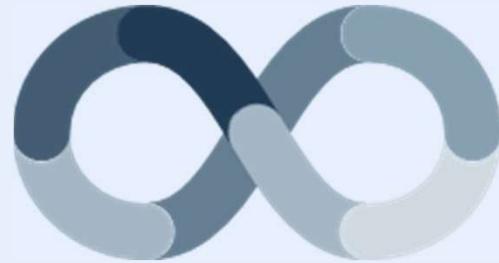
Objective



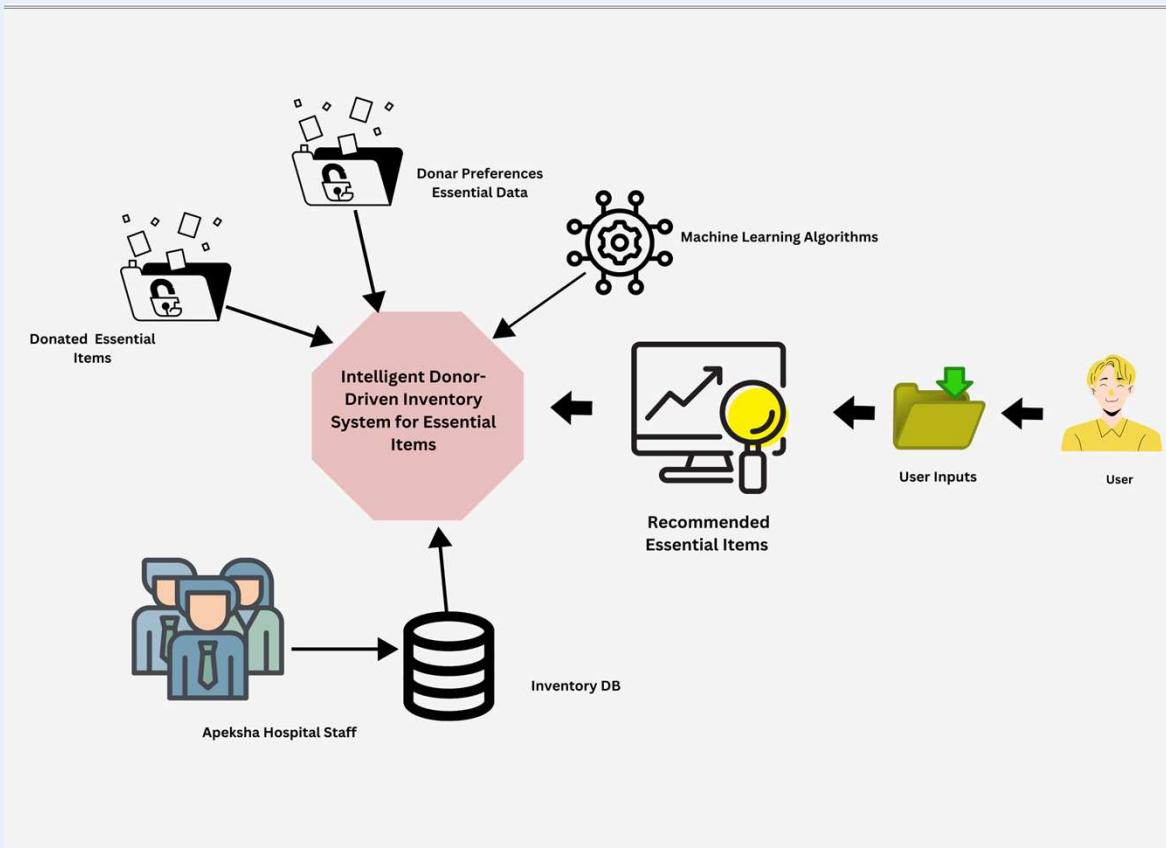
- ❖ Develop predictive models utilizing historical Inventory data and relevant parameters to forecast inventory demand.
 - ❖ Design a user-friendly dashboard enabling inventory managers to input variables and access real-time predictions.
 - ❖ Establish a monitoring system to continuously update predictions as inventory variables evolve.
- 



Research Methodology



Component Diagram



Technologies and Techniques

Programming Language

- ❖ Python

Tools

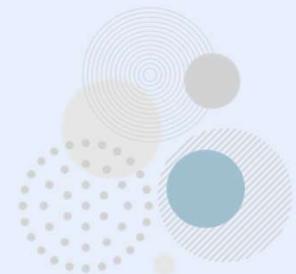
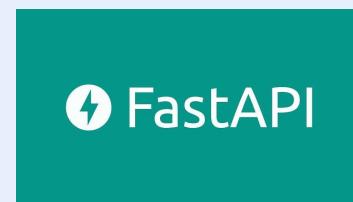
- ❖ Jupyter Notebook
- ❖ Scikit Learn
- ❖ Anaconda Navigator

