

Distributed Health Care Framework for Patient Health Record Management and Pharmaceutical Diagnosis

2022-110

Progress Presentation 1





Supervisor:

Mr. Jeewaka Perera
Lecturer

FACULTY OF COMPUTING | COMPUTER SCIENCE & SOFTWARE ENGINEERING

- | MSc. in Computer Science, California State University, Fresno, USA.
- | BSc. in Computer Science, California State University, Fresno, USA.
- | Minor in Mathematics, California State University, Fresno, USA.



Co-Supervisor:

Ms. Laneesha Ruggahakotuwa

Instructor

FACULTY OF COMPUTING | COMPUTER SYSTEMS ENGINEERING

- | Bachelor of Science (Honours) in Information Technology (Specialization-Cyber Security), Srilanka Institute of Information Technology
- | CCNA Routing and Switching [Academic Module]
- | Reading Master of Science (Honours) in Information Technology (Specialization-Cyber Security), Srilanka Institute of Information Technology
- | Reading Cyber Operations - CISCO CCNA



External-Supervisor:

Prof. Muditha Vidanapathirana

University of Sri Jayewardenepura · Department of Forensic Medicine
Doctor of Medicine, MA, MBBS, DLM, FFFLM (UK)

- Professor of Forensic Medicine, University of Sri Jayewardenepura, Sri Lanka
- Consultant Judicial Medicine
- He has published more than 150 publications including international and local publications. Won several awards for research and publications including presidential awards in 2014

Team Members



**Wickramarathna
W.G.M.S**
IT19004778
Group Leader

De Silva K.H.K.L
IT19006994
Member

**Lekamalage
U.L.V.M**
IT19111766
Member

Chathuranga S.J
IT19043388
Member





Introduction



The world is confronting various healthcare issues with the COVID-19 pandemic, and healthcare automation is more crucial than ever.

The pandemic has revealed the limitations of existing digital healthcare systems to handle public health emergencies while maintaining service continuity when people stay at home conducting social distancing.

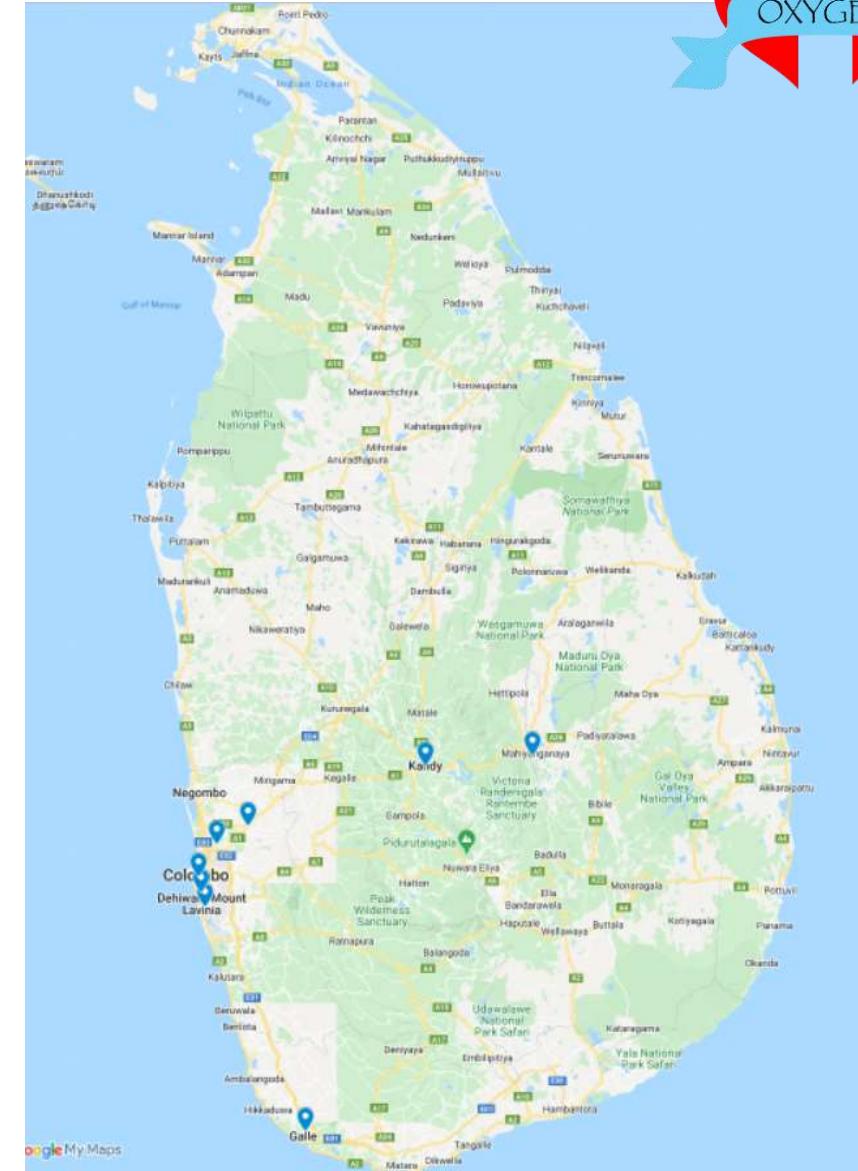




Background Study

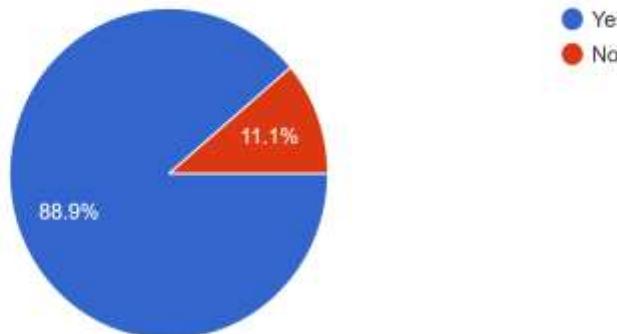
A research was conducted among Medical Practitioners and Medical Students around Sri Lanka to gather information on health care problems confronted by the medical practitioners during COVID-19

<https://forms.gle/xQ4or8naPXzj9Sw69>



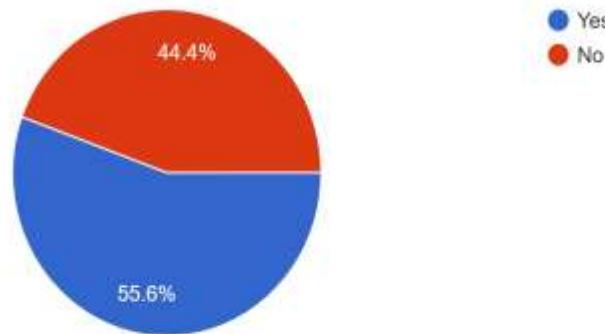
As healthcare practitioners do you face any healthcare issues during the COVID-19 pandemic?

9 responses



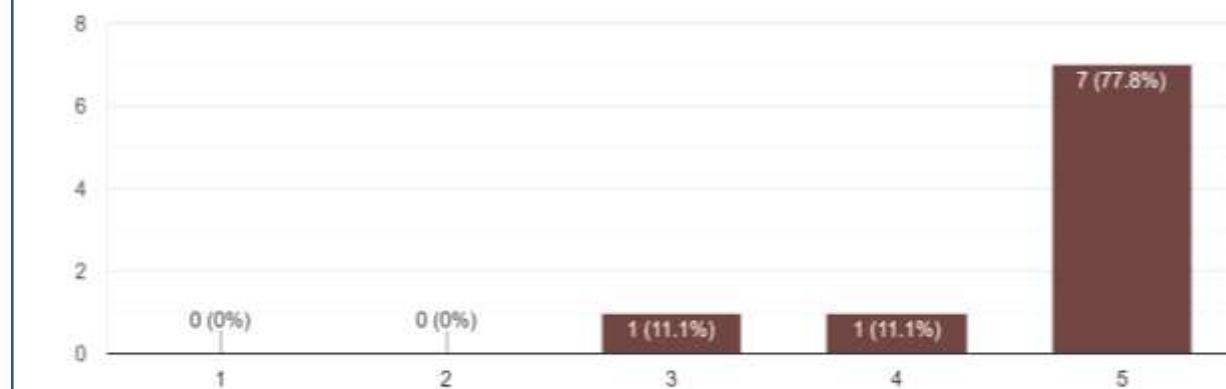
Is there any electronic health record systems in your current working hospital or have you seen one in any Sri Lankan hospital?

9 responses



Do you agree that storing, accessing and sharing patient details in electronic format is critical in the occurrence of a pandemic?

9 responses



Feedbacks and Suggestions from Medical Practitioners



“This concept must be there in our health systems. This is efficient, accurate, faster, money-saving and needs less Human Resources. Drawbacks- It can be erased and have problems regarding security. Complex process and need trained personnel. Less privacy.”

- Dr. B W B Rathninda (Acting consultant, Radiology, Base hospital Mahiyanganaya) -

“Existing healthcare systems are limited to a few wards. An automated solution is a dire need!”

- Prof Muditha Vidanapathirana (Professor, Forensic Medicine, Teaching hospital, Colombo South) -

“In existing healthcare applications, details are not up to date. It would be good if a system creates to store details of the patients in an automated way, because as a practitioner then we can get patients' details up to date.”

- Dr. Renuka Marapana (Registered medical officer, Maliban Rehabilitation Unit - Teaching hospital Karapitiya) -

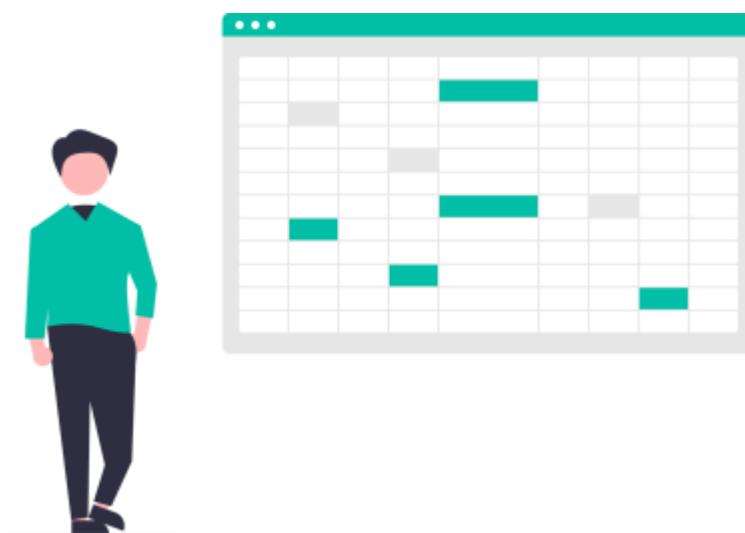
“In existing healthcare applications, accessibility to the network is a bit difficult in rural settings and introducing an automated solution will be a very appropriate intervention.”

- Dr. Uthpala Muhandiram (Community Medicine, Ministry of Health)-

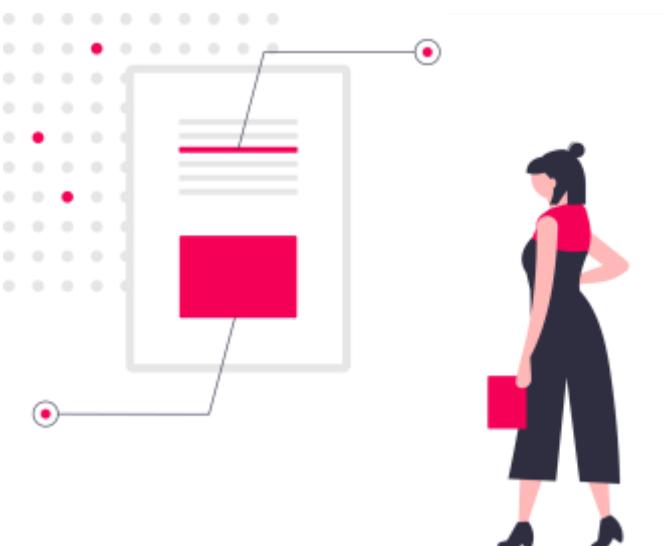
Research Problem



There is no registered population for any health care institution in Sri Lanka, as a result, there is a communication gap resulting in inadequate coordination of care.



EHR (Electronic Health Record) systems are becoming more popular to share patient details between hospitals but accessing scattered data across several EHRs while safeguarding patient privacy remains a challenge.



Research Problem Contd..



Most of these medical records and documents are in printed format and manually entering those into EHR systems is time-consuming and error prone.



Pharmaceutical error is a critical healthcare problem, but it is even riskier to visit doctors for pharmaceutical diagnosis during a pandemic. Healthcare domain is in a need for a conversational agent to give reminders to take medication on time.





Research Gap

Product Name	Securely Store Patient's Data	Access Scattered Data across several EHRs	Scan printed Medical Documents	Drug Identification and give required details(Dosage, Side effects)	Virtual Conversational Medical Chatbot
eHealth 香港特別行政區政府 HKSAR GOVT	✓	✗	✗	✗	✗
EncryptScan	✗	✗	✗	✗	✗
Drugs.com	✗	✗	✗	✗	✗
OXYGEN	✓	✓	✓	✓	✓



Not Present



Partially Present



Present



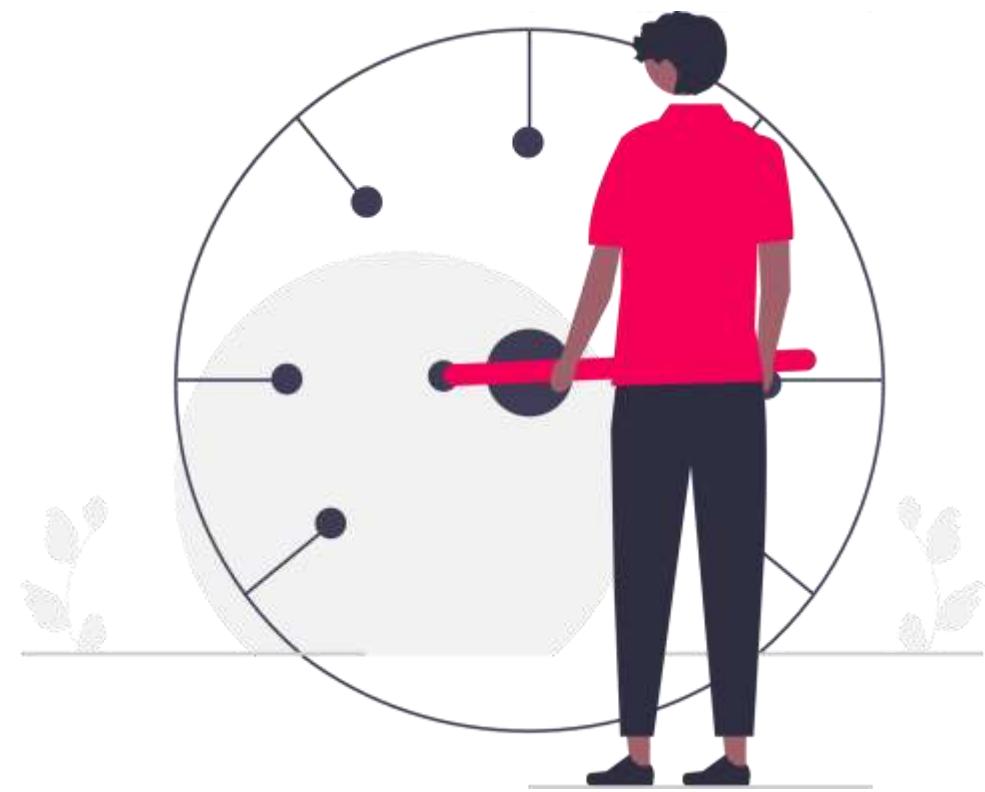
Research

**Implementing a distributed Healthcare
Framework for Patient Health Record
Management through Electronic Health
Records (EHR) and Pharmaceutical
Diagnosis.**

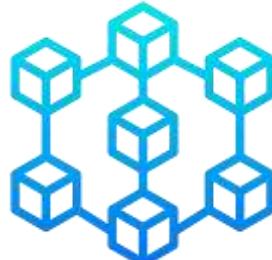


Main Objective

Solving healthcare issues during COVID-19 by providing a healthcare framework for automatically storing patients' records protecting users' privacy while providing healthcare services for people staying at home conducting social distancing.



