SETHU INSTITUTE OF TECHNOLOGY

Department of

COMPUTER SCIENCE AND BUSINESS SYSTEMS

19UCB801 – FINAL YEAR PROJECT

E – BOT

An Advanced Chatbot built using NLP and Keras Neural Networking

---

Mr. V. Rajesh Kumar B.E, M.E,
Assistant Professor
Department of CSBS

Aravind S – 2020113002
Aravinth S – 2020113003
Palaniappan M – 2020113306

Abstract

E – Bot is web based chatbot application that aims to simplify the user access to Tamilnadu E-Service websites by seamlessly guiding them through various processes. E – Bot uses Natural language processing (NLP) and Keras neural network algorithms to comprehend and respond human like answers. The proposed system not only allows users to fetch details and make payments but also adapts to their prompt and recording valuable data for the future training.

Literature Review – 1

Journal Name	Journal of System and Management Science Vol:13 ***[2023]***
Title	Improving Chatbot Performance using Hybrid Deep Learning Approach.
Authors	Palanisami Naveen[1], Sue-Cheng Haw[2], Devakumar Nadathan[3]
Methodology Used	Data gathering and data Pre-Processing is the initial step of the this theoretical model. The next crucial step is building hybrid model which generates real time text using pre defined model and the output is refined using encoder.
Limitations	Restricted Capabilities and Loss of Generic Inputs.

Source - https://www.aasmr.org/jsms/Vol13/No.3/Vol.13.3.34.pdf

Literature Review – 2

Journal Name	Science Direct vol:11, Edition: 100198 ***[2023]***
Title	A comparative study of retrieval – based and generative – based chat bot using Deep Learning and Machine Learning.
Authors	Sumit Pandey[1], Srishti Sharma[2]
Methodology Used	Data gathering and data Pre-Processing using quasi-statistical method to analyse the importance of school based mental health service (SBMHSs). It is a multi-tiered approach which trains the processed data to generate an accurate response.
Limitations	Limited responses, Once started, the training model cannot be modified.

Source - https://www.sciencedirect.com/science/article/pii/S2772442523000655

Literature Review – 3

Journal Name	Journal of Management and Services Science Vol.: 02, Article: 15 ***[2022]***
Title	Artificial Intelligence based Chat bot: A Case Study.
Authors	Nidhi Singh Kushwaha[1], Pawan Singh[2]
Methodology Used	It is a Rule-based chat bot guided with pre defined questions and it's respective answers. It uses NLP engine to communicate with user which has internet classifier and entity extractor.
Limitations	It does not shift from thing it already knows.

Source - https://jmss.a2zjournals.com/index.php/mss/article/view/15/15

Literature Review – 4

Journal Name	IEEE – Springer Vol.: 11-18 Article: ICTCS 15 ***[2022]***
Title	AI-Based Interactive Agent for Health Care Using NLP and Deep Learning.
Authors	Hemavathi U[1], Ann C. V. Medona[2]
Methodology Used	AI based interactive agent using Natural language processing and Deep learning which deals with simple queries and provide health cares services. It uses NLP and neural network to process data.
Limitations	Requires intense training of pre-processed data.

Source - https://link.springer.com/chapter/10.1007/978-981-19-0095-2_2

Literature Review – 5

Journal Name	IEEE – Springer Vol.: 398 2nd Edition ***[2021]***
Title	Music Genre Classification Chat Bot.
Authors	Rishit Jain[1], Ritik Sharma[2], Preeti Nagrath[3] and Rachan Jain[4]
Methodology Used	It is a music information retrieval (MIR) the uses traction of Convolutional neural networking (CNN) to differentiate between audio files by assessing the visual representation of the timbral features.
Limitations	Cannot be trained with pre-processed data. Training for each and individual user is important.

Source - https://link.springer.com/chapter/10.1007/978-981-16-0733-2_27

Problem Statement

- There are 6,868 plus websites provided by our government.
- But, only less than half of the websites are being used.
- One of the main reason for that it is hard to navigate and identify the genuine webpage.

Total websites : 6,868 (approx.)