Version Controlling

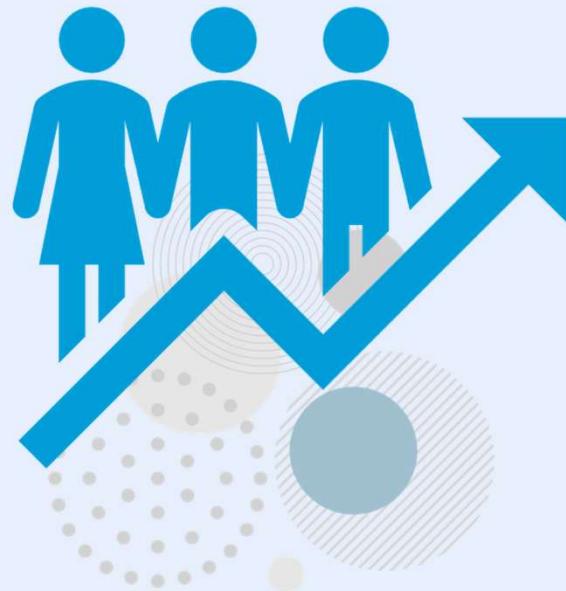
- ❖ GitHub

Algorithm

- ❖ Decision Tree Regression



PROGRESS



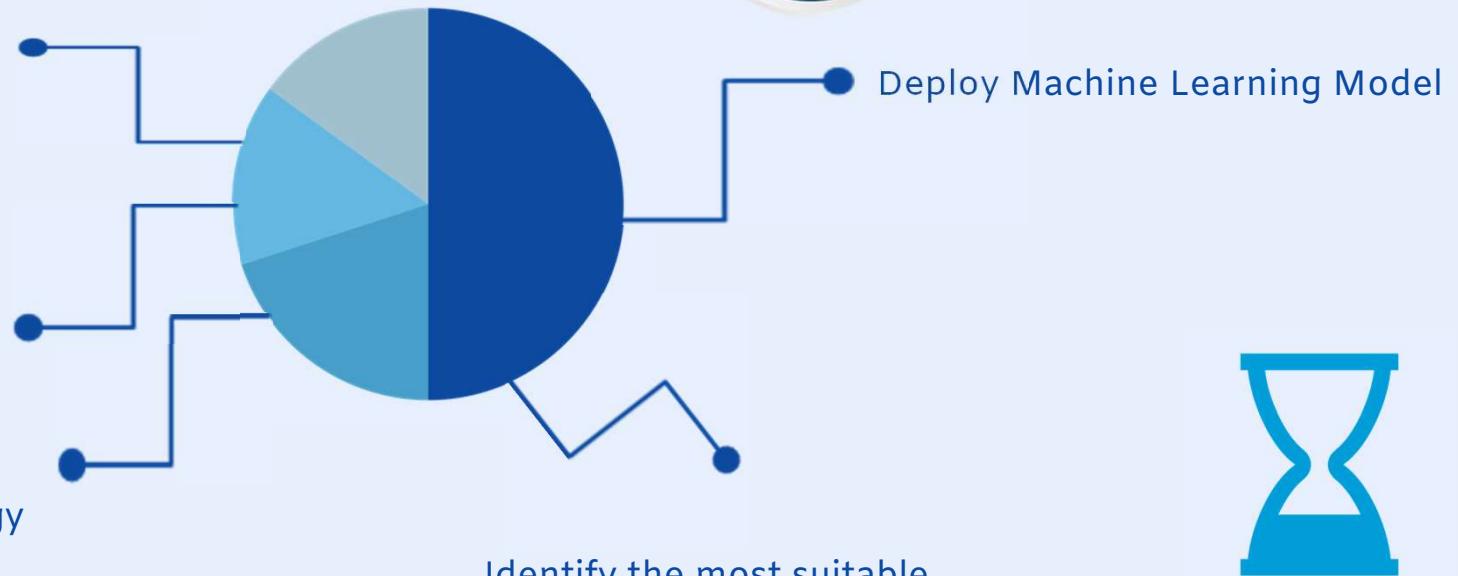
Current Progress

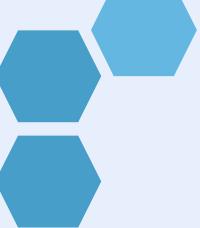
Implement Machine Learning Backend using FastAPI

Component integration 75% completed

Implement web application using MERN Stack Technology

Identify the most suitable algorithm for Machine learning model train





Next Expected Progress

- ❖ Enhance User Interface
- ❖ Complete component integration

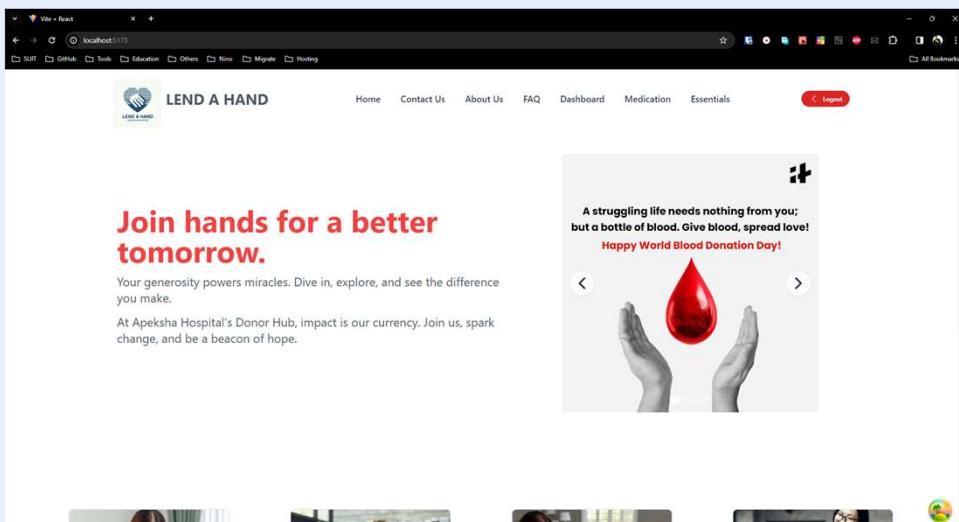




Project Evidence



Project Interfaces



A screenshot of a web browser showing the "Staff Dashboard" for "LEND A HAND". The dashboard has tabs for General, Essentials, Medication, and Blood Donation. The "Essentials" tab is active, displaying a table titled "Donation Request". The table lists items like BedCovers with a quantity of 5, donor information (sahanpunchihewa18@gmail.com), donation type (Essential), and status (Accepted, Rejected, Pending). A green "Essential" button and a red "Medication" button are visible above the table.

Machine Learning Implementation

Predicted Essential Items

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

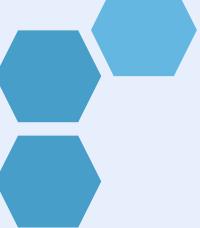
Staff Dashboard

GENERAL ESSENTIALS MEDICATION BLOOD DONATION

DONATION DONORS ESSENTIALS ITEMS PREDICTION ESSENTIAL ITEMS

Bar Chart for Predicted Values and Items

January



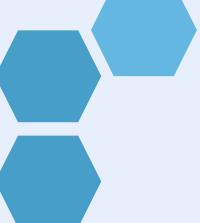
Risk Mitigation

- ❖ **Data privacy and Security Risks**

Implement robust encryption methods and access controls to protect sensitive donor information. Regularly audit and update security measures to stay ahead of potential threats.

- ❖ **Donor trust and engagement**

Maintain transparent communication with donors about how their data is used and the benefits of the system. Highlight success stories and demonstrate the impact of donations to enhance donor trust and engagement.



Functional Requirements

- ❖ Create an interactive and user-friendly inventory management interface
- ❖ Essentials Items prediction
- ❖ Donation Utilization
- ❖ Intergration with hospital workflow
- ❖ Optimization



Non-Functional Requirements



RELIABILITY



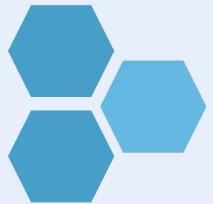
USER FRIENDLY



EFFICIENCY



ACCURACY



IT20665098

Prabodha K.W.D.S

Bachelor of Science (Hons) in Information Technology
Specializing in Software Engineering





Critical Medication Priority Recommender System

Introduction

- **Research Problem**

We're addressing medication supply challenges at Apeksha Hospital with a 'Critical Medication Priority Recommender System.' Using machine learning and optimization, we aim to identify crucial medications, predict shortages, optimize procurement, and elevate patient care quality.

- **Proposed Solution**

The system uses machine learning to predict shortages, identify critical medications, and optimize recommendations at Apeksha Hospital, improving patient care. It consists of a Shortage Prediction Model and a Critical Medication Identification Model.





Critical Medication Priority Recommender System

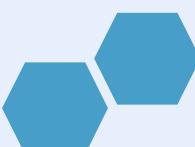
Introduction

- **Main Objective**

"Developing an intelligent system for Apeksha Hospital using machine learning to predict shortages, identify critical medications, and optimize their procurement, thus enhancing patient care."