Sub Objective s



To protect patients' data privacy while tracking/sharing healthcare records with healthcare professionals using Blockchain



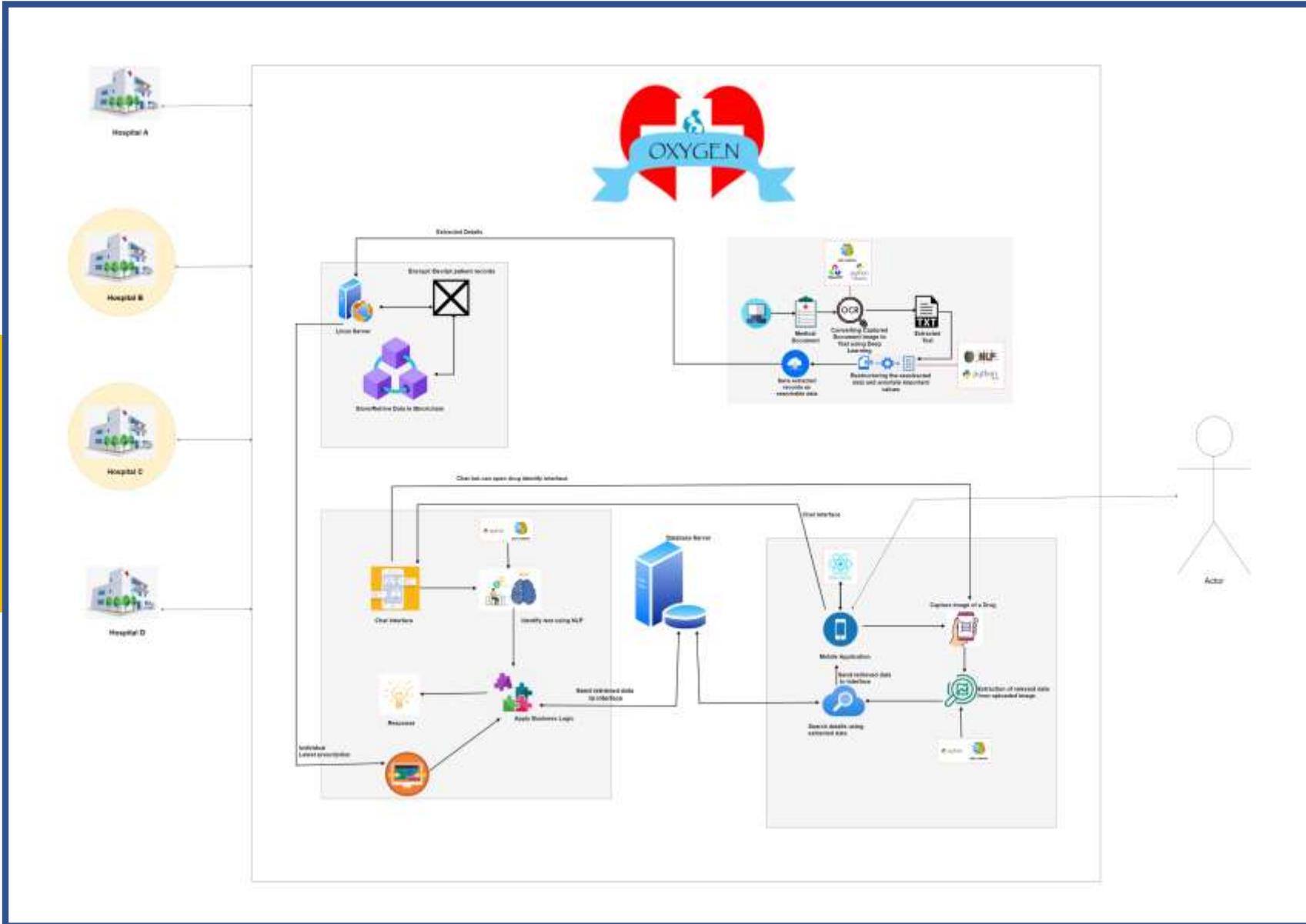
To scan and extract relevant data from Patient Clinical Laboratory Reports using Optical Character Recognition and Natural Language Processing while preventing human errors that cause when manually entering data.



To identify Drugs using Image Processing and extracting pharmaceutical data such as its side effects, dosage, etc.

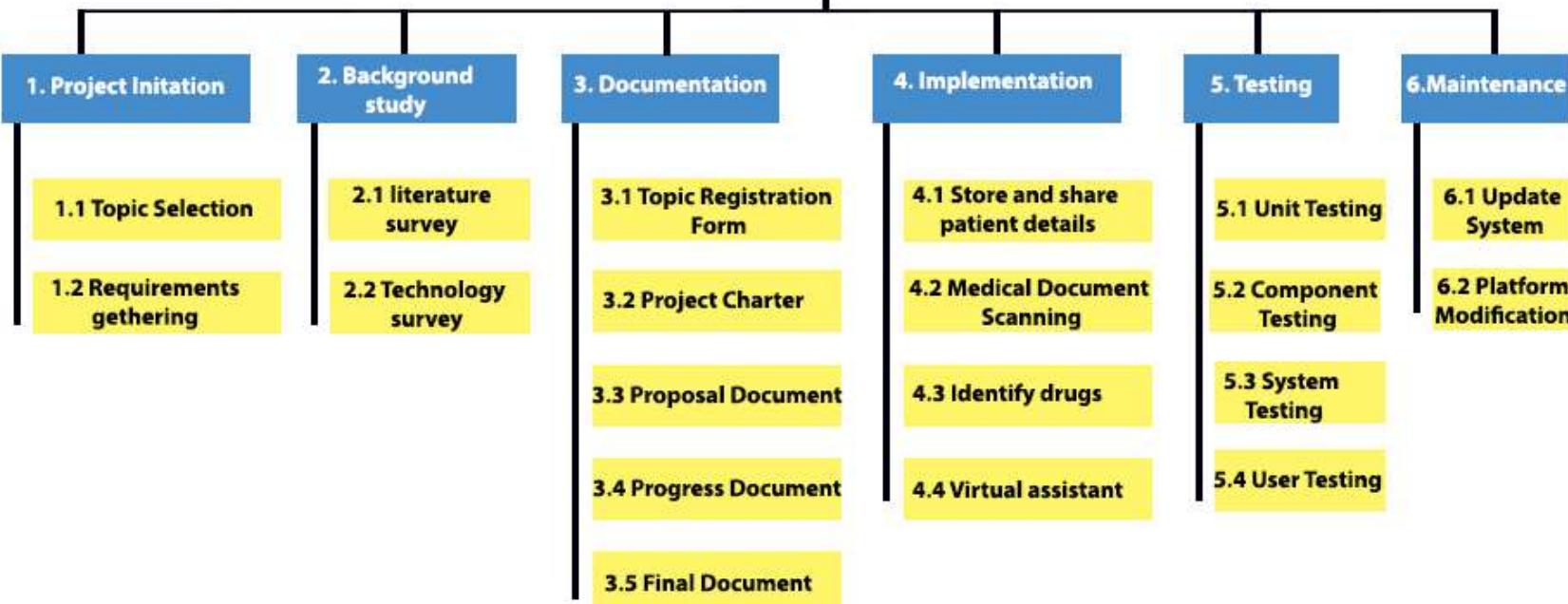


To assist patients with a smart chatbot based on Machine learning and Natural Language Processing for health care assistance