Used websites : 2,998

Unused websites : 3,870+

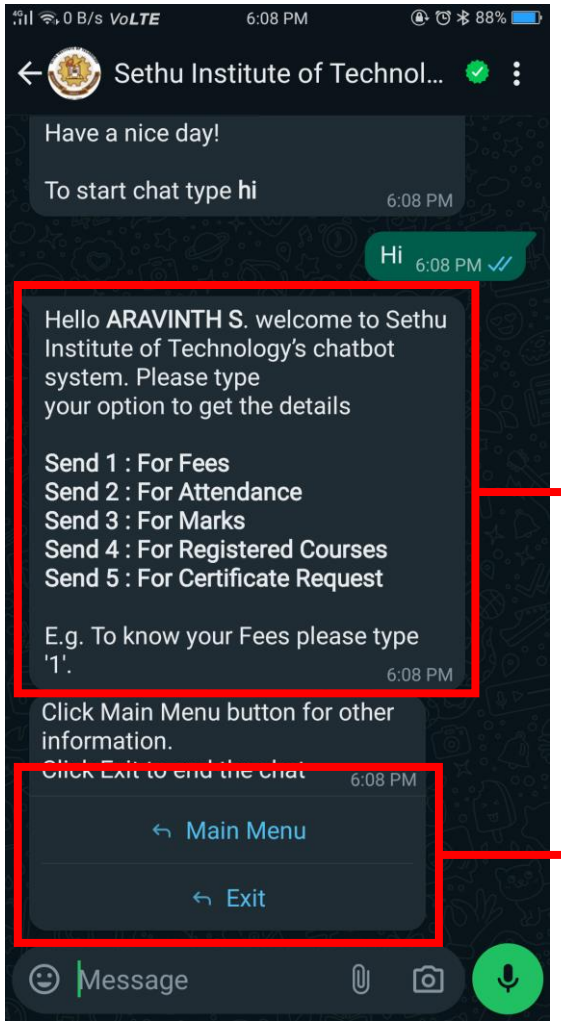
Existing System

- **Menu based Chatbot**

  A Menu based chatbot generates replies using simple heuristics rule-based system to answers queries. It has only limited personalization.

- **Rule-based Chatbot**

  A rule-based chatbot generates replies using decision tree. It uses simple if – else condition to satisfy the users queries.



Rule-Based Chatbot

Menu-Based Chatbot

Drawbacks in Existing System

- There is no prior feature to navigate users through E-Service website.
- Manual navigation using human knowledge.
- Absence of feedback system to know about users experience.
- Unavailability of live training and pre-processing model web based chat bot.

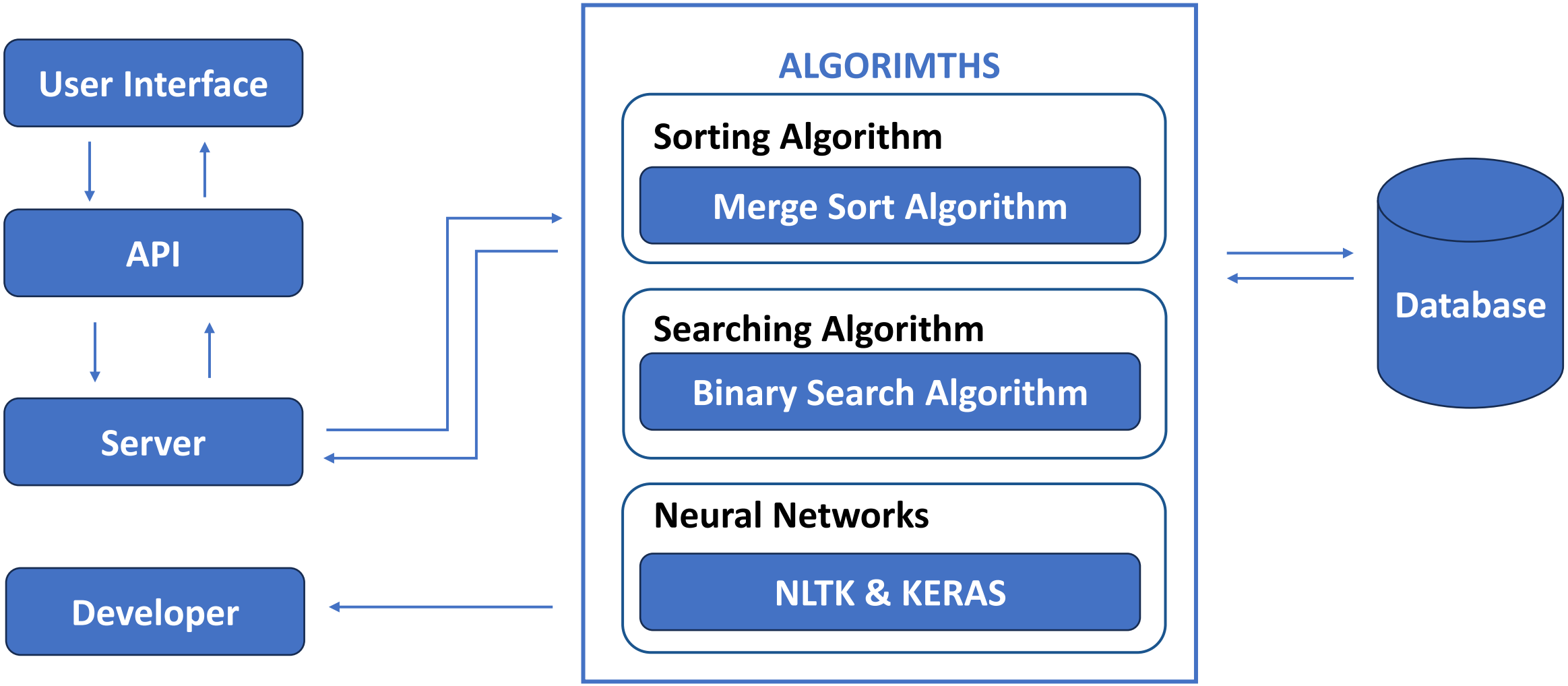
Proposed System

- Proposed solution is a conventional chatbot.
- This helps users to answer their queries and navigate through the E-Service government website.
- Users will be directed straight to the webpage they seek.
- This chatbot has features like,
  - ❑ Common Queries
  - ❑ Bill Payment
  - ❑ Document Download
  - ❑ Toll – Free Numbers