- **Specific Objectives**

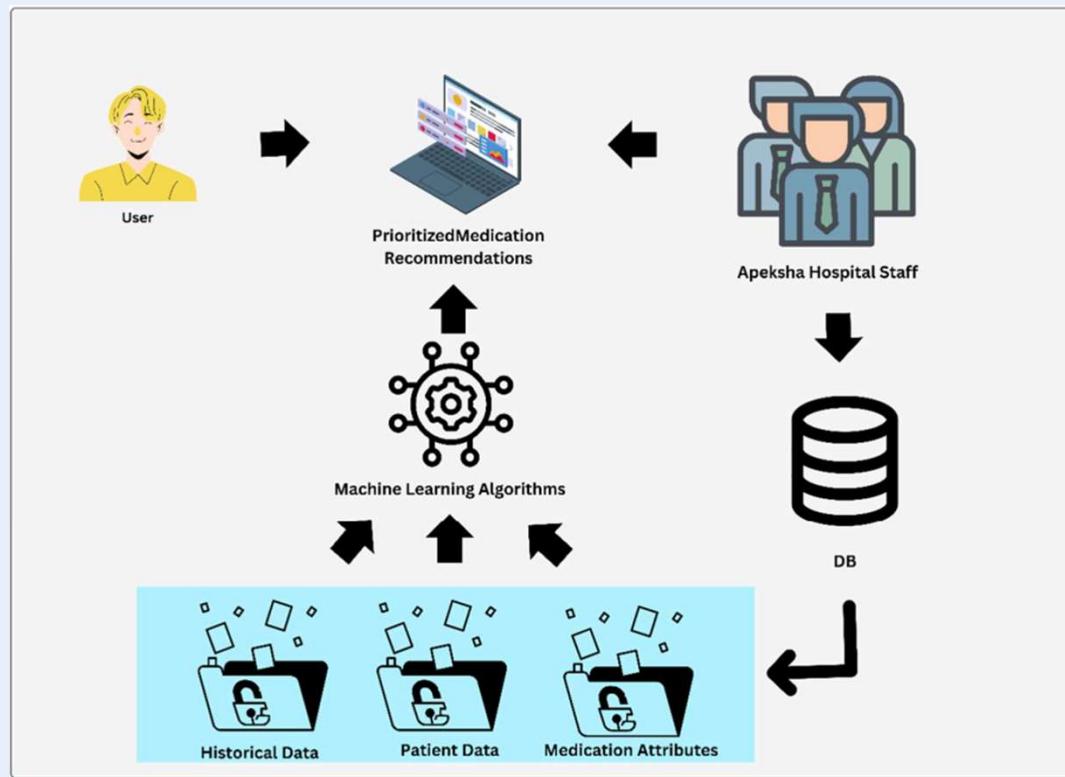
1. Critical Medication Identification Model
2. Medication Shortage Prediction Model
3. Model Integration
4. User-Friendly Interface



Critical Medication Priority Recommender System

Methodology

- Component Diagram



Critical Medication Priority Recommender System

Methodology

- **Tools And Technologies**

- Programming Language**

- ❖ Python

- Tools**

- ❖ Jupyter Notebook
 - ❖ Anaconda Navigator
 - ❖ Scikit Learn
 - ❖ Fast API
 - ❖ Render

- Version Controlling**

- ❖ GitHub

- Algorithms**

- ❖ Random Forest Classification



ANACONDA®





Critical Medication Priority Recommender System

Methodology

Sub objective 01 - Critical Medication Identification Model

Data Collection

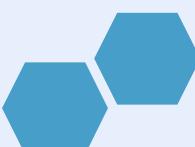
- Collected data from Apeksha Hospital regarding shortaged medications.

Data Pre-Processing

- Conversion of Categorical Variables to Numerical Format
- Date Column Conversion
- Date Feature Extraction

Implement Prediction Model

- Implement Machine Learning Model using "**Random Forest Classifier**" Algorithm





Critical Medication Priority Recommender System

Methodology

Sub objective 01 - Critical Medication Identification Model – Sample Codes

Random Forest Classifier Accuracy- 86%

```
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
```





Critical Medication Priority Recommender System

Methodology

Sub objective 01 - Critical Medication Identification Model

Random Forest Classifier

```
Accuracy: 0.8604651162790697
```

```
Train set shape: (171, 3)
```

```
Test set shape: (43, 3)
```





Critical Medication Priority Recommender System

Methodology

Sub objective 02 - Medication Shortage Prediction Model

Data Collection

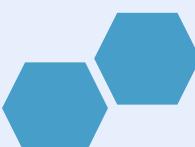
- Collected data from Apeksha Hospital.

Data Pre-Processing

- Conversion of Categorical Variables to Numerical format
- Date Column Conversion
- Date Feature Extraction
- Applied feature scaling using the StandardScaler from scikit-learn

Implement Prediction Model

- Implement Machine Learning Model using "**Logistic Regression**" Algorithm
- Implement Machine Learning Model using "**Random Forest**" Algorithm





Critical Medication Priority Recommender System

Methodology

Sub objective 02 - Medication Shortage Prediction Model – Sample Code

Logistic Regression Algorithm Accuracy- 78%

```
1 # Import necessary libraries
2 import pandas as pd
3 from sklearn.model_selection import train_test_split
4 from sklearn.preprocessing import StandardScaler, OneHotEncoder
5 from sklearn.compose import ColumnTransformer
6 from sklearn.pipeline import Pipeline
7 from sklearn.linear_model import LogisticRegression
8 from sklearn.impute import SimpleImputer
9 from sklearn.metrics import accuracy_score
10 import joblib
11
12 # Load the dataset containing all necessary information
13 df = pd.read_csv("medication_shortage.csv")
14
15 # Convert date columns to datetime
16 df['shelf_life'] = pd.to_datetime(df['shelf_life'])
17 df['Admin_Timestamp'] = pd.to_datetime(df['Admin_Timestamp'])
18
19 # Extract date features
20 df['Shelf_Life_Day'] = df['shelf_life'].dt.day
21 df['Shelf_Life_Month'] = df['shelf_life'].dt.month
22 df['Shelf_Life_Year'] = df['shelf_life'].dt.year
23 df['Admin_Day'] = df['Admin_Timestamp'].dt.day
24 df['Admin_Month'] = df['Admin_Timestamp'].dt.month
25 df['Admin_Year'] = df['Admin_Timestamp'].dt.year
26
27 # Drop unnecessary columns
28 df.drop(['shelf_life', 'Admin_Timestamp'], axis=1, inplace=True)
29
30 # Split features and target variable
31 X = df.drop(columns=['shortaged', 'Name', 'ID']) # Exclude non-numeric columns
32 y = df['shortaged']
33
34 # Split data into train and test sets
35 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=21)
36
37 # Define column transformer to handle numeric and categorical features
38 numeric_features = X.select_dtypes(include=['int64', 'float64']).columns
39 numeric_transformer = Pipeline(steps=[('imputer', SimpleImputer(strategy='median')),
40 ('scaler', StandardScaler())],
```

Random Forest Classifier

Accuracy- 91%

```
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=[('imputer', SimpleImputer(strategy='median')), ('scaler', StandardScaler())])
38
39 categorical_features = X.select_dtypes(include=['object']).columns
```