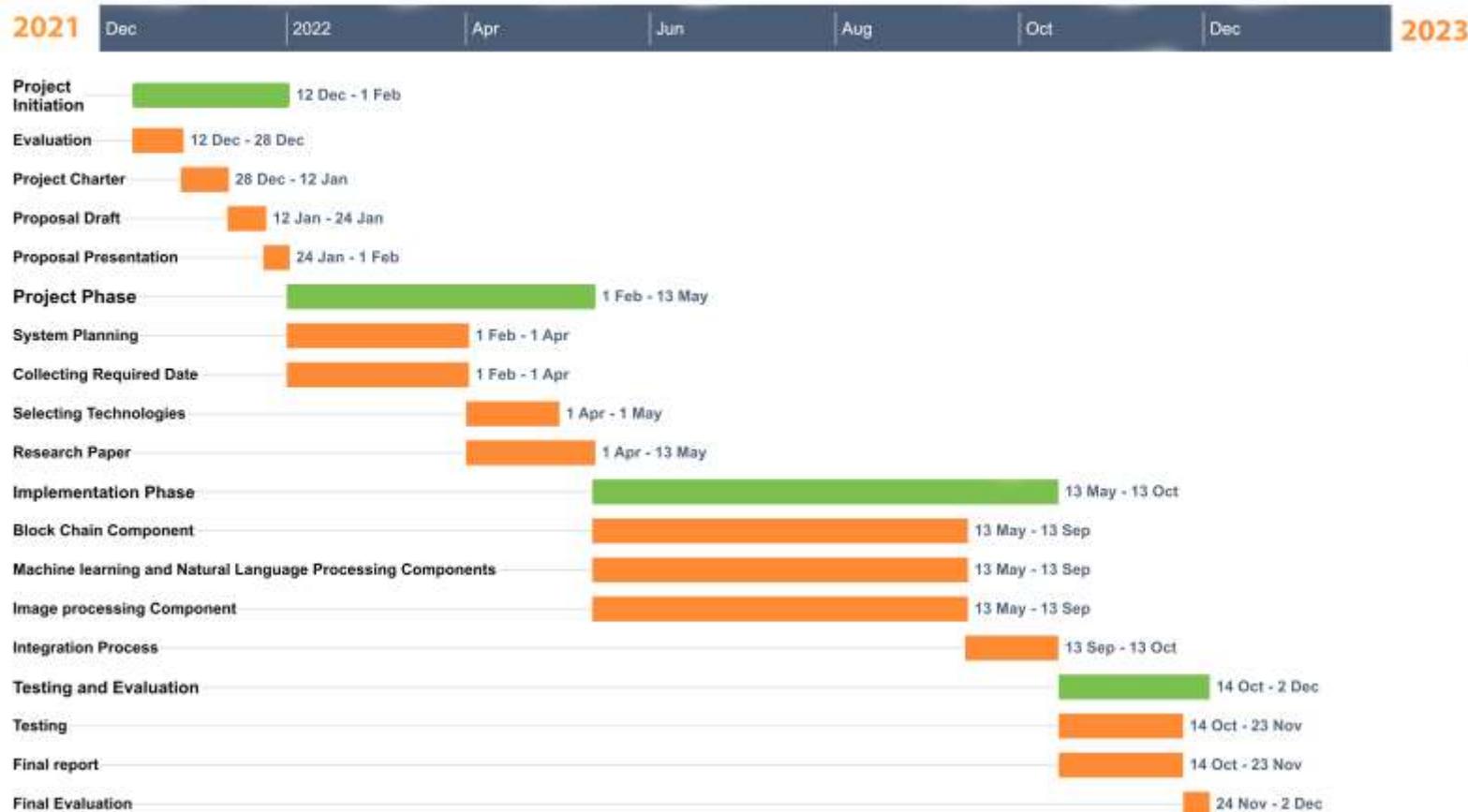


Overall System Diagram

DISTRIBUTED HEALTH CARE FRAMEWORK FOR PATIENT HEALTH RECORD MANAGEMENT AND PHARMACEUTICAL DIAGNOSIS



Work Breakdown Structure



Gantt Chart



IT19004778 | Wickramarathna W.G.M.S

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



There is no automated method to track patients' medication history

Absence of a secure method to store patient details



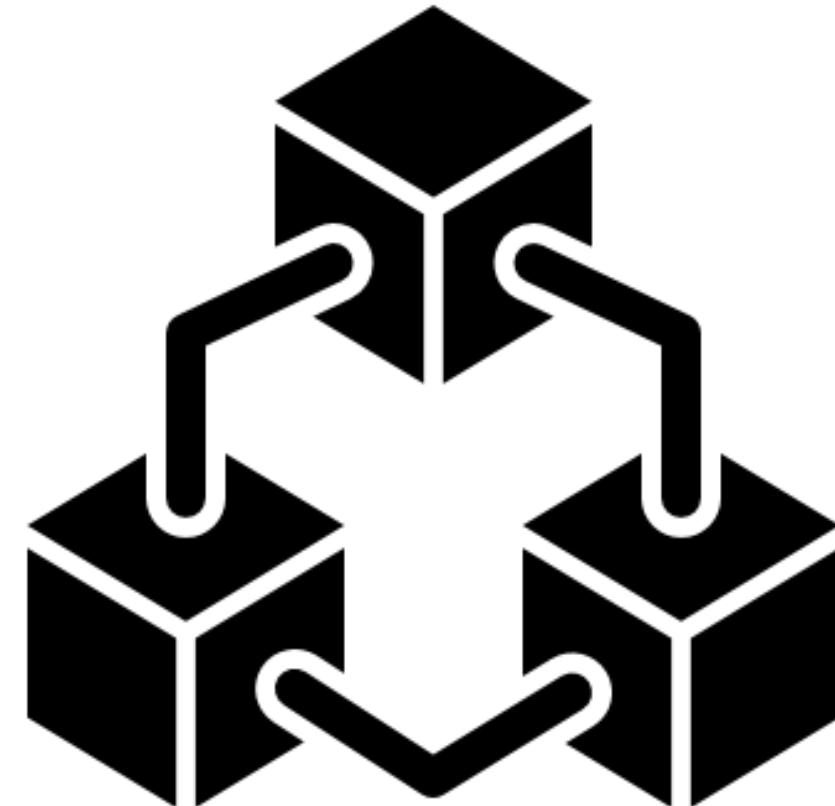
Research Problem

Sharing patient details with authorized people/organizations

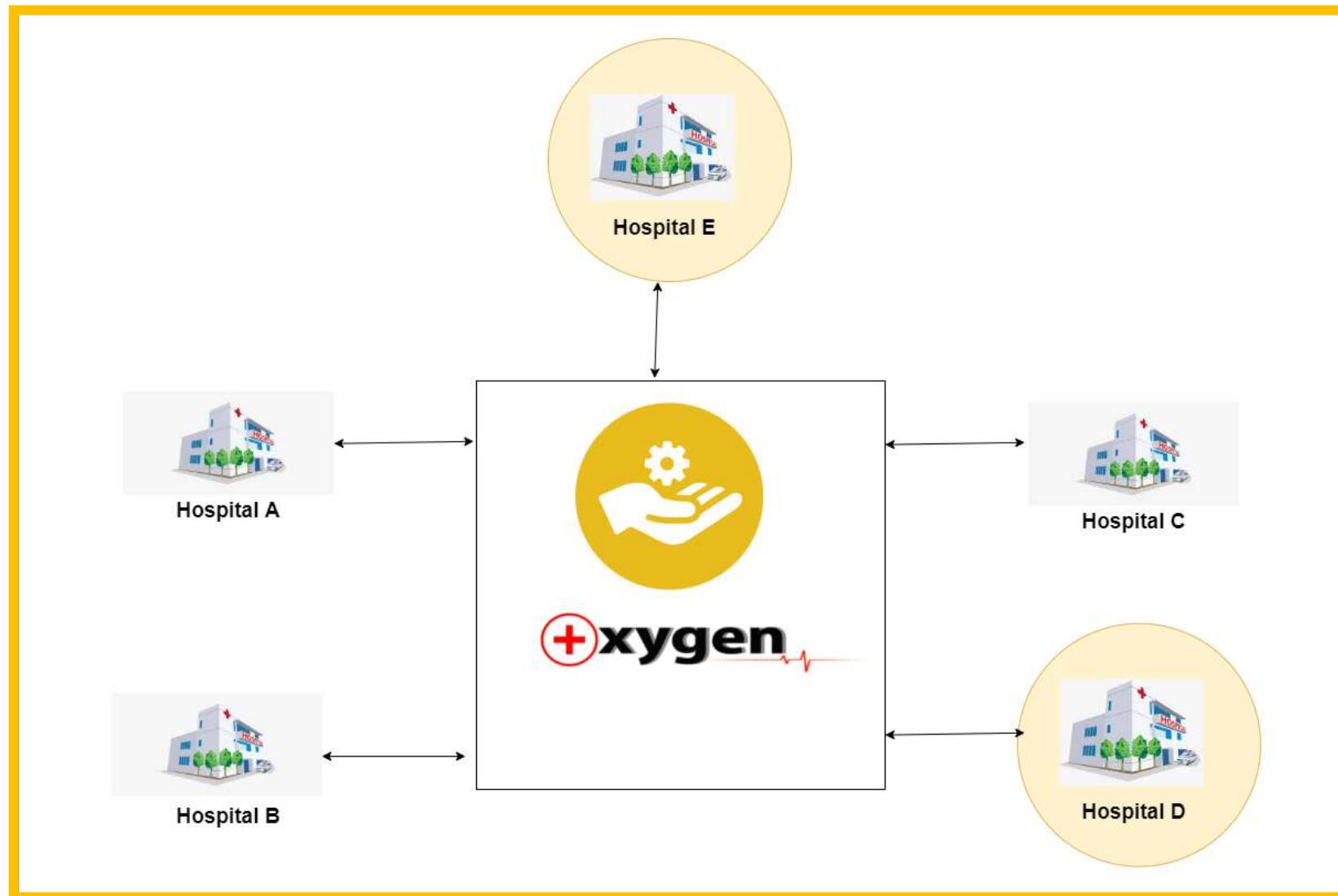
Research Gap



- There is no much health care product on Blockchain technology
- Most previous research work are for the generic usage, but our solution is designed specifically targeting the healthcare domain
- Use of smart contract technology in Blockchain for automating execution.



Research Gap Cont.





Objectives

Main Objective

- To protect patients' data privacy while tracking/sharing healthcare records with healthcare professionals.

Sub Objectives

- Prevent unauthorized access to the system data.
- Use smart contacts to automate the execution
- Prevent unauthorized apply changes to the system data
- Accessible from anywhere

Completion of Project



✓ Smart Contract Deployment

Server is up and running

MongoDB Connection success!

Attempting to deploy from account 0xE84D206CecD4Bd0B493a62BA1DfAb4A2643f9d2D
contract deployed to 0x78B45779734C4c8395d77433Cb904eC7E32cBF56

Address 0xE84D206CecD4Bd0B493a62BA1DfAb4A2643f9d2D

Overview	More Info	More
Balance: 0.144189034824043807 Ether	My Name Tag: Not Available	

Transactions

Latest 25 from a total of 87 transactions

Txn Hash	Method ⓘ	Block	Age	From ↗	To ↘	Value	Txn Fee
0xc530d10cb9481df705...	0x60808040	10618173	2 mins ago	0xe84d206cecd4bd0b49...	OUT Contract Creation	0 Ether	0.012302094796



Completion of Project



Get Patient Details By NIC

http://localhost:8070/patient/getPatientDetails/025

Save

GET http://localhost:8070/patient/getPatientDetails/025 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Cookies

Body Cookies Headers (8) Test Results

Status: 200 OK Time: 810 ms Size: 1.57 KB Save Response

Pretty Raw Preview Visualize

```
{"basicInformation": {"genaralInformation": {"fName": "amal", "mName": "amal", "lName": "amal", "profilePic": "location", "dob": "2022-01-01", "gender": "male", "maritalStatus": "married"}, "contactDetails": {"mobileNumber": "0775099995", "email": "abc@gmail.com"}, "emergencyContact": {"mobileNumber": "0775099995", "homeNumber": "0775099995"}, "otherInformation": {"height": "123", "weight": "25", "bloodGroup": "b"}}, "prescriptions": [{"doctorName": "prescription 01", "date": "2022/01/01", "summary": "medication summary", "comment": ""}, {"doctorName": "prescription 02", "date": "2022/01/01", "summary": "medication summary", "comment": ""}], "vaccinationDetails": [{"vaccineName": "vaccine 01", "date": "2022/01/01", "placeOfVaccination": "Kandy", "batchNumber": "001", "dose": "1", "time": "9.20 am", "comment": "vaccine 01"}, {"vaccineName": "vaccine 01", "date": "2022/01/01", "placeOfVaccination": "Kandy", "batchNumber": "001", "dose": "1", "time": "9.20 am", "comment": "vaccine 01"}], "AllergiesDetails": [{"date": "2022/01/01", "comment": "Allergic data"}, {"date": "2022/01/01", "comment": "Allergic data"}], "detailsOfMedicalDocuments": [{"documentName": "name_date", "location": "./ac/"}], "chronicDeseasesDetails": [{"identifiedDate": "2022/01/01", "Disease": "Disease 01", "level": "level"}, {"identifiedDate": "2022/01/01", "Disease": "Disease 01", "level": "level"}]}
```



Completion of Project



Add New Patient Record

POST <http://localhost:8070/patient/add> Send

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 "nic": "026",
2 "data": {
3     "basicInformation": {
4         "genaralInformation": {
5             "fName": "amal",
6             "mName": "amal",
7             "lName": "amal",
8             "profilePic": "location",
9             "dob": "2022-01-01",
10            "gender": "male",
11            "maritalStatus": "married"
12        },
13        "contactDetails": {
14            "mobileNumber": "0775099995",
15            "email": "abc@gmail.com"
16        },
17        "emergencyContact": {
18            "mobileNumber": "0775099995",
19            "homeNumber": "0775099995"
20        }
}
```



Completion of Project



Transaction Status

Completion of Project



Status in ledger

Overview State

① Transaction Hash: 0x93d50f882d072132b6251221f2bff91ac86f19d16cfcb927d7bc11c74f6f2b1f ⓘ

② Status: Success

③ Block: 10618209 | 5 Block Confirmations

④ Timestamp: 1 min ago (May-04-2022 03:32:52 PM +UTC)

⑤ From: 0xe84d206cecd4bd0b493a62ba1dfab4a2643f9d2d ⓘ

⑥ To: Contract 0x1d8d0b2701235c30fbe875f01f7cef593112850 ⓘ

⑦ Value: 0 Ether (\$0.00)

⑧ Transaction Fee: 0.062123373496767744 Ether (\$0.00)

⑨ Gas Price: 0.000000023813883636 Ether (23.813883636 Gwei)

⑩ Gas Limit & Usage by Txn: 2,608,704 | 2,608,704 (100%)

⑪ Gas Fees: Base: 21.313883636 Gwei | Max: 47.294833758 Gwei | Max Priority: 2.5 Gwei

⑫ Burnt & Txn Savings Fees: Burnt: 0.066601613496767744 Ether (\$0.00) | Txn Savings: 0.06125404850184448 Ether (\$0.00)

⑬ Others: Txn Type: 2 (EIP-1559) | Nonce: 84 | Position: 9

⑭ Input Data:

```
0x511DAx1iwlZGF825161jIwHjIvMDEvRDE1LCJwlgFjZU9nVmFjY21uYXRpB2410131yW5keS1sIm3hdGhoTrVtYnlyTjoiMDAx1iwlZG9jZ5161jE1lC38w11Tjjo1o54yMCBh51sInVtik11bn010132YwNjw511TmFtZ5161nZhY2NpbelUgMDE1LC3kYXR1joiMjAyM18wMS8wMS1sInBsYwH1T2ZwYwNjw5hdG1vb1i61ktbmr511wiYmFvY2hodw112Xt1011wHDE1lC3kb3N11joiMS1sInRpBwU101T51jIwIGFtZ1iwiY29tbwVudC16InZhY2NpbelUgMDE1fV0sTkFsBgVyz211c0R1dgFpbHM101t7imRhGU101TyMDTyLzAxLzAx1iwlY29tbwVudC16IkF1bGVyZ21jIGRhGU101f5x7ImRhGU101iyMDTyLzAxLzAx1iwlY29thiVuicT61kF1bGVyZ211tGRhvgE1FvRsInR1dGFnbhNP2k312G1iYwxEh2N1hWvudHm101t7imRvY3Vt2w50Tmft25161nShhWf2Gf0251sInmxvY2F0aW9uT1o1
```

View Input As: ▾



Developed Interfaces

Search61John

Basic Information

NIC
922212661V

Name
Alex B.

Age
Alex

[More Details ▶](#)

Last Medication Details

Doctor Name
Dr.Alex

Date
mm/dd/yyyy

Note
Special Note

[More Details ▶](#)

Vaccination Details

Vaccine 01

[+ Add New Vaccine Details](#)

Allergies Details

[+ Add New Allergies Details](#)

Medication History

Prescription 01

Prescription 01

[More Details ▶](#)

Chronic Diseases Condition

Prescription 01

Prescription 01

[More Details ▶](#)



Developed Interfaces

The screenshot shows a web-based application for managing patient information. The top navigation bar includes a search bar, a message center with 6 notifications, a bell icon with 1 notification, and a user profile for 'John'. The main interface is divided into several sections:

- Basic Information:** Shows NIC: 922212661V, Name: Alex B., Age: Alex, and other fields partially visible.
- Patient Information:** A modal dialog box containing fields for First Name (Alex B.), Middle Name (Alex B.), Last Name (Alex B.), Date Of Birth (Alex B.), Gender (Alex B.), Marital Status (Alex B.), NIC (Alex B.), Address (Alex B.), and two additional address fields (Alex B. and Alex B.). An 'X' button is in the top right corner of the modal.
- Allergies Details:** A section with a '+ Add New Allergy' button.
- Vaccination Details:** A section with a '+ Add New Vaccine Details' button.
- Diseases Condition:** A section showing 'Condition 01' and 'Condition 01' with a 'More Details ▶' button.



Developed Interfaces

The screenshot shows a medical application interface. At the top, there is a navigation bar with a search bar, a message center (6 notifications), a bell icon (1 notification), and a user profile for John. On the left, there are sections for 'Basic Information' (NIC: 922212661V, Name: Alex B., Age: Alex) and 'Allergies Details' (with a 'Add New Allergy' button). A central modal window titled 'Add New Vaccination' contains fields for 'Name of the Vaccine' (Alex B.), 'Place of vaccination' (Alex B.), 'Dose' (empty), 'Date' (mm/dd/yyyy), 'Batch Number' (Alex B.), 'Time' (empty), and 'Comment' (empty). Below the modal is a button labeled 'Secondary'. At the bottom of the screen, there are sections for 'Vaccination Details' (01) and 'Diseases Condition' (01), each with a 'More Details' button.

Add New Vaccination

Name of the Vaccine: Alex B.

Place of vaccination: Alex B.

Dose:

Date: mm/dd/yyyy

Batch Number: Alex B.

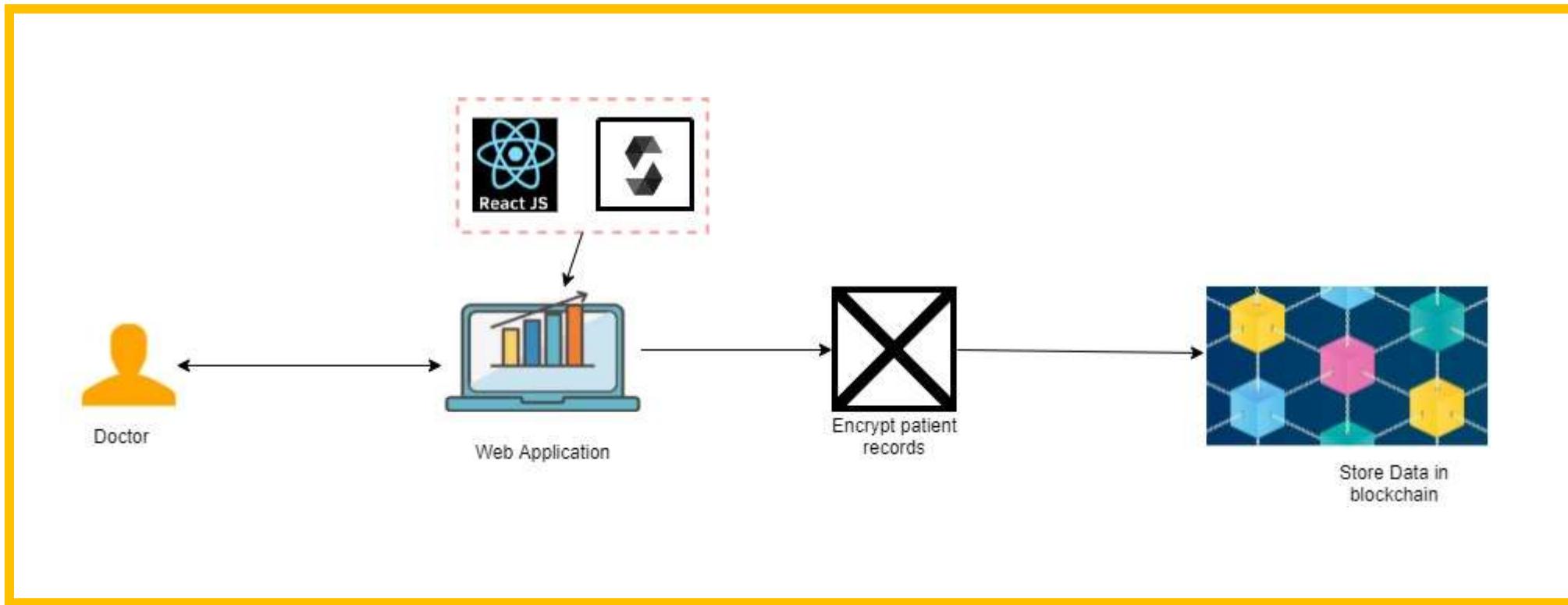
Time: --:-- --

Comment:

Secondary

More Details ▾

System Diagram



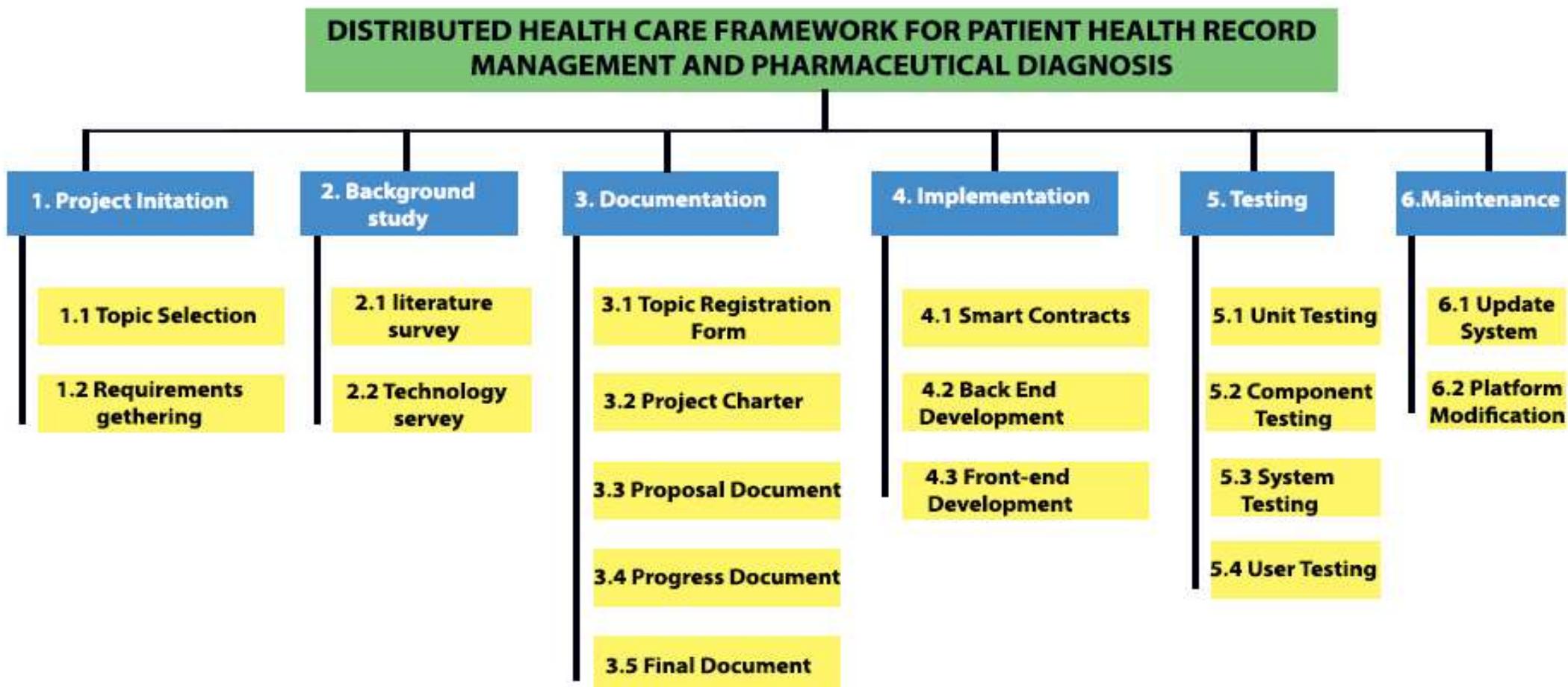
Tools and Technologies



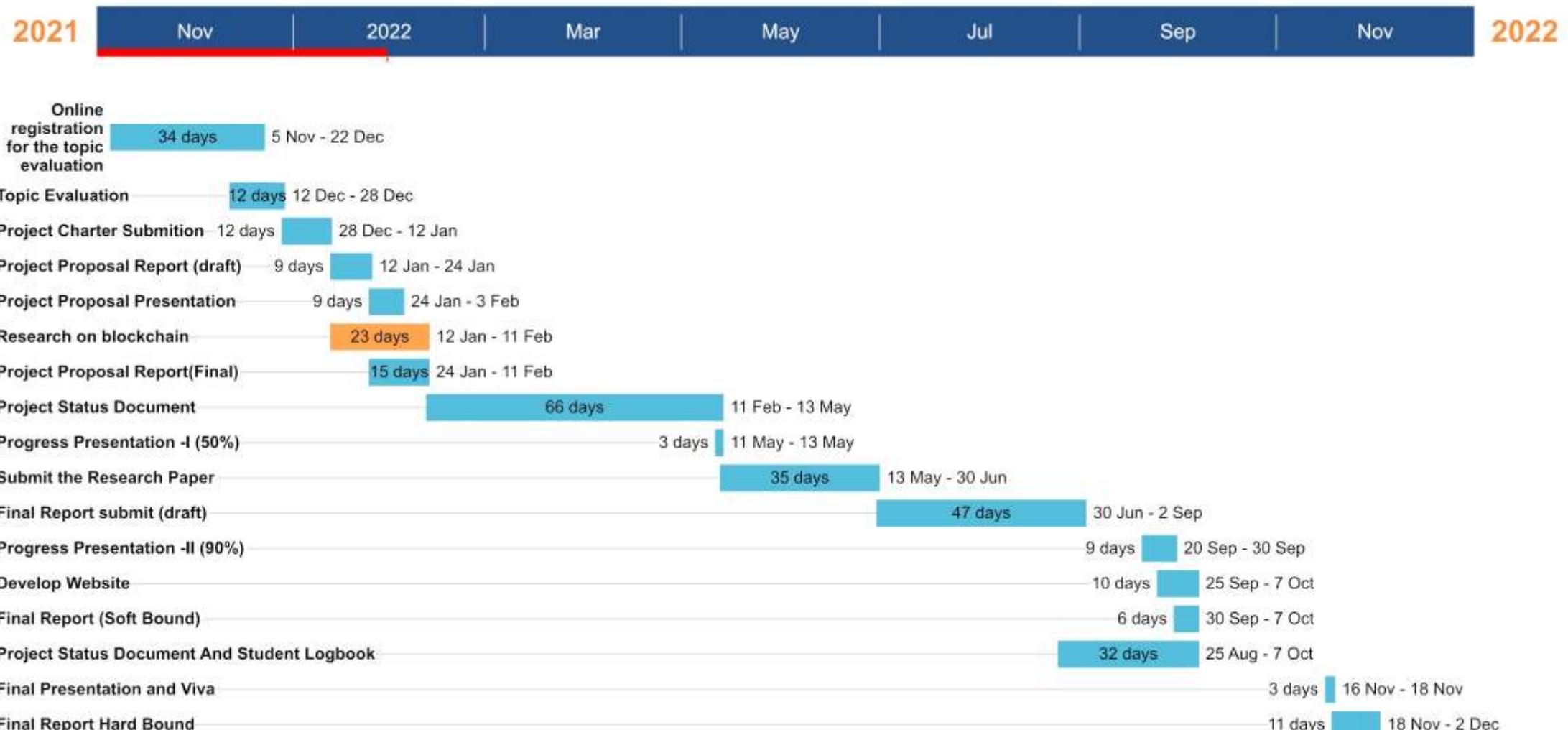
- Solidity
- Ethereum Network
- React JS
- Node.JS
- Web3JS
- INFURA.IO



Work Breakdown Structure



Gantt Chart



Project Requirements



Functional Requirements

- ✓ Store / Access patient details
- ✓ Share patient details among authorized people

Non-Functional Requirements

- ✓ Accuracy
- ✓ Speed
- ✓ Reliability
- ✓ Usability

Future Work



- Development of Role Based Access Control Protocol
- Share data among authorized people and organizations
- Integrate Frontend with Backend
- Host the server
- Testing



IT19006994 | De Silva K.H.K.L.

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





Introduction



Patient Medical Documents are an important source of information and healthcare professional use these documents to ensure continuity of care for the patient



Most of the medical documents such as clinical laboratory test reports, prescriptions from hospitals are in printed format



Converting these data into Electronic Health Records (EHR) and entering these details into blockchain often need to follow the manual data entering procedure



Research Problem



Most medical papers are in printed format and extracting information from them and transferring them to electronic health records takes a lot of time.



Manually entering these data into Blockchain is a risky task that frequently results in human errors.



As a result, an automated method for extracting textual data from printed medical records and converting them to editable and searchable formats should be introduced.

Research Gap



Reference ID	Modelled for Healthcare Domain-Specific words	Text Recognition	Important Entity Recognition	Extract Text from Low Quality Images
Research [1]	✓	✓	✗	✗
Research [2]	✗	✓	✗	✗
Research [3]	✗	✓	✗	✗
Research [5]	✓	✓	✗	✗
Our Solution	✓	✓	✓	✓

Name of the Application	Available Format / Platform	Limitations
Worldview Mobile Complete	Mobile Application designed for iPhones and iPads	Textual data will not be extracted Do not capture the important values or entities Data will be stored locally
EncryptScan by HIPAA 	Mobile Application designed for iOS & Android	Textual data will not be extracted Do not capture the important values or entities
Abby FineReader PDF 	Windows 10	Generic Document scanner Not specifically designed for healthcare-related documents
VueScan	Windows, macOS, and Linux	Generic Document scanner Not specifically designed for healthcare-related documents



Objectives



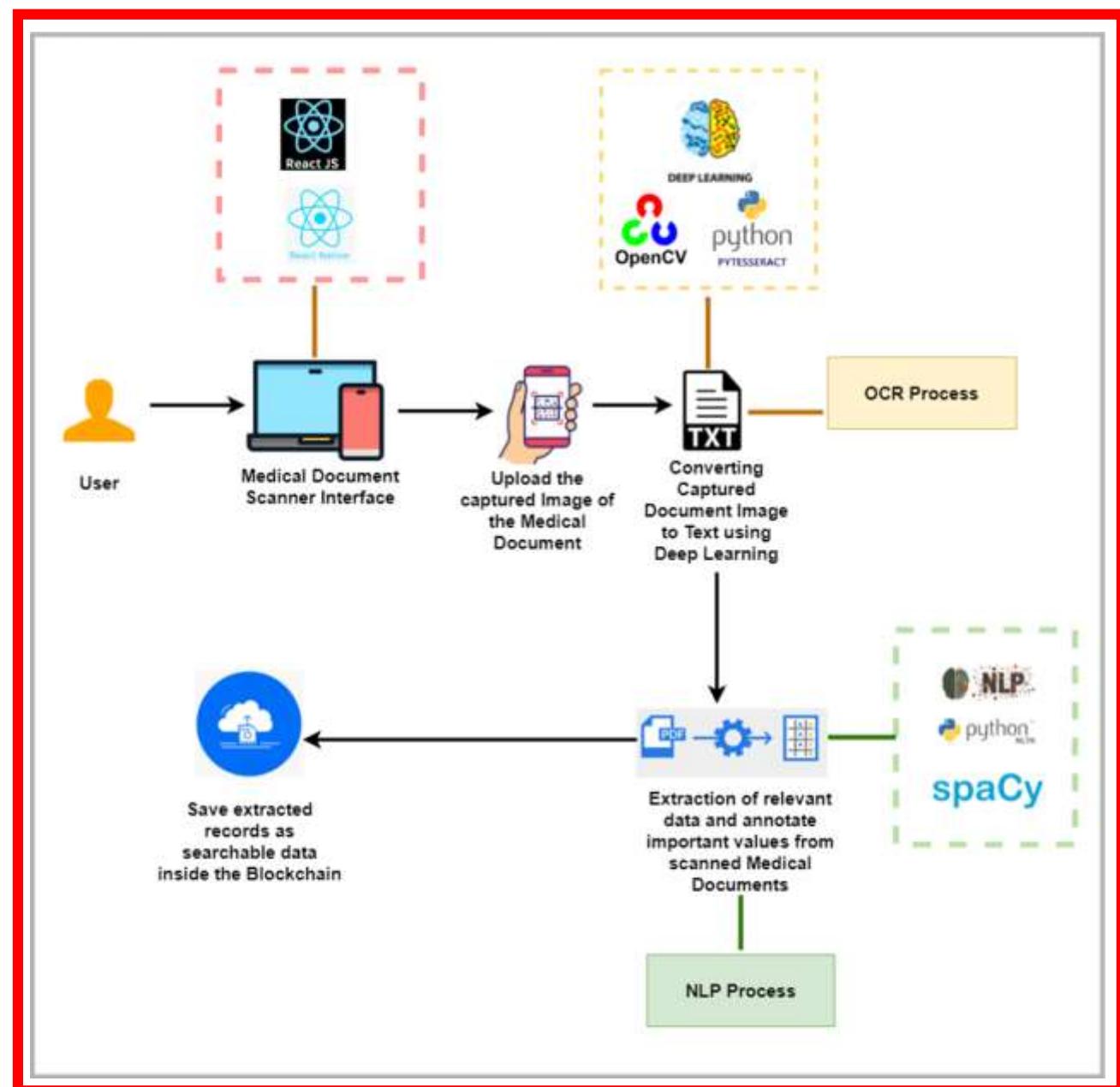
Main Objective

- To scan and extract text and important entities from Clinical Laboratory Test Reports using Optical Character Recognition and Named Entity Recognition while preventing human errors that cause when manually entering data.

Sub Objectives

- Prevent the errors that cause when manually entering data into Blockchain
- Automatically extract structured data from the captured images of the medical documents using Text Recognition
- Annotate and Recognize important entities from the recognized text

System Diagram



Methodology



First upload an Image of the Clinical Laboratory Test Reports through the Medical Document Scanner Interface



In *Computer Vision module*, we scan the document, identify the location of text and finally extract text from the image. Convert the text in the captured image using techniques in Optical Character Recognition



Then in *Natural language processing*, we will extract the entities from the text and do necessary text cleaning and parse the entities form the text.



Dataset to Train the Spacy Model

1. This Data set consist of images of Clinical Laboratory Test Reports issued by 24 laboratories in Egypt.

Link to download the Data set:

<https://data.mendeley.com/datasets/bygfmk4rx9/2>

2. A google form to collect Data from real users:

<https://forms.gle/YnwdShAsw1Mgqa6o7>

Section 1 of 2

Distributed Health Care Framework for Patient Health Record Management and Pharmaceutical Diagnosis

I am a final year Software Engineering Undergraduate at the Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, Sri Lanka. I am conducting a research to develop a Medical Document Scanner to intelligently extract data from Medical Documents and I need to gather images of Clinical Laboratory Test Reports to train our Machine Learning Models. Please spare 5 minutes of your valuable time to participate in the survey and share with us the images of the Clinical Laboratory Test Reports that you have. The information is being gathered solely for research purposes, and your responses are greatly welcomed and the sensitive data are securely handled with care.

For more info please contact:

Name: Kithmini De Silva (Software Engineering Undergraduate)
Address: Sri Lanka Institute of Information Technology New Kandy Road, Malabe, Sri Lanka
Email: desilvakithmini@gmail.com

Do you have any Clinical Laboratory Test Reports with you?

Yes

No

Would you like to share those Lab Test Reports with us to develop a Healthcare application for the healthcare community in Sri Lanka?

Yes

No

After section 1 Continue to next section

Section 2 of 2

Please take a photograph of your Clinical Laboratory Test Report

Description (optional)

Please upload the Image of the Clinical Lab Test Report that you have. *

Add file View folder

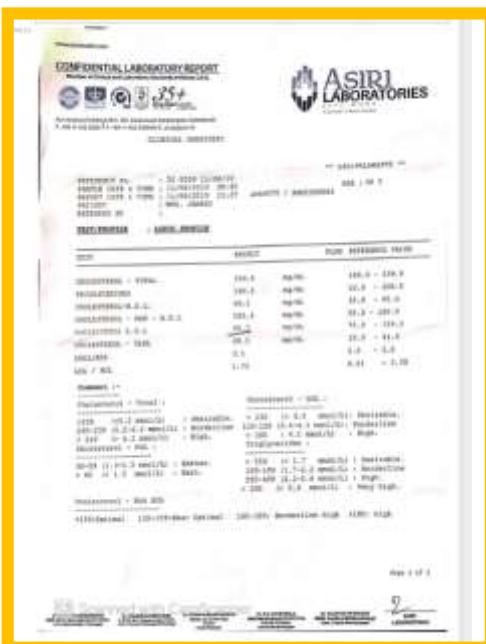
Completion of Project – 50%



✓ Software Installations

Tesseract-OCR, Pytesseract, Spacy Libraries, numpy, Pandas, matplotlib, OpenCV-python, Jupyter

✓ Load Medical Document using OpenCV and extract Text from Image



Extract Text from Image