Critical Medication Priority Recommender System

Methodology

Sub objective 02 - Medication Shortage Prediction

Model

Random Forest Classifier

```
Predicted_Shortage_RF
 0           1
 1           1
 2           1
 3           1
 4           1
 ...
 ...
1207         0
1208         0
1209         1
1210         1
1211         0

[1212 rows x 3 columns]
Random Forest Model Accuracy: 0.9125412541254125
```



Implemented Medication Dashboard

LEND A HAND

Home Contact Us About Us FAQ Dashboard Logout

Predicted Medication Shortages

Filter by: 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/5mlVial	not shortaged
5	Abiraterone Acetate Tab.250mg	not shortaged

Medication Counts

Legend: Medication Shortage

Previous Next

LEND A HAND

Home Contact Us About Us FAQ Dashboard Logout

Medication Priority Level

Search Medications Filter by Priority

Medications

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

Priority Distribution

Legend: Priority Counts



Critical Medication Priority Recommender System

Current Progress

Implemented Shortage
Prediction model

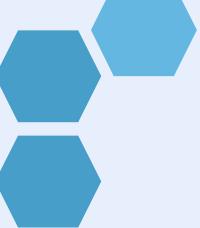
Implemented Critical
medication Identification
model

Model Integration and User
Interface Implementation



Deploy Machine Learning Models

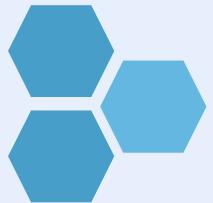




Next Expected Progress

- ❖ Enhance User Interfaces and Complete UI implementation.
- ❖ Complete component integration

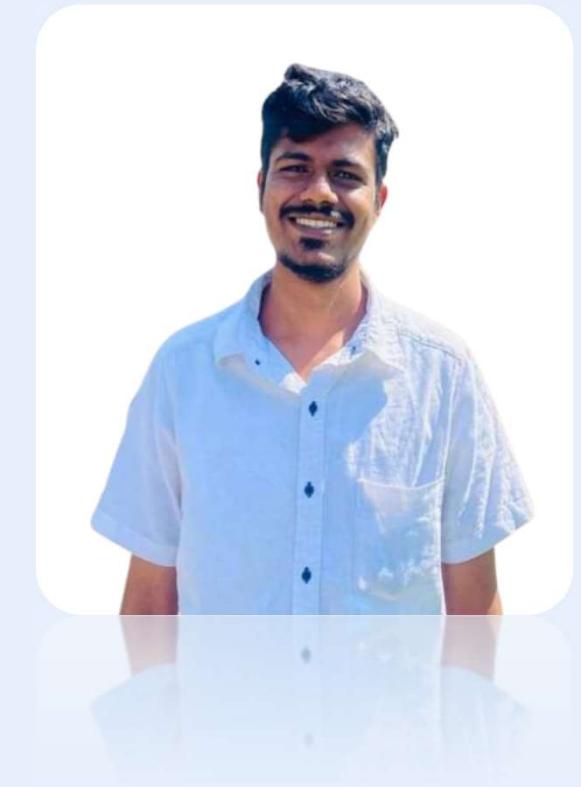




IT20662028

Bandara H.R.H.S

Bachelor of Science (Hons) in Information Technology
Specializing in Software Engineering

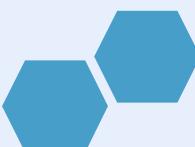




Introduction

Blood Donation Campaign Organizing Challenges

- There is High Demand for blood in Apeksha hospital.
- Organize donation Campaigns without Prediction.
- Food Wastage and Crowd Management Issues.
- Waste campaign marketing cost and another cost.





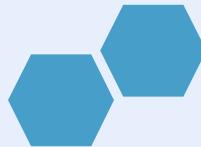
Research Problem & Solution

- Problem:

Apeksha Hospital's blood donation campaigns face challenges due to low donor participation and resource inefficiencies.

- Solution:

Using predictive analytics and machine learning, Apeksha Hospital can boost campaign effectiveness by predicting success factors and optimizing resource allocation.





Specific and Sub Objective

- **Main Objective**

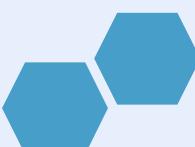
Implement machine learning to enhance the efficiency of blood donation campaigns.

- **Specific Objectives**

- 1.Optimization Algorithms Integration

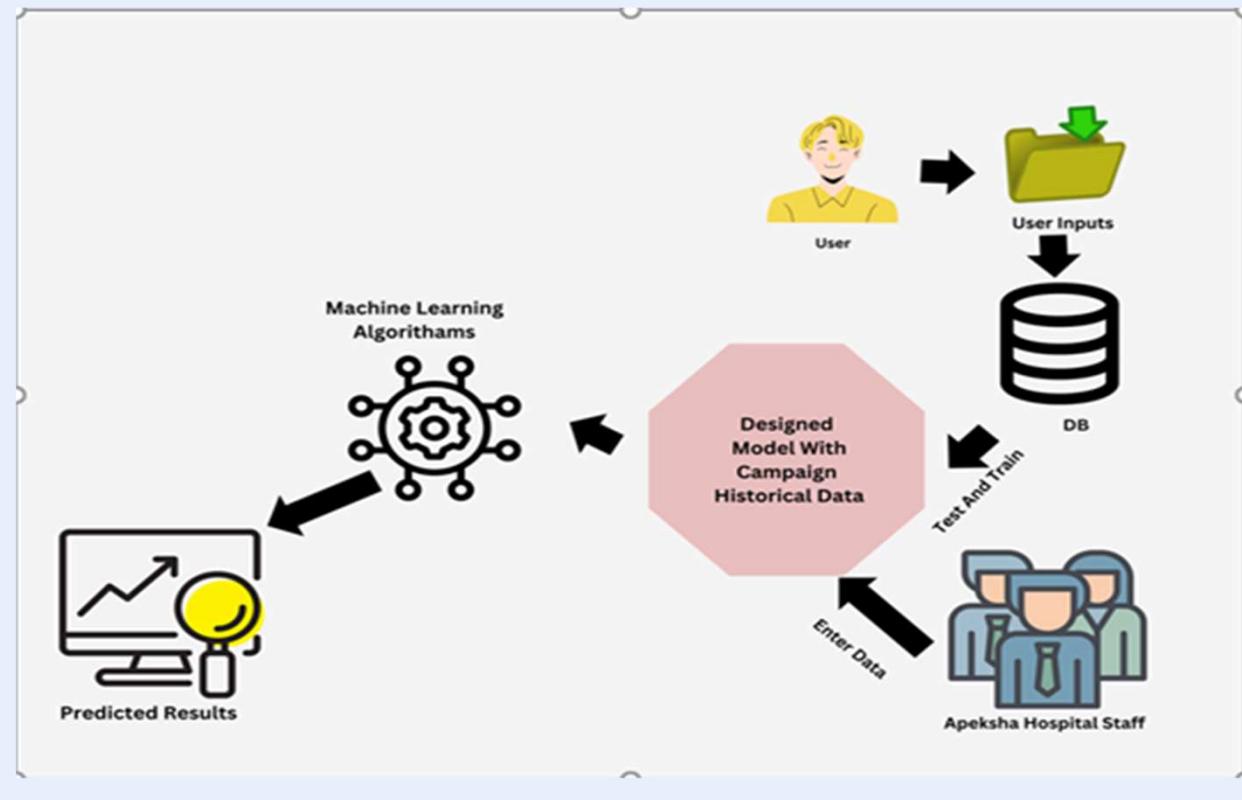
- 2.Predict People amount

- 3.User-Friendly Interface



Methodology

- Component Diagram



Methodology

- Tools And Technologies

Programming Language

- ❖ Python
- ❖ Java Script

Tools

- ❖ Jupyter Notebook
- ❖ Anaconda Navigator
- ❖ Scikit Learn
- ❖ Joblib

Version Controlling

- ❖ GitHub

Algorithm

- ❖ RandomForestRegressor





Methodology

Evidence of Completion

Data collection

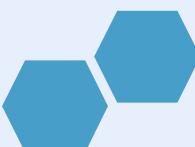
Data pre-processing

Used Random Forest Regressor Algorithm for training models

Data Visualization

Host the Current model in the flask server and Render

Displayed the prediction in web application



Data Collection and Pre-processing

This screenshot shows a Jupyter Notebook interface with two code cells and their outputs.

In [7]:

```
import numpy as np
import pandas as pd
data = pd.read_csv('mydata2.csv')
data
```

Out[7]:

	Date	Day_Type	Number_of_Attendees
0	1/1/2021	public	200
1	1/3/2021	weekend	129
2	1/10/2021	weekend	172
3	1/12/2021	weekday	120
4	1/27/2021	weekday	149
...
173	11/22/2023	weekday	136
174	11/25/2023	weekend	197
175	12/1/2023	weekday	141
176	12/6/2023	weekday	63
177	12/29/2023	weekday	147

178 rows × 3 columns

In [8]:

```
df = pd.DataFrame(data)
```

This screenshot shows a Jupyter Notebook interface with two code cells and their outputs.

In [8]:

```
df = pd.DataFrame(data)

# 1. Convert "Date" to datetime
df['Date'] = pd.to_datetime(df['Date'])

# 2. Encode "Day_Type" using one-hot encoding
df_encoded = pd.get_dummies(df, columns=['Day_Type'], prefix='Day_Type')

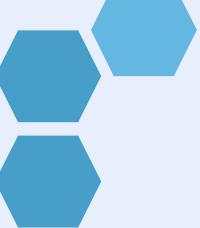
# Extract date components from "Date"
df_encoded['Month'] = df_encoded['Date'].dt.month

# Drop the original "Date" column
df_encoded.drop('Date', axis=1, inplace=True)
```

In [9]:

```
print(df_encoded)
```

	Number_of_Attendees	Day_Type_poyday	Day_Type_public	Day_Type_weekday	Day_Type_weekend	Month
0	200	0	1	0	0	0
1	128	0	0	0	0	0
2	172	0	0	0	0	0
3	120	0	0	1	0	0
4	149	0	0	1	0	0
...
173	136	0	0	0	1	0
174	197	0	0	0	0	0
175	141	0	0	1	0	0
176	63	0	0	1	0	0
177	147	0	0	1	0	0



Random Forest Regressor Accuracy

```
Enter the month (1-12): 2
```

```
Is it a Poyaday? (0 or 1): 1
```

```
Is it a public holiday? (0 or 1): 0
```

```
Is it a weekday? (0 or 1): 0
```

```
Is it a weekend? (0 or 1): 1
```

```
Predicted Number of Attendees: 342.16
```

```
R-squared (R2) on the entire dataset: 0.9292643387482032
```



Display the results in Web UI

Screenshot of a web browser showing a donation system interface titled "LEND A HAND".

The browser tabs include:

- Donor section by hasitha1998
- Presentation - OneDrive
- 2023-24-100 -Progress Presenta
- Vite + React

The main content area shows a "Attendance Prediction" form with dropdown menus for "Month" (February) and "Day Type" (Public), and buttons for "Predict", "Organize Camp", and "Refresh". The predicted number of attendees is displayed as 202.

To the right of the form is a cartoon illustration of a red blood drop character and a bottle character dressed as a doctor, holding hands and surrounded by medical icons like a heart with a plus sign, a stethoscope, and a first aid kit.

The browser status bar at the bottom shows various icons and the date/time: 4:15 AM, 3/19/2024.



Critical Medication Priority Recommender System

Current Progress

Implemented people amount Prediction model

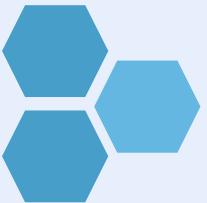
Implemented blood camp organizers and donation camp

Model Integration and User Interface Implementation



Deploy Machine Learning Models





IT20660352

Wijesooriya P.L.P.G.D.S

Bachelor of Science (Hons) in Information Technology
Specializing in Software Engineering



Promoting Quality Hair Donation for Cancer Patients

Apeksha hospital-based hair donation process currently has no specific method. Applying to build the system to obtain hair from hair donors according to the standards recommended by the hospital and to eliminate the distance between the hospital and the donor. Using CNN Architecture in image processing.



How to determine whether hair donated by hair donors meets the standards and qualities of hair recommended by the Apeksha hospital?



I donated my hair to cancer patients.
But,

Apeksha Hospital could not donate hair due to the informal procedure there.



Introduction Background



Specific and Sub Objective

Specific
Objective

Implementation of Deep Learning Algorithm (CNN) with online platform to examine key quality factors of hair. – main objective

- Model Implemented to Hair Identification- 1.1
- Model Implemented to identify Color of Hair -1.2
- Model Implemented to identify bleached of Hair – 1.3
- Model Implemented to identify Dryness of Hair – 1.4
- Model Implemented to identify lice & dandruff of Hair -1.5