```
In [6]: text_cv = pytesseract.image_to_string(img_cv)
In [7]: print(text_cv)
```

TEST RESULT	FLAG	REFERENCE	VALUE
140.0	-	239.0	

CHOLESTEROL - TOTAL 154.5 mg/dL
TRIGLYCERIDES 100.5 mg/dL 10.0 - 200.0
CHOLESTEROL-H.D.L. 49.1 mg/dL 35.0 - 85.0
CHOLESTEROL - NON - H.D.L 105.4 mg/dL 55.0 - 189.0

JOLESTEROL L.D.L 85.3 mg/dL 75.0 - 159.0
CHOLESTEROL 1 35.3, 9.
CHOLESTEROL - VLDL 20.1 mg/dL 10.0 - 41.0
CHOL/HDL 3.1 2.0 - 5.0
LDL / HDL 1.73 0.01 - 3.30
Comment :~

Completion of Project



Convert Image to Text to Data frames

Image to Data

```
In [10]: data = pyteseract.image_to_data(img_cv)

In [11]: # data.split('\n')
dataList = list(map(lambda x: x.split('\t'), data.split('\n')))
df = pd.DataFrame(dataList[1:], columns=dataList[0])

In [12]: df.head(10)
Out[12]:
  level page_num block_num par_num line_num word_num left top width height conf  text
0     1      1       0       0       0       0  0  3  0  1002  1600   -1
1     2      1       1       0       0       0  0  45  40  139  17   -1
2     3      1       1       1       0       0  0  45  40  139  17   -1
3     4      1       1       1       1       1  0  45  40  139  17   -1
4     5      1       1       1       1       1  1  45  40  139  17  38  wwwasirihealth.com
5     2      1       2       0       0       0  0  667  103  40  51   -1
6     3      1       2       1       0       0  0  667  103  40  51   -1
7     4      1       2       1       1       1  0  667  103  40  24   -1
8     5      1       2       1       1       1  1  667  103  40  24  95
9     4      1       2       1       2       0  0  667  127  10  27   -1

In [13]: print(data)
level page_num block_num par_num line_num word_num left top width height conf  text
1     1      1       0       0       0       0  1082  1600   -1
2     1      1       0       0       45      40  139  17   -1
3     1      1       1       0       45      40  139  17   -1
4     1      1       1       1       45      40  139  17   -1
5     1      1       1       1       45      40  139  17   -1
6     1      1       1       1       45      40  139  17   -1
7     1      2       0       0      667      183  40  51   -1
8     1      2       1       0      667      183  40  51   -1
9     1      2       1       1      667      183  40  24   -1
10    1      2       2       0      697      127  16  27   -1
11    1      2       2       1      697      127  16  27  95
12    1      3       0       0      47       91  438  28   -1
13    1      3       1       0      47       91  438  28   -1
14    1      3       1       1      47       91  176  27  96  CONFIDENTIAL
15    1      3       1       2      230      99  152  28  96  LABORATORY
16    1      3       1       3      387      99  98  29  93  REPORT
17    1      4       0       0      75      124  379  17   -1
```



Draw Bounding Box around each word





Completion of Project



**Extract Text
and Data from
Medical
Documents**



**Save Data in
CSV**



**Labelling Data
in CSV to train
Spacy Model**

B-DATE

O

B-AGE

I-AGE

O

B-PATIENTNAME

I-PATIENTNAME

O

B-TEST

I-TEST

O

B-RESULT

I-RESULT

O

B-COMMENTS

I-COMMENTS

A	B	C	D	E	F
337	10.jpeg	:	O		
338	10.jpeg	MRS.	B-PATIENTNAME		
339	10.jpeg	JANAKIE	I-PATIENTNAME		
340	10.jpeg	GUNAWAFI	PATIENTNAME		
341	10.jpeg	REFERRED	O		
342	10.jpeg	:	O		
343	10.jpeg	AGE	O		
344	10.jpeg	:	O		
345	10.jpeg	50	B-AGE		
346	10.jpeg	Y	I-AGE		
347	10.jpeg		O		
348	10.jpeg		O		
349	10.jpeg		O		
350	10.jpeg		O		
351	10.jpeg		O		
352	10.jpeg		O		
353	10.jpeg		O		
354	10.jpeg	TEST	O		
355	10.jpeg	RESULT	O		
356	10.jpeg	FLAG	O		
357	10.jpeg	REFERENC	O		
358	10.jpeg	VALUE	O		
359	10.jpeg	SERUM	B-RESULT		
360	10.jpeg	ALT	I-RESULT		
361	10.jpeg	(S.G.P.T.)	I-RESULT		

BIO Tagging

B - Beginning

I - Inside

O - Out

Completion of Project



```
In [12]: allDocumentsData = []
for document in documents:
    documentData = []
    grouparray = group.get_group(document)[['text','tag']].values
    content = ''
    annotations = {'entities':[]}
    start = 0
    end = 0
    for text, label in grouparray:
        text = str(text)
        stringLength = len(text) + 1

        start = end
        end = start + stringLength

        if label != 'O':
            annot = (start,end-1,label)
            annotations['entities'].append(annot)

        content = content + text + ' '

    documentData = (content,annotations)
    allDocumentsData.append(documentData)
```

```
In [13]: allDocumentsData
[(520, 522, 'I-COMMENTS'),
 (523, 526, 'I-COMMENTS'),
 (527, 534, 'I-COMMENTS'))],
('laboratories 23082021 us fe pa 24082021 que 1 ay 7 al 27 y microbiology unit ref range stool examination physical examination colour brown consistency semiformed blood absent mucus absent worms absent microscopic examination pus cells 01 hpf 05 rbc 01 hpf 05 eggs oabsent giardia lamblia absent entamoeba histolytica absent undigested food muscle fibres fat others absent note aid pice cle 5 a yall 48 res pt branch line reviewed by page 1 of 1 printed by system pm printed on 22102021 ',
{'entities': [(31, 39, 'B-DATE'),
 (54, 56, 'B-AGE'),
 (57, 58, 'I-AGE'),
 (87, 92, 'B-TEST'),
 (93, 104, 'I-TEST'),
 (126, 132, 'B-RESULT'),
 (133, 138, 'I-RESULT'),
 (139, 150, 'B-RESULT'),
 (151, 161, 'I-RESULT'),
 (162, 167, 'B-RESULT'),
 (168, 174, 'I-RESULT'),
 (175, 180, 'B-RESULT'),
 (181, 187, 'I-RESULT')]
```

- ✓ Load and convert data into Spacy format
- ✓ Cleaning Text and convert Data into Spacy format



Completion of Project

Trained Spacy Pipeline

jupyter Logout

```
[2022-05-04 00:09:18,122] [INFO] Initialized pipeline components: ['tok2vec', 'ner']
✓ Initialized pipeline

===== Training pipeline =====
i Pipeline: ['tok2vec', 'ner']
i Initial learn rate: 0.001
E   #      LOSS TOK2VEC  LOSS NER  ENTS_F  ENTS_P  ENTS_R  SCORE
---  ---  -----  -----  -----  -----  -----  -----
  0     0      0.00    50.30    0.00    0.00    0.00    0.00
  0    200    202.68   7273.06   79.82   76.62   83.31    0.80
  1    400    361.65   2899.33   87.54   86.88   88.21    0.88
  3    600    373.46   1773.07   87.15   86.04   88.28    0.87
  4    800    466.86   1756.66   88.77   88.27   89.28    0.89
  6   1000    493.15   1371.85   88.45   87.78   89.13    0.88
  8   1200    612.65   1433.24   87.98   86.64   89.36    0.88
 11   1400    690.27   1391.24   89.02   88.08   89.97    0.89
 15   1600   2878.89   1422.90   89.19   88.96   89.43    0.89
 19   1800   911.98   1519.50   88.74   87.90   89.59    0.89
 25   2000   926.61   1589.84   88.77   89.18   88.36    0.89
 32   2200   934.71   1741.64   89.35   88.45   90.28    0.89
 41   2400   997.61   1781.44   89.29   89.84   88.74    0.89
 49   2600   990.80   1751.58   89.09   89.43   88.74    0.89
 58   2800   996.74   1699.21   89.30   89.78   88.82    0.89
 67   3000   992.15   1658.52   88.57   88.16   88.97    0.89
 76   3200   992.59   1559.82   88.58   86.67   90.58    0.89
 84   3400   921.29   1588.36   88.41   89.55   87.29    0.88
 93   3600   791.82   1488.97   88.62   88.12   89.13    0.89
102   3800   924.88   1466.16   89.25   89.52   88.97    0.89
✓ Saved pipeline to output directory
output\model-last
PS D:\Y4S1\Medical Document Scanner\Notes\MedicalDocumentScanner>
```

Completion of Project



Final Prediction Pipeline

The Jupyter Notebook code in the left window:

```
import Predictions as pred

In [15]: img = cv2.imread('../data/17.jpeg')
cv2.namedWindow('original',cv2.WINDOW_NORMAL)
cv2.imshow('original',img)
cv2.waitKey(0)
cv2.destroyAllWindows()

In [*]: img_results, entities = pred.getPredictions(img)

print(entities)
cv2.namedWindow('predictions',cv2.WINDOW_NORMAL)
cv2.imshow('predictions',img_results)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The laboratory report from Asiri Laboratories (right) shows the following test results:

TEST	RESULT	FLAG	REFERENCE VALUE
TOTAL WHITE CELL COUNT	3.3	10 ⁹ /L	8.0 - 11.0
DIFFERENTIAL COUNT			
NEUTROPHILS	37.5	%	2.0 - 7.5
LYMPHOCYTES	30.6	%	1.0 - 3.0
MONOCYTES	1.9	%	0.2 - 1.0
EOSINOPHILS	6.6	%	0.0 - 0.9
BASOPHILS	0.2	%	0.0 - 0.1
HAEMOGLOBIN AND RED CELL PARAMETERS			
HAEMOGLOBIN	12.2	g/dL	11.0 - 14.0
RED BLOOD CELLS	4.26	10 ¹² /L	3.8 - 4.8
MEAN CELL VOLUME	84.3	fL	76.0 - 96.0
HAEMATOCRIT	35.4	L/L (V)	35.0 - 45.0
MEAN CELL HAEMOGLOBIN	29.0	pg	27.0 - 33.0
MCHC CONCENTRATION	34.5	g/dL	32.0 - 36.0
RED CELLS DISTRIBUTION WIDTH	13.3	%	12.0 - 15.0
PLATELET COUNT	252	10 ⁹ /L	150 - 400



Developed Interfaces

The screenshot shows a web-based application for scanning medical documents. At the top, there's a navigation bar with icons for user profile, search, notifications (6), and other account details. The main content area has a heading "Scan the Medical Documents" with a brief description of what the service does. Below this is a form where users can "Select an Image" and upload files. A large blue button labeled "Wrap Document and Extract Text" is prominently displayed. In the center, a preview window shows a sample document from "GNU Solidario Hospital". The document includes patient information (Name: Ana Betz, Date: 2011-08-25 08:32, Doctor: Cameron Cordara), demographic details (Age: 25y 10m 26d, Sex: Female, Test id: B165AAAF4), and a "COMPLETE BLOOD COUNT" section with a table. The table has columns for Test Name (Hemoglobin), Result (12), Normal Range (11.8 - 16.0), and Units (g/dL).

Scan the Medical Documents

Medical Document Scanner is a free online OCR(Optical Character Recognition) service and can analyze the text in the image you upload. The function of OCR is to help digitize documents in a timely manner and fashion. Instead of retyping files or manually typing out text within image files, you can use OCR to automatically scan and recognize text within your image or your scanned document.

Select an Image

Choose Files: No file chosen

Upload

Wrap Document and Extract Text

GNU Solidario Hospital
Autovía del Norte 12485
Las Palmas de Gran Canaria
Spain

LABORATORY REPORT

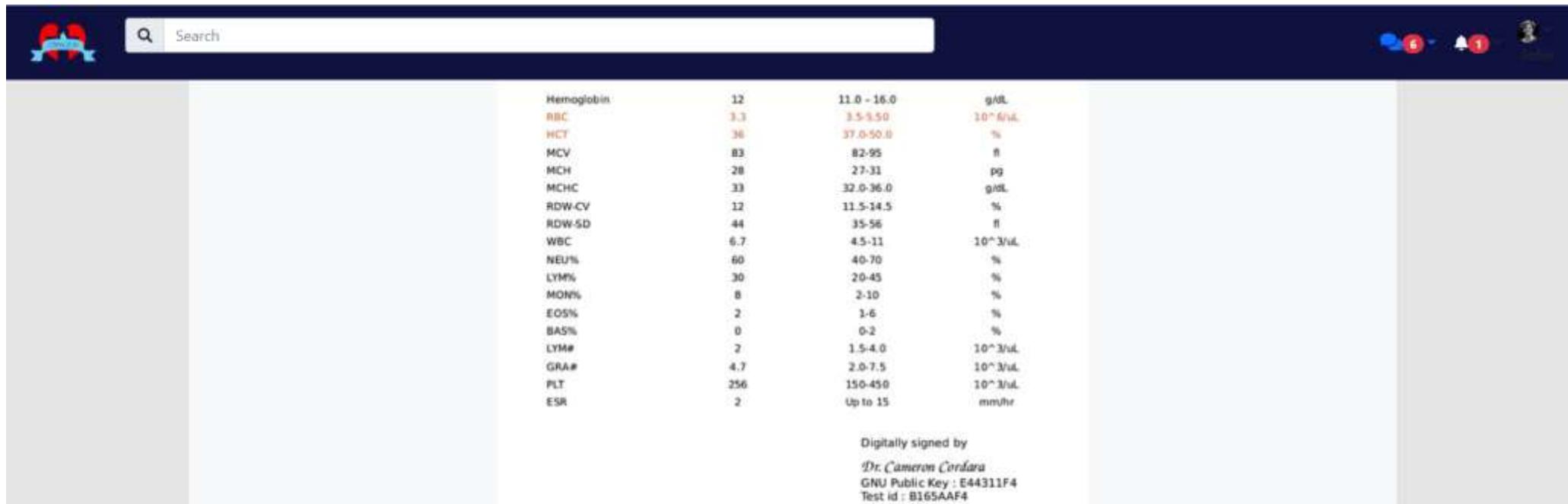
Name	Test Name	Patient ID	Result	Normal Range	Units
Doctor	Hemoglobin	BAC001	12	11.8 - 16.0	g/dL
Date					

COMPLETE BLOOD COUNT

Test Name	Result	Normal Range	Units
Hemoglobin	12	11.8 - 16.0	g/dL



Developed Interfaces



The screenshot shows a medical test result interface. At the top, there is a navigation bar with a search bar and various icons. Below the bar is a table of blood parameters:

Hemoglobin	12	11.0 - 16.0	g/dL
RBC	3.3	3.5-5.50	10 ⁶ /uL
HCT	36	37.0-50.0	%
MCV	83	82-95	fL
MCH	28	27-31	pG
MCHC	33	32.0-36.0	g/dL
RDW-CV	12	11.5-14.5	%
RDW-SD	44	35-56	fL
WBC	6.7	4.5-11	10 ³ /uL
NEU%	60	40-70	%
LYM%	30	20-45	%
MON%	8	2-10	%
EOS%	2	1-6	%
BAS%	0	0-2	%
LYM#	2	1.5-4.0	10 ³ /uL
GRA#	4.7	2.0-7.5	10 ³ /uL
PLT	256	150-450	10 ³ /uL
ESR	2	Up to 15	mm/hr

Below the table, a digital signature is displayed:

Digitally signed by
Dr. Cameron Cordara
GNU Public Key : E44311F4
Test id : B165AAF4

The interface is divided into three numbered sections at the bottom:

- 1 Choose File
Choose an Image file from the device
- 2 Upload Image
Upload the selected Image for scanning
- 3 Wrap Document and Extract Text
Click on wrap document button to extract text



Developed Interfaces

Search

BAS%	0	0-2	%	BAS%	0	0-2	%
LYM#	2	1.5-4.0	$10^3/\mu\text{L}$	LYM#	2	1.5-4.0	$10^3/\mu\text{L}$
GRA#	4.7	2.0-7.5	$10^3/\mu\text{L}$	GRA#	4.7	2.0-7.5	$10^3/\mu\text{L}$
PLT	256	150-450	$10^3/\mu\text{L}$	PLT	256	150-450	$10^3/\mu\text{L}$
ESR	2	Up to 15	mm/hr	ESR	2	Up to 15	mm/hr

Digital signature by
Dr. Cameron Cordara
GNU Public Key : E44311F4
Test id : B165AAF4

Digital signature by
Dr. Cameron Cordara
GNU Public Key : E44311F4
Test id : B165AAF4

Entities	Extracted Text
<input type="checkbox"/> Patient Name	
<input type="checkbox"/> Test/Profile	
<input type="checkbox"/> Age	
<input type="checkbox"/> Report Date & Time	
<input type="checkbox"/> Reference No.	
<input type="checkbox"/> Test	
<input type="checkbox"/> Test	

Save Details

Tools and Technologies



1. Text Recognition

- Optical Character Recognition using Deep Learning
 - ✓ OpenCV
 - ✓ Pytesseract
 - ✓ Numpy
 - ✓ pandas
 - ✓ matplotlib
 - ✓ pillow
 - ✓ Jupyter



NumPy



2. Named Entity Recognition and extract important values

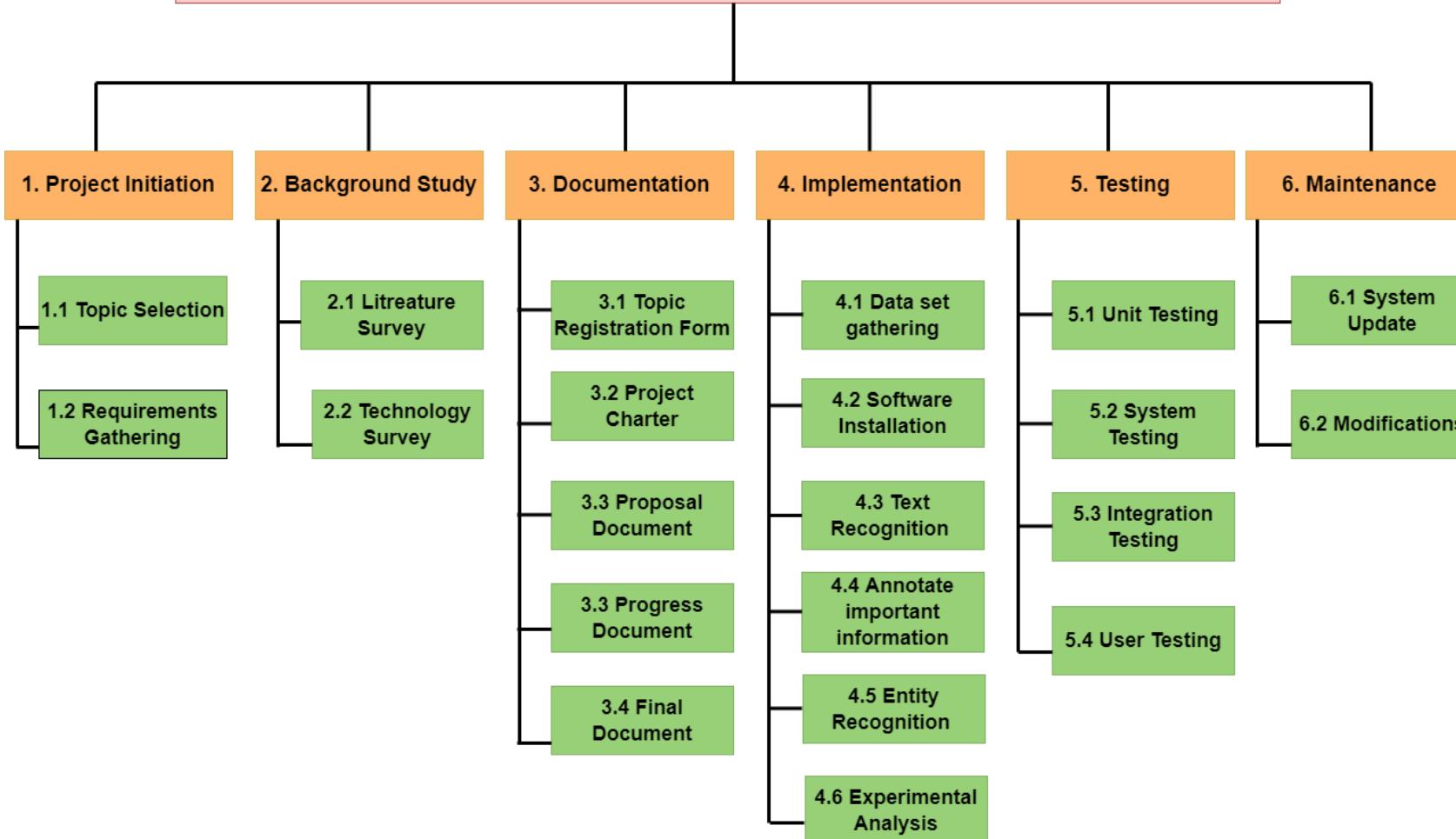
- Natural Language Processing
- Spacy Technology



Work Breakdown Structure



DISTRIBUTED HEALTH CARE FRAMEWORK FOR PATIENT HEALTH RECORD
MANAGEMENT AND PHARMACEUTICAL DIAGNOSIS



Gantt Chart

Task Name	Project Timeline												
	Dec	Jan	Feb	March	April	May	June	July	August	Sep	Oct	Nov	Dec
Description													
Project Initiation													
Topic Registration (12th Dec 2021)													
Topic Evaluation (28th Dec 2021)													
Project Charter Submission (12th Jan 2022)													
Project Proposal Presentation (3rd Feb 2022)													
Project Phase													
System Planning													
Data Gathering													
Implementation Phase													
Software Installation													
Load document using OpenCV and PIL													
Pytesseract: Extract Text from Image													
Image to Text to Dataframe													
Draw Bounding Box around each word													
Extract Text and Data from Medical Documents													
Save data in CSV													
Labeling Data using manual BIO Tagging													
Spacy Training Data													
Cleaning Text, convert data into Spacy format													
Train Named Entity Recognition Model													
Predictions													
Develop Document Scanner Web App													
Host the Server													
Testing phase and Evaluation													
Project Status Document													
Final Presentation and Viva													
Final Report and Research Paper													
Final Evaluation													

Project Requirements



Functional Requirements

-  Extract Textual Data from Clinical Laboratory Reports
-  Annotate and Extract important values from Clinical Laboratory Reports

Non-Functional Requirements

-  Accuracy
-  Speed
-  Reliability
-  Usability

Future Work



- Apply Image Pre-Processing Techniques to Extract Text from Low Quality Images
- Complete the Backend components and integrate all into one python file.
- Develop the Document Scanner Web App
- Integrate Frontend with Backend
- Host the server
- Testing

REFERENCE



S

- [1] Xue, W., Li, Q. and Xue, Q., 2019. Text detection and recognition for images of medical laboratory reports with a deep learning approach. *IEEE Access*, 8, pp.407-416.
- [2] Sahu, N. and Sonkusare, M., 2017. A study on optical character recognition techniques. *The International Journal of Computational Science, Information Technology and Control Engineering*, 4, pp.1-14.
- [3] Karthikeyan, S., de Herrera, A.G.S., Doctor, F. and Mirza, A., 2021. An OCR Post-correction Approach using Deep Learning for Processing Medical Reports. *IEEE Transactions on Circuits and Systems for Video Technology*.
- [4] D'hondt, E., Grouin, C. and Grau, B., 2017, November. Generating a training corpus for OCR post-correction using encoder-decoder model. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 1006-1014).
- [5] Moon, S., Liu, S., Chen, D., Wang, Y., Wood, D.L., Chaudhry, R., Liu, H. and Kingsbury, P., 2019. Salience of medical concepts of inside clinical texts and outside medical records for referred cardiovascular patients. *Journal of Healthcare Informatics Research*, 3(2), pp.200-219.
- [6] Özgen, A.C., Fasounaki, M. and Ekenel, H.K., 2018, May. Text detection in natural and computer-generated images. In *2018 26th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). IEEE.
- [7] Widiastuti, N.I., 2019, November. Convolution neural network for text mining and natural language processing. In *IOP Conference Series: Materials Science and Engineering* (Vol. 662, No. 5, p. 052010). IOP Publishing.
- [8] Sheikhalishahi, S., Miotto, R., Dudley, J.T., Lavelli, A., Rinaldi, F. and Osmani, V., 2019. Natural language processing of clinical notes on chronic diseases: systematic review. *JMIR medical informatics*, 7(2), p.e12239.
- [9] Mishra, A., Shekhar, S., Singh, A.K. and Chakraborty, A., 2019, September. Ocr-vqa: Visual question answering by reading text in images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)* (pp. 947-952). IEEE.



IT19111766 | Lekamalage U.L.V.M

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



Introduction



Medication are one of the most useful medical services in the world because meds help to improve wellbeing for the ages



People use medication in their daily routine without knowing what are the reason to consume those and what are the side effect of them.



Sometimes those medication can make path to loss of a human life. Therefore, it is required way to overcome problems course by the medication



Research Problem



Currently there are no drug identification systems to identify drugs using just an image



Drugs cannot be identified without the help of Doctors and Medical practitioners.



Most of the people aren't aware about the reasons to consume the prescribed medication and what are the side effects of them.



Research Gap

Product Name	Can Identify drug using image	Provide Details about drug	Provide Side Effects about drug
WebMD	✗	✓	✓
Drugs.com	✗	✓	✓
RxList	✗	✓	✗
Medscape	✗	✓	✗
Oxygen	✓	✓	✓



Objective

Main Objective

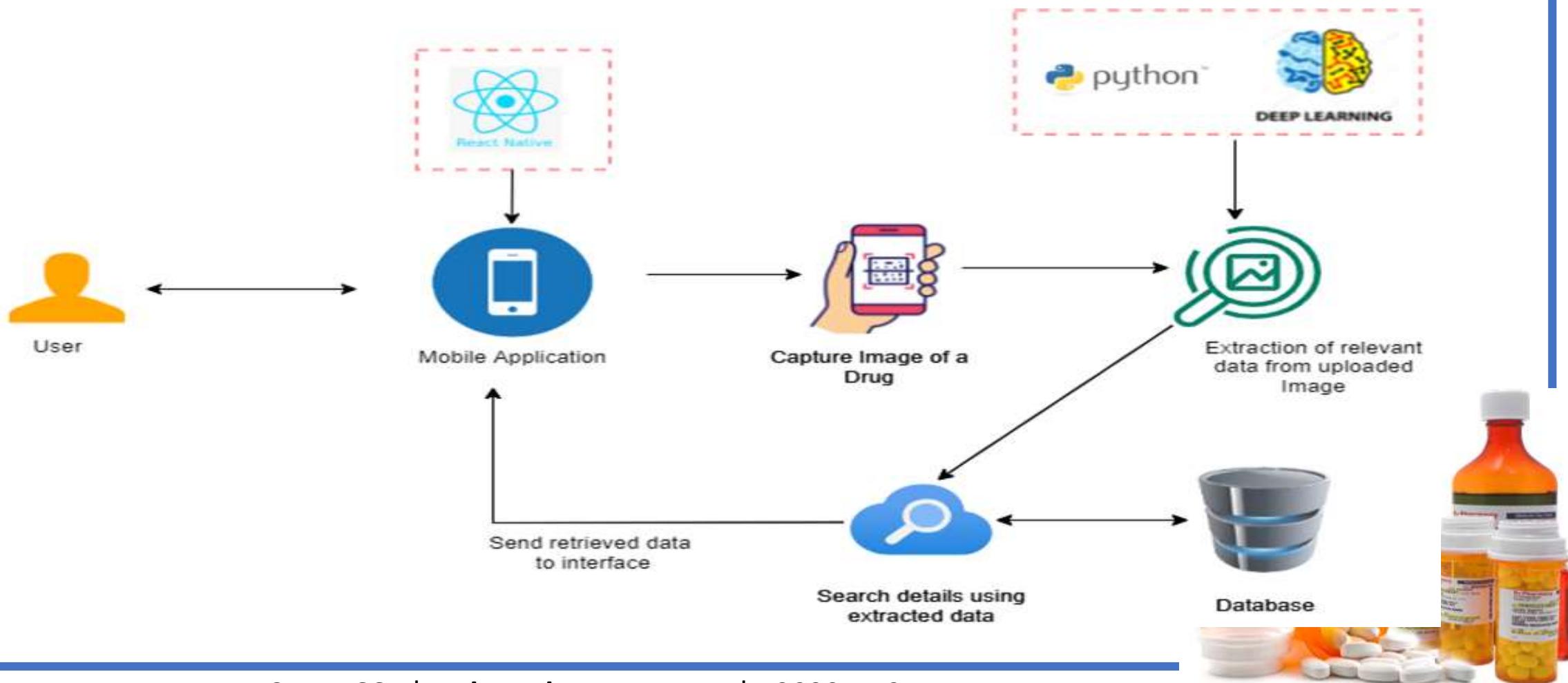
- Develop a system which can identify the medication and provide Detailed summary about the relevant drug.

Sub Objective

- Implement a dependable communication strategy between the application and cloud base server.
- Create A solid decision-making process in the cloud base server.
- Create an application to Carry out all the related functions.



System Diagram



Methodology

1. First upload drug's image to server using mobile application
 - In here we use React Native, NodeJS
2. Enhance the image quality and determined the color, shape and imprint of the pill
 - In here we use colorgram, contours and Amazon Rekognition
3. Find a most suitable result for retrieved data and make a summary and gave to the user.



Completion of Project

Software Installations

- Anaconda(Jupyter), OpenCV, Numpy, Flask, boto3, webcolors

Load Drug's image and extract imprint from Image

```
response=client.detect_text(Image={'S3Object':{'Bucket':'drugidentification','Name':photo}})

textDetections=response['TextDetections']
print ('Detected text\n-----')
text2 = ""
for text in textDetections:
    if text['DetectedText'] not in text2:
        text2 = text2 + text['DetectedText']
text2 = ''.join(text2.split())
print(text2)
```

Detected text

L403

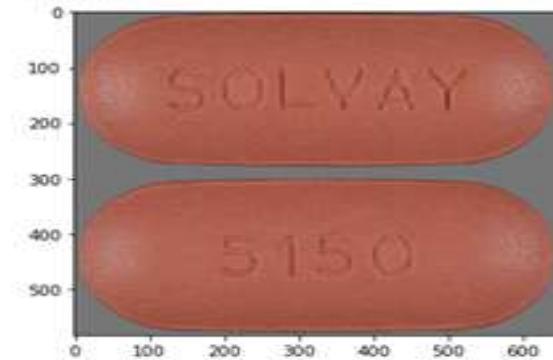


```
response=client.detect_text(Image={'S3Object':{'Bucket':'drugidentification','Name':photo}})

textDetections=response['TextDetections']
print ('Detected text\n-----')
text2 = ""
for text in textDetections:
    if text['DetectedText'] not in text2:
        text2 = text2 + text['DetectedText']
text2 = ''.join(text2.split())
print(text2)
```

Detected text

SOLVAY5150



Completion of Project

Extract color from Image

Detected text

L403



```
|: requested_colour = color(photo)
|: print(requested_colour)
```

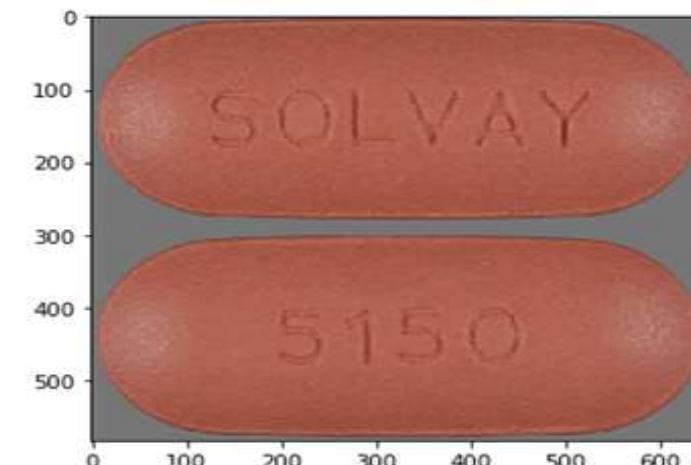
```
Rgb(r=208, g=201, b=203)
```

```
|: closest_name = get_colour_name(requested_colour)
|: print(closest_name)
```

```
lightgray
```

Detected text

SOLVAY5150



```
|: requested_colour = color(photo)
|: print(requested_colour)
```

```
Rgb(r=178, g=102, b=86)
```

```
|: closest_name = get_colour_name(requested_colour)
|: print(closest_name)
```

```
indianred
```



Completion of Project

Extract color from Image

Detected text

L403



```
|: requested_colour = color(photo)
|: print(requested_colour)
```

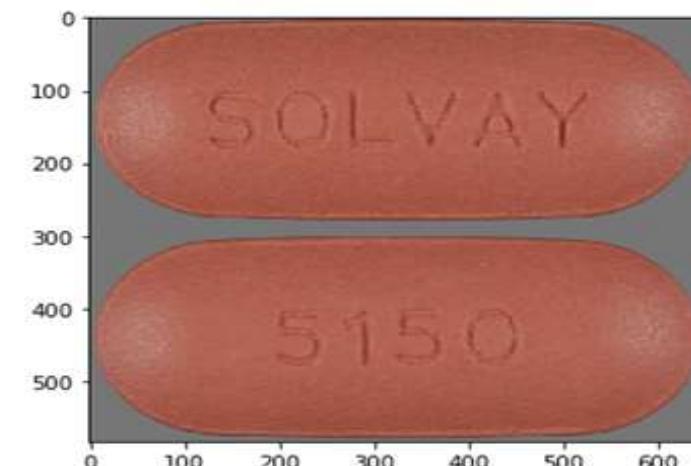
```
Rgb(r=208, g=201, b=203)
```

```
|: closest_name = get_colour_name(requested_colour)
|: print(closest_name)
```

```
lightgray
```

Detected text

SOLVAY5150



```
|: requested_colour = color(photo)
|: print(requested_colour)
```

```
Rgb(r=178, g=102, b=86)
```

```
|: closest_name = get_colour_name(requested_colour)
|: print(closest_name)
```

```
indianred
```



Completion of Project

Extract shape from Image

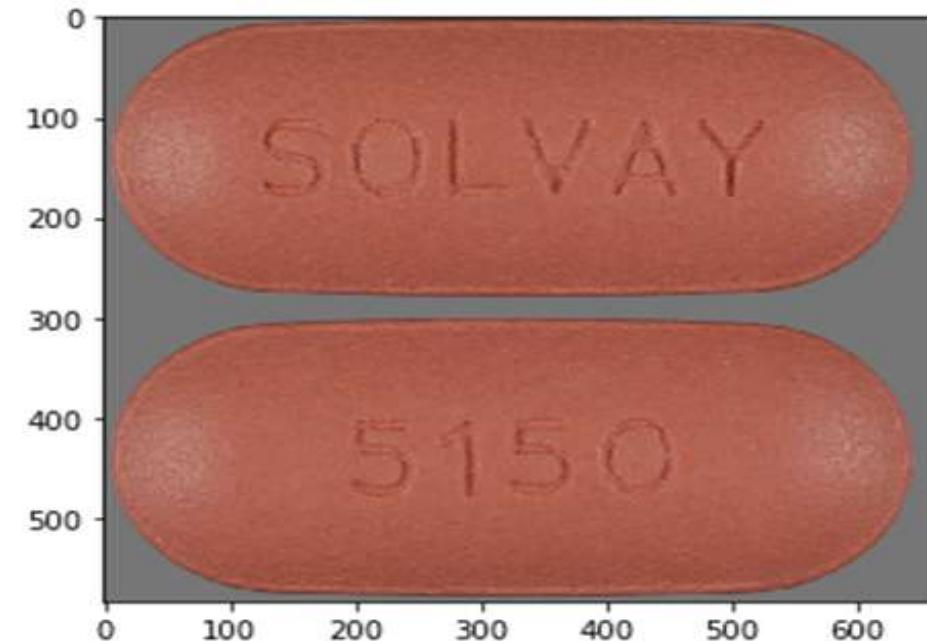
```
imshow(photo)  
print(shape)
```

CIRCLE



```
imshow(photo)  
print(shape)
```

OVAL



Completion of Project

Extract Data Using REST API

POST <http://localhost:8070/drug/identify>

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE
<input checked="" type="checkbox"/> drug_image	acetaminophen.jpg 
Key	Value

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON 

```
1 "Imprint": "L403",
2 "color": "WHITE",
3 "shape": "CIRCLE"
```

POST <http://localhost:8070/drug/identify>

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE
<input checked="" type="checkbox"/> drug_image	15.jpg 
Key	Value

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON 

```
1 "Imprint": "SOLVAY5150",
2 "color": "RED",
3 "shape": "OVAL"
```



Technology & Tool Selection

➤ Technologies

- Image Processing and machine learning

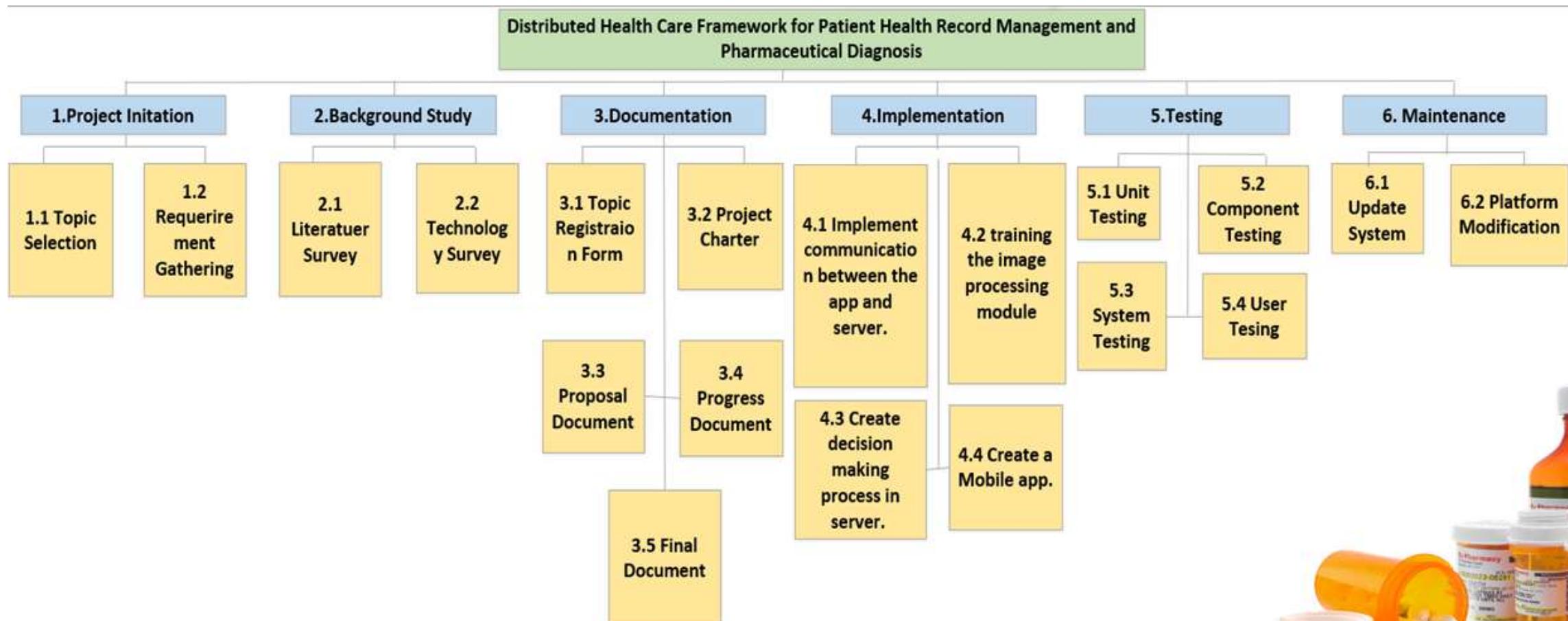


➤ Tools

- For image Processing- OpenCV
- For imprint Recondition - Amazon Rekognition
- For mobile application- React Native and Node JS
- For version controlling- GitLab
- Project Management



Work Breakdown Structure



Gantt Chart

Description	December	January	February	March	April	May	June	July	August	September	October	November	December
Project Initiation													
Evaluation													
Project Charter													
Proposal Draft													
Proposal Presentation													
Project Phase													
Collecting Required Data													
System Planning													
Selecting Technologies													
Research Paper													
Implementation Phase													
Implement communication between the app and server.													
training the image processing module													
Create an application to Carry out all the related functions.													
Create decision making process in server.													
Testing Phase and Evaluation													
Testing													
Final Report and Research paper													
Final Evaluation													



Project Requirements

➤ Functional requirements

- Extract data from uploaded drug's image and identify the drug and provide a summary about drug

➤ Non-Functional requirements

- Less manual work to use the overall system
- Take less time to extract data from drug's image.
- Accurate recognition.

➤ User Requirements

- Cell phone which has camera
- Internet Connection





IT19043388 | Chathuranga S.J.

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



Introduction



The importance of taking medication on time.



Reasons to forget to take medication on time.



Therefore, a way is needed to avoid problems by not taking medication on time

Research Problem



People are uniquely engaged in their current daily routine.



They tend to focus more on their work rather than their personal lives.



As a result, they often forget to get their medication on time and often have no one to remind them of it.

Research Gap



- Many healthcare virtual assistants use channeling to make necessary appointments to patients' relevant physicians and to diagnose their ailments diseases.
- Use to Smart Virtual Assistant for giving the medication time and give responses according to the prescription.





Objectives

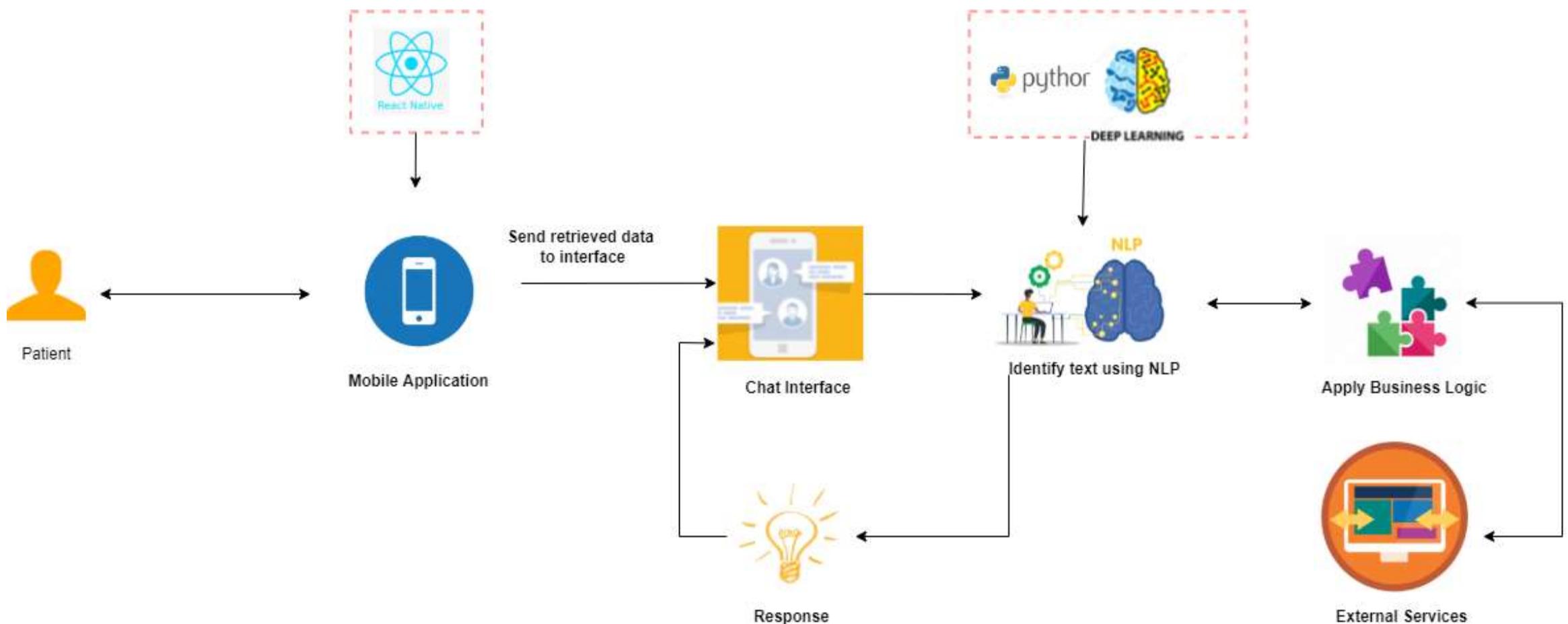
Main Objective

- To assist patients with a smart chatbot based on Machine learning and Natural Language Processing for health care assistance.

Sub Objectives

- Manage Medication time system.
- Manage System Setting
- User friendly interface.

System Diagram



Methodology



Get input from the patient.



Identify input using natural language processing and machine learning



Give an appropriate response to the relevant input

Tools and Technologies

Technologies

- Natural Language and machine learning



Tools

- Chatbot Development (Rasa/Wit.Ai)
- For mobile application- React Native & Node Js
- For version controlling- GIT



Completion of Project

Training the model

```
Processed story blocks: 100% [██████████] | 5/5 [00:00<00:00, 1798.90it/s, # trackers=1]
Processed story blocks: 100% [██████████] | 5/5 [00:00<00:00, 519.73it/s, # trackers=5]
Processed story blocks: 100% [██████████] | 5/5 [00:00<00:00, 70.51it/s, # trackers=30]
Processed story blocks: 100% [██████████] | 5/5 [00:00<00:00, 32.05it/s, # trackers=50]
Processed rules: 100% [██████████] | 2/2 [00:00<00:00, 1998.24it/s, # trackers=1]

2022-05-04 20:20:07 INFO rasa.engine.training.hooks - Starting to train component 'MemoizationPolicy'.
Processed trackers: 100% [██████████] | 5/5 [00:00<00:00, 1245.34it/s, # action=16]
Processed actions: 16it [00:00, 5355.45it/s, # examples=16]
2022-05-04 20:20:08 INFO rasa.engine.training.hooks - Finished training component 'MemoizationPolicy'.
2022-05-04 20:20:08 INFO rasa.engine.training.hooks - Starting to train component 'RulePolicy'.
Processed trackers: 100% [██████████] | 2/2 [00:00<00:00, 1563.58it/s, # action=5]
Processed actions: 5it [00:00, ?it/s, # examples=4]
Processed trackers: 100% [██████████] | 5/5 [00:00<00:00, 1671.44it/s, # action=16]
Processed trackers: 100% [██████████] | 2/2 [00:00<00:00, 2005.40it/s]
Processed trackers: 100% [██████████] | 7/7 [00:00<00:00, 2719.03it/s]

2022-05-04 20:20:08 INFO rasa.engine.training.hooks - Finished training component 'RulePolicy'.
2022-05-04 20:20:08 INFO rasa.engine.training.hooks - Starting to train component 'TEDPolicy'.
Processed trackers: 100% [██████████] | 350/350 [00:00<00:00, 2985.93it/s, # action=126]
Epochs: 100% [██████████] | 100/100 [00:15<00:00, 6.54it/s, t_loss=1.46, loss=1.29, acc=1]
2022-05-04 20:20:25 INFO rasa.engine.training.hooks - Finished training component 'TEDPolicy'.
Processed trackers: 100% [██████████] | 350/350 [00:00<00:00, 4410.35it/s, # intent=42]
Epochs: 100% [██████████] | 100/100 [00:10<00:00, 9.65it/s, t_loss=0.149, loss=0.0508, acc=0.994]

2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Finished training component 'UnexpectTEIntentPolicy'.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'CountVectorsFeaturizer' from cache.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'CountVectorsFeaturizer' from cache.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'DIEIClassifier' from cache.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'EntitySynonymMapper' from cache.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'LexicalSyntacticFeaturizer' from cache.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'RegexFeaturizer' from cache.
2022-05-04 20:20:38 INFO rasa.engine.training.hooks - Restored component 'ResponseSelector' from cache.

Your Rasa model is trained and saved at 'models\20220504-202005-metal-ticket.tar.gz'.
```



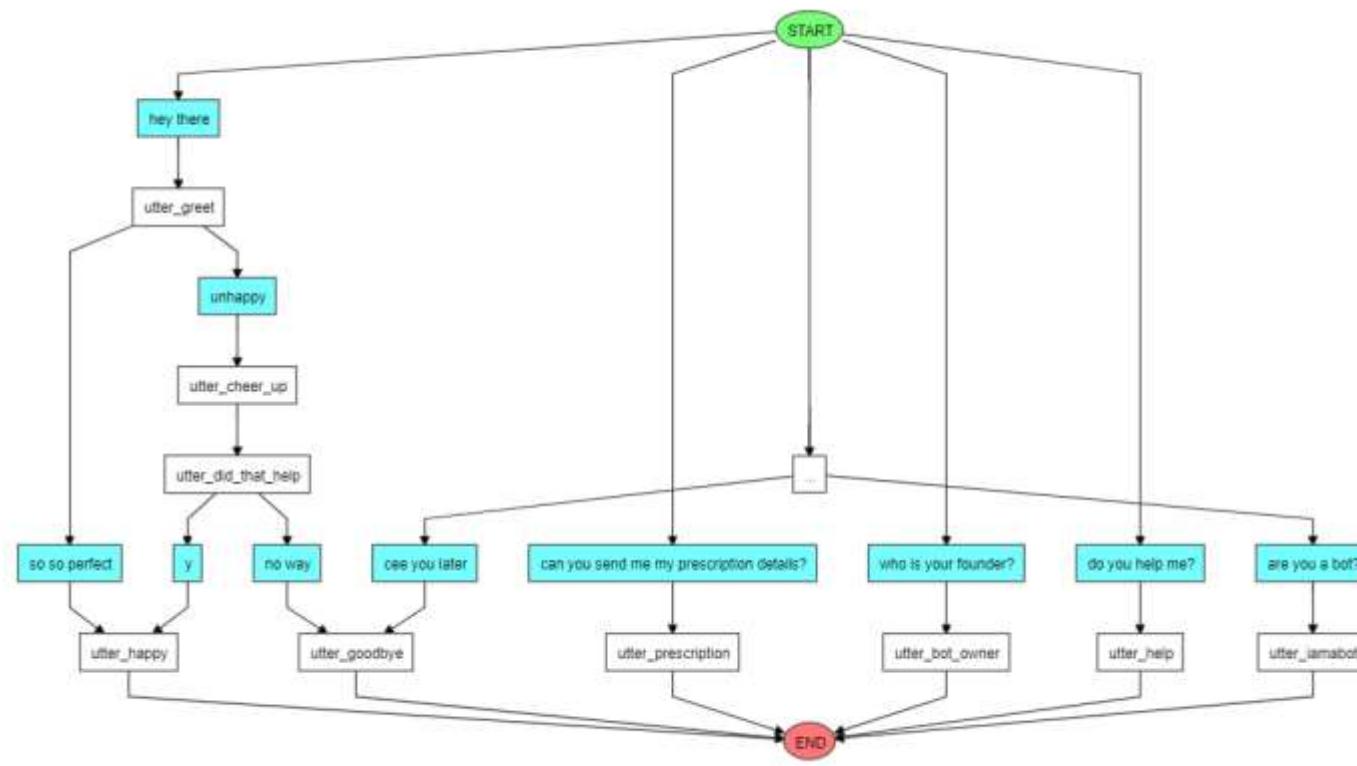
Completion of Project

Sample chat output

```
Your input -> hi
Hey! How are you?
Your input -> im good
Great, carry on!
Your input -> can you help me?
yes tell me i can help you.
Your input -> can you help me to find my prescription details?
Please Wait Im find your history.....
Your input -> ||
```

Completion of Project

Graph of chatbot model





Completion of Project

Chat history with details

```
Chat History

#   Bot                               You.

1   action_listen

2                               hi
                                intent: greet 1.00

3   utter_greet 1.00
    Hey! How are you?
    action_listen 1.00

4                               im good
                                intent: mood_great 0.97

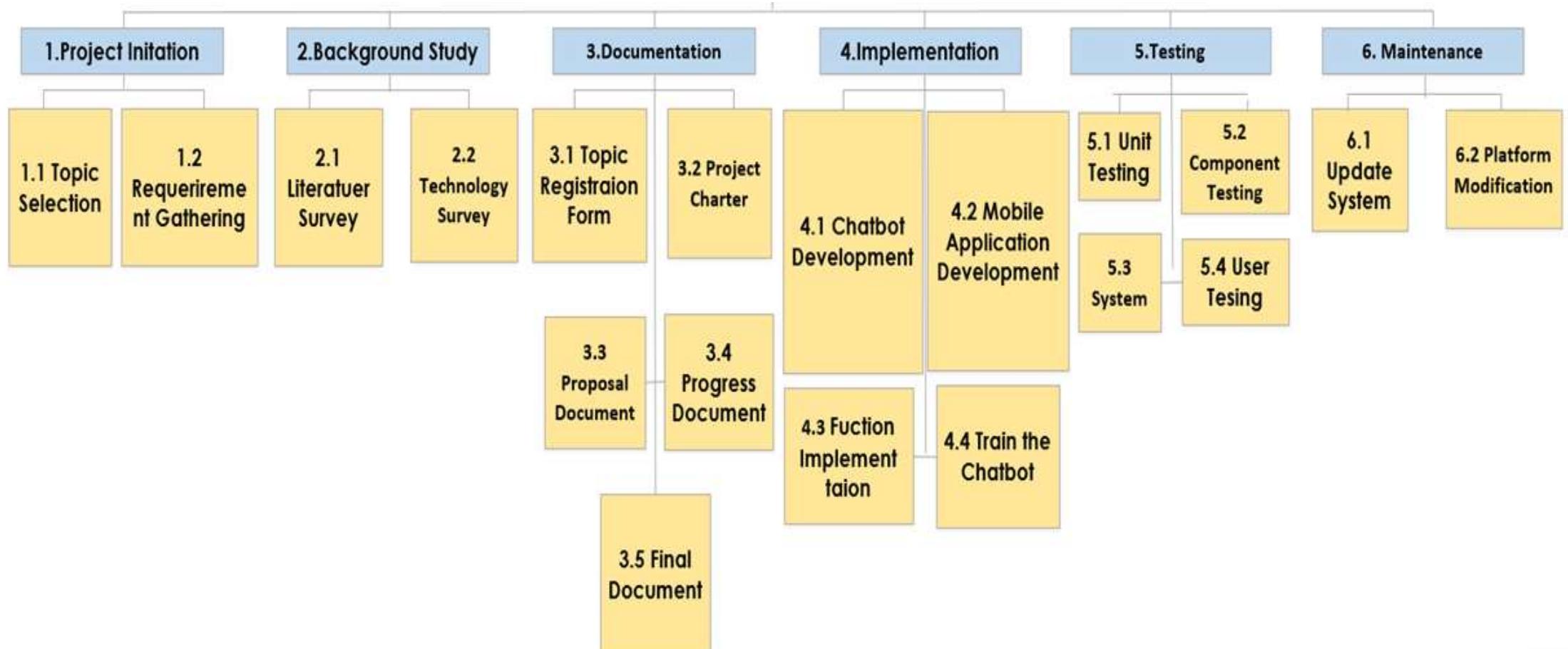
5   utter_happy 1.00
    Great, carry on!
    action_listen 1.00

6                               can you help me?
                                intent: help 1.00

7   utter_help 0.99
    yes tell me i can help you.
    action_listen 1.00

8                               can you help me to find my prescription details?
                                intent: prescription 1.00
```

Work Breakdown Structure



Gantt Chart

Task Name	Timeline												
	November	December	January	February	March	April	May	June	July	August	September	October	November
Project Initiation													
Evaluation													
Topic Assessment form													
Charter													
Proposal Draft													
Proposal Presentation													
Project Phase													
System Planning													
Collecting Required Data													
Selecting Algorithm technologies													
Implementation Phase													
Implement mobile application													
chatbot implementation													
Function Implementation													
Train the chatbot													
Testing Phase and Evaluation													
Testing													
Final Report and Research paper													
Final Evaluation													

Project Requirements

➤ Functional Requirements

- Identify the message and respond appropriately to the patient, identify the patient's prescription.

➤ Non-Functional requirements

- Accuracy
- Speed.
- Reliability.

➤ User Requirements

- Smart Mobile Phone
- Internet Connection

Future Work



- Mobile application development
- API integration with chatbot
- Integrate mobile application with chatbot
- Host the server
- Testing

Business Potential



Supports any Hospital Chain across the world

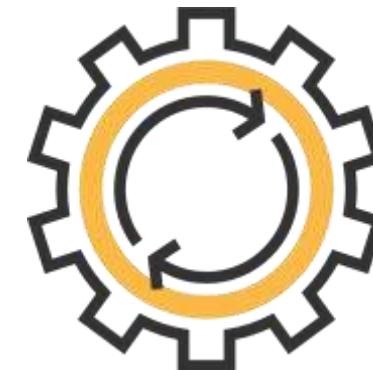


24/7 support with no downtime



DATA PROTECTION

High Data Security



100% Automatic Solution



Free Drug Identification



Medical Document Scanning from anywhere



24/7 Virtual Assistant

Business Plan



Standard

What you'll Get

-  **24/7 support with no downtime**
-  **Free Virtual Assistance**
-  **Unlimited Users (10 Users only)**
-  **30GB Storage**

Choose

Premium

What you'll Get

-  **24/7 support with no downtime**
-  **Free Virtual Assistance**
-  **Unlimited Users**
-  **Unlimited Storage**

Choose



Oxygen is a healthcare framework to address the healthcare difficulties that may occur due to the COVID-19 pandemic. The pandemic exposed healthcare's shortcomings, and this framework will automate the existing healthcare services. Key objective of this product is to securely store patients' healthcare information while protecting users' privacy and to provide healthcare services for Medical Documents Scanning, Conversational Chatbot for Virtual Assisting and remote pharmaceutical diagnosis.

CARE YOUR
HEALTH

100% PATIENT DATA SECURITY

FREE DRUG IDENTIFICATION

FREE MEDICAL DOCUMENT SCANNING

24/7 VIRTUAL ASSISTANT

Sci Lanka Institute of Information Technology,
New Kandy Rd, Malabe
0117 544301



Business Potential

Healthcare Flyer



THANK YOU