Sub
Objectives

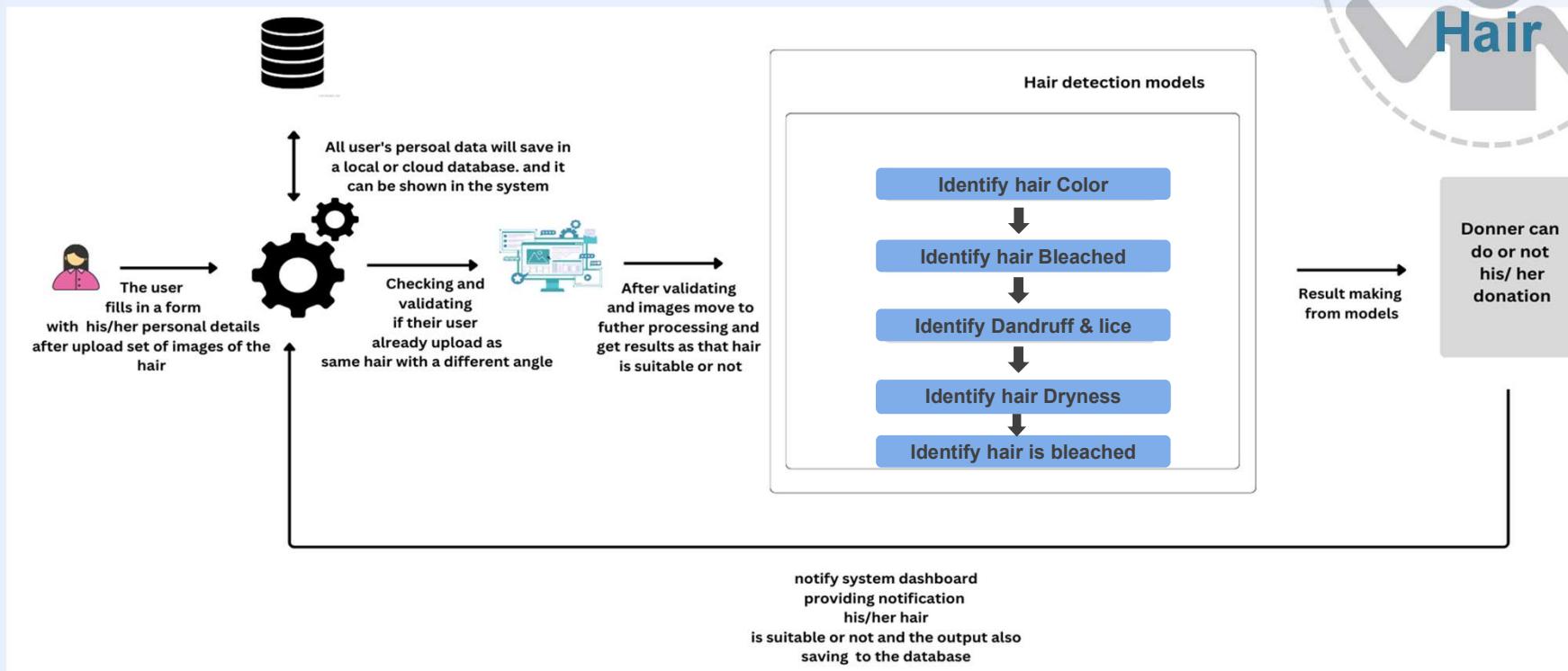
01 - Collected data which needs to train the prediction models

02 - Combine all the models to implement the final model

03 - Implement the Web Application with User-friendly Interface.



Component Diagram



Methodology



Data collection



Data Pre Processing and Create Data Set With Data Augmentation



Used transfer learning based Convolutional Neural Network(CNN) architectures for training models



Data augmentation and selected the best technique to retrain models to achieve good fitting



Tuning hyper-parameters and retrained models to select the best architecture



Data Pre Processing and Create Data Set With Data Augmentation



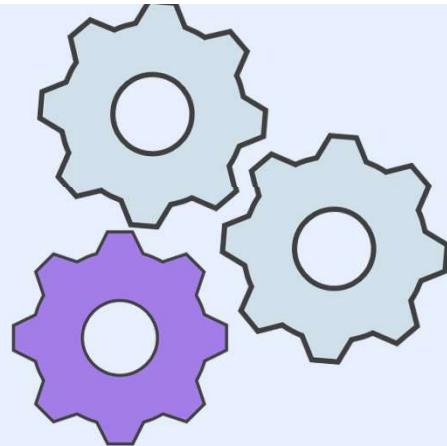
Data Visualization



Host the finalized model in the flask server



Displayed the results in web application



Used Techniques and Technologies



Techniques

- Transfer learning
- Data Augmentation
- Normalization



Technologies

- React
- Python
- TensorFlow
- Anaconda Environment
- Node Server
- Jupyter Notebook
- Google Colab
- Visual Studio Code



Preprocessing for the image data sets

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

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

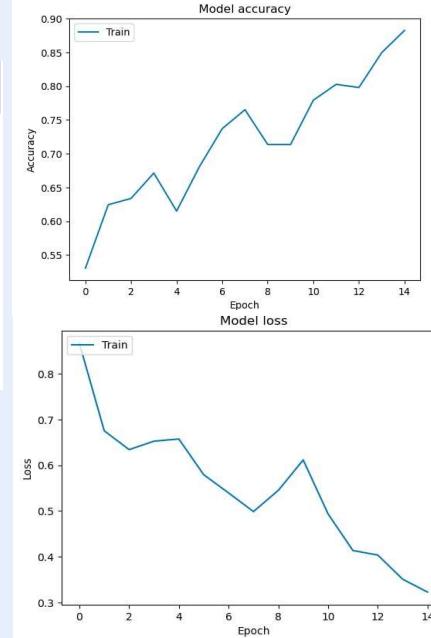
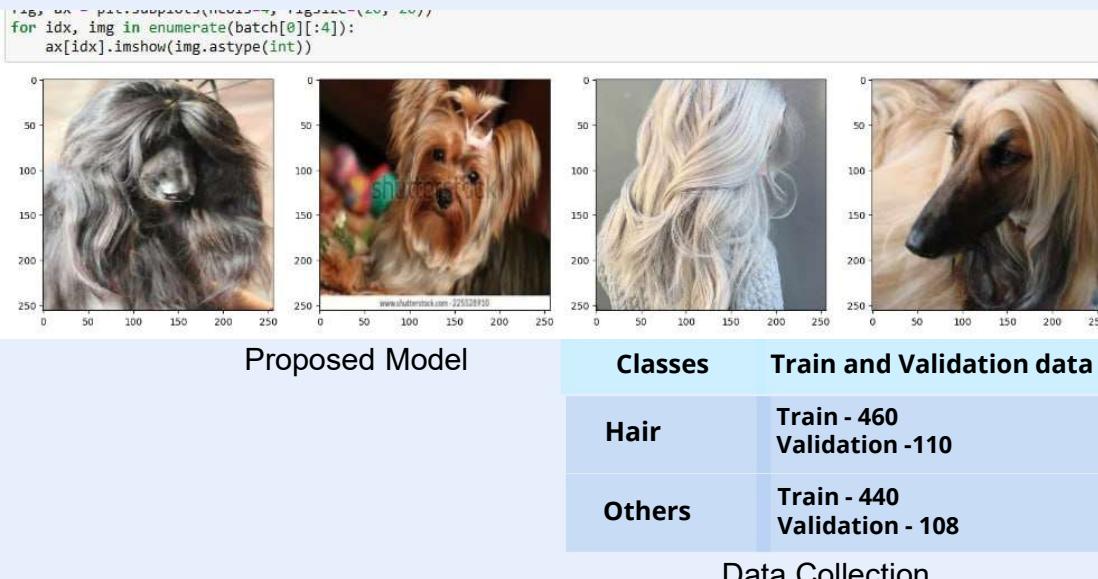
In [7]: data_dir='C:/Users/Ridma/Downloads/Hairs'
for image_class in os.listdir(data_dir):
    for image in os.listdir(os.path.join(data_dir, image_class)):
        image_path = os.path.join(data_dir, image_class, image)
        try:
            img = cv2.imread(image_path)
            tip = imghdr.what(image_path)
            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
```

01. Model for the Hair identification

The donor will confirm whether the images uploaded to the system are related to a hair or not. It was found that **MobileNetV2** is the best architecture to implement in image processing.

- To implement this model the data was collected through [istockphoto](#) data collection.



Accuracy: 0.9560
Loss: 0.0453

Best Architecture

MobileNetV2

02. Model for the Hair Color Detection

According to the hair donation rules of Apeksha Hospital, currently they only receive black hair. Other colors hair is not accepted. This system uses a model to identify the color of the donor's hair. Only black color hair can be donated here.

- To implement this model the data was collected through istockphoto data set.

```
In [5]: data = tf.keras.utils.image_dataset_from_directory('C:/Users/Ridma/Downloads/newimages')
Found 989 files belonging to 2 classes.

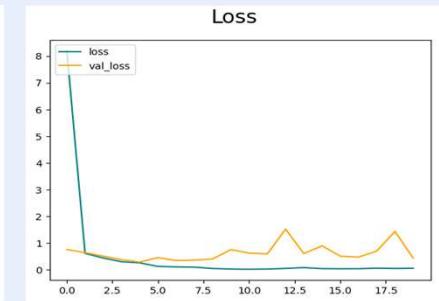
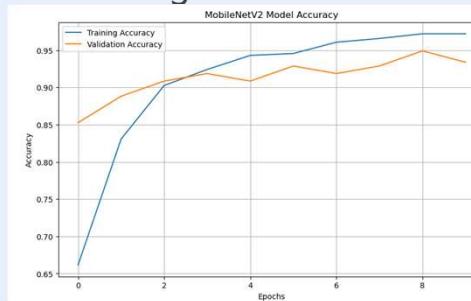
In [6]: data_iterator = data.as_numpy_iterator()

In [7]: batch = data_iterator.next()

In [8]: 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



Loss: 0.0576
Accuracy: 0.9785

Best Architecture

MobileNetV2

Classes	Train and Validation data
Black Hair	Train - 480 Validation - 100
Other colours Hair	Train - 450 Validation - 105



Data Collection

03. Model for the hair Bleached Detection

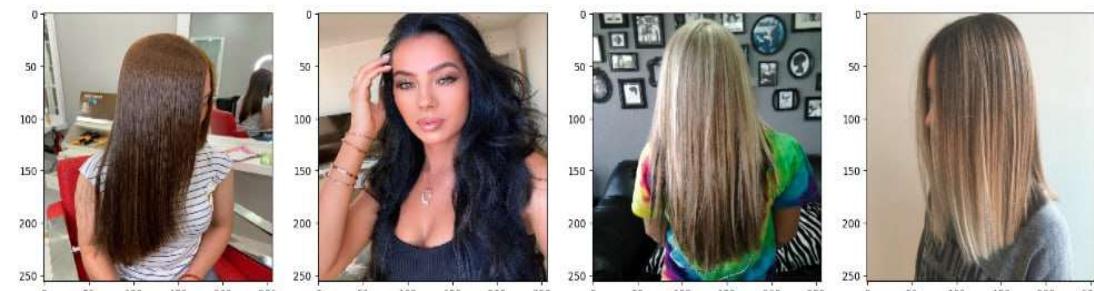
It verifies whether the hair images uploaded by the donor are bleached or not.

- To implement this model the data was collected through a [Kaggle](#) data set.



Best
Architecture

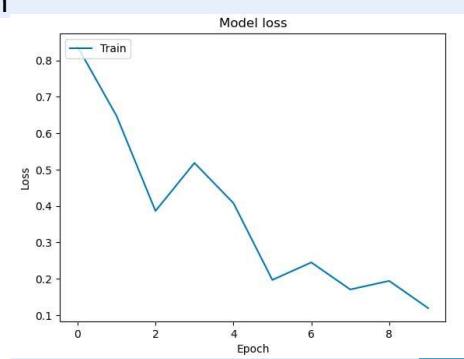
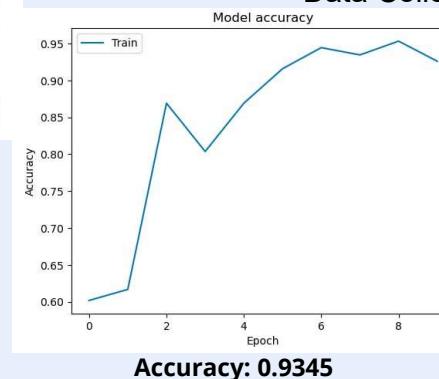
```
fig, ax = plt.subplots(ncols=4, figsize=(20, 20))
for idx, img in enumerate(batch[0][4]):
    ax[idx].imshow(img.astype(int))
```



Proposed Model

Classes	Train, Validation and Test data
bleached Hair	Train - 150 Validation - 37
Not bleached Hair	Train - 430 Validation - 76

Data Collection

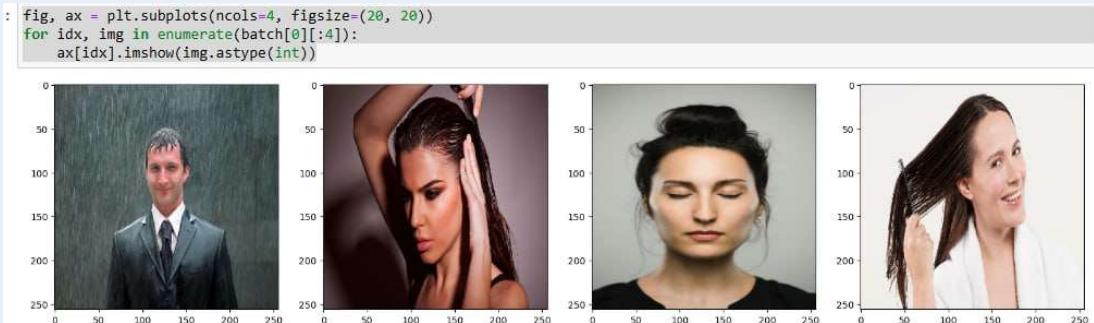


04. Model for the hair Dryness Detection

The hair images uploaded by the donor will confirm whether the hair is dry or not.



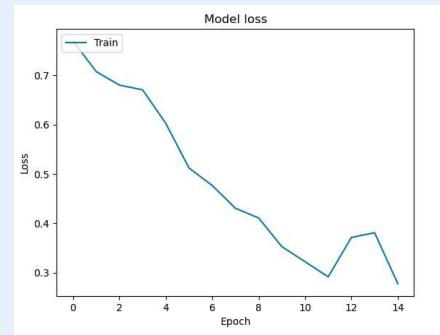
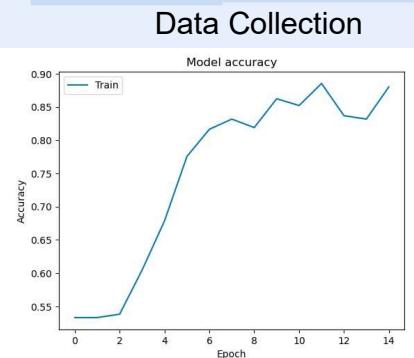
- To implement this model the data was collected through a [Kaggle](#) data set.



Proposed Model

Classes	Train, Validation and Test data
Dry Hair	Train - 468 Validation - 67
Wet Hair	Train - 130 Validation - 36

VGG16



05. Model for the Dandruff & Lice Detection

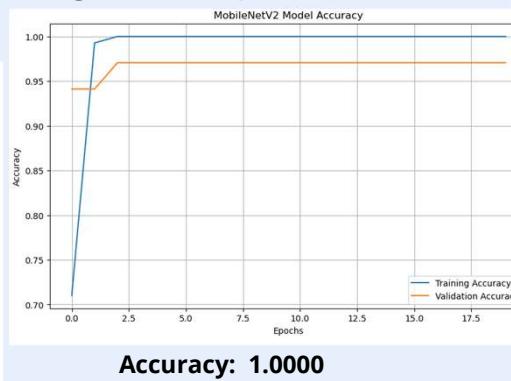
As per hair donation rules at Apeksha Hospital, they require clean, healthy hair free from dandruff and lice infestation. This system checks whether the donor's hair is healthy and free from dandruff and lice.

- To implement this model the data was collected through [istockphoto](#) data set.

```
n [12]: data_iterator = data.as_numpy_iterator()  
n [13]: batch = data_iterator.next()  
n [14]: fig, ax = plt.subplots(ncols=4, figsize=(20,20))  
for idx, img in enumerate(batch[0][1:4]):  
    ax[idx].imshow(img.astype(int))  
    ax[idx].title.set_text(batch[1][idx])
```

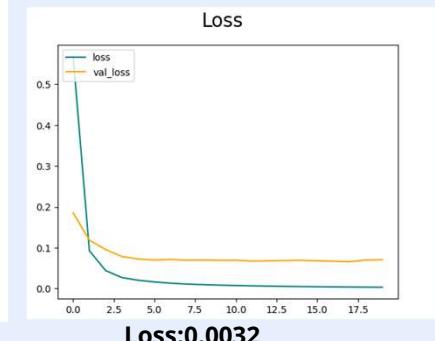


Proposed Model



Classes	Train, Validation and Test data
Lice & Dandruff	Train - 160 Validation - 48
Hair	Train - 150 Validation - 40

Data Collection



Best
Architecture

MobileNetV2

Display the results in Web UI

The image displays a grid of nine screenshots of a web-based user interface for a hair donation application. The application has a pink and white color scheme with heart-shaped icons.

- Step 1: Take better hair photos**
A banner with a woman holding a phone and a pink ribbon. Below it is a section titled "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.
 - An "Upload Images" button with the text "Upload a file or drag and drop".
- Step II: Upload Your Hair Sample Images**
A banner with a woman smiling. Below it is a section titled "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.
 - An "Upload Images" button with the text "Upload a file or drag and drop".
- Step III: Upload Your Hair Sample Images**
A banner with a woman holding her hair. Below it is a section titled "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.
 - An "Upload Images" button with the text "Upload a file or drag and drop".
- Sorry, We Can't Accept Your Hair**
A banner with a cartoon character. Below it is a section titled "Sorry, We Can't Accept Your Hair".
 - Text: "Thank you very much for your dedication to hair donation and your generosity. We appreciate it very much."
 - Text: "According to the hair donation conditions of Apexka Cancer Hospital, your hair is not yet suitable for donation."
 - Text: "But don't give up your efforts until you can donate your hair. We wish that you will have the opportunity to donate your hair soon!"
 - A "DONATE" button with a heart icon.
- Thinking of donating your hair to Apexka Cancer Hospital - Sri Lanka?**
A banner with a woman holding her hair. Below it is a section titled "Thinking of donating your hair to Apexka Cancer Hospital - Sri Lanka?".
 - Text: "Before you consider donating your hair, we want to assure you that we are committed to making the process as smooth and painless as possible. We understand that the decision to donate hair can be a difficult one, and we are here to support you every step of the way."
 - Text: "We offer a variety of services to help you throughout the process, including professional styling, hair loss support, and emotional guidance. Our team is dedicated to ensuring that you feel comfortable and supported throughout the entire process."
 - Text: "We believe that everyone deserves the right to feel confident and beautiful, regardless of their hair length or style. That's why we offer a range of services to help you achieve your goals, whether you're looking for a new hairstyle or simply want to feel more confident in your own skin."
 - Text: "We are committed to making a difference in the lives of women and girls around the world. By donating your hair, you are helping to provide hope and support to women and girls who are facing challenges like cancer, hair loss, and other health issues."
 - Text: "We are also committed to making a difference in the lives of women and girls around the world. By donating your hair, you are helping to provide hope and support to women and girls who are facing challenges like cancer, hair loss, and other health issues."
- Step-by-Step Guide**
A banner with a woman holding her hair. Below it is a section titled "Step-by-Step Guide".
 - Text: "Before donating your hair, follow the steps below to check if the hair meets the standards recommended by the hospital."
 - Three steps:
 - Step 1: Hair Preparation**
 - Text: "First, prepare the hair sample you plan to donate. Make sure the hair is clean and free from any debris or contaminants. You may need to wash the hair before taking a photo. Make sure the lighting and angles are correct for a accurate representation of the hair's texture and color."
 - Step 2: Capture Detailed Images**
 - Text: "Capture high quality photos of your hair sample. Make sure the lighting and angles are correct for a accurate representation of the hair's texture and color."
 - Step 3: Upload and Receive Evaluation**
 - Text: "Upload the captured images to the website. The website will evaluate the suitability of your hair sample through a series of automated checks and manual review. If your hair is suitable, you will receive a confirmation email and a link to the next step in the process."

Functional Requirements



- VERIFY THAT THE PHOTOGRAPHS PROVIDED BY THE DONORS ARE OF THE HAIR



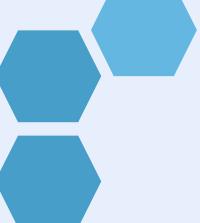
- GET EXACT MATCH OF DONOR PROVIDED PHOTOS WITH THE FACTORS CHECKED BY THE SYSTEM.



- ACCORDING TO THE FINAL OUTPUT OF THE SYSTEM, IT MUST BE CORRECTLY IDENTIFIED WHETHER IT IS SUITABLE OR UNSUITABLE HAIR FOR DONATION.

Non-Functional Requirements

- Accuracy
- Usability
- Availability
- User-friendliness
- Efficiency



Completion and Future works

Completion of the components

Collected data which needs to train the prediction models - Sub Objective 1

Identification of best architecture for transfer learning

Model Implemented to Hair identification- Main Objective 1.2

Model Implemented to identify Color of Hair – Main Objective 1.3

Model Implemented to identify Dandruff & lice of Hair - Main Objective 1.4

Model Implemented to identify Dryness of Hair - Main Objective 1.5

Model Implemented to identify bleached of Hair - Main Objective 1.6

Combine all the models to implement the final model - Sub Objective 2

Implement the Web Application with User-friendly Interface - Sub Objective 3

Future Implementations

System Integration with other functions

Host the finalized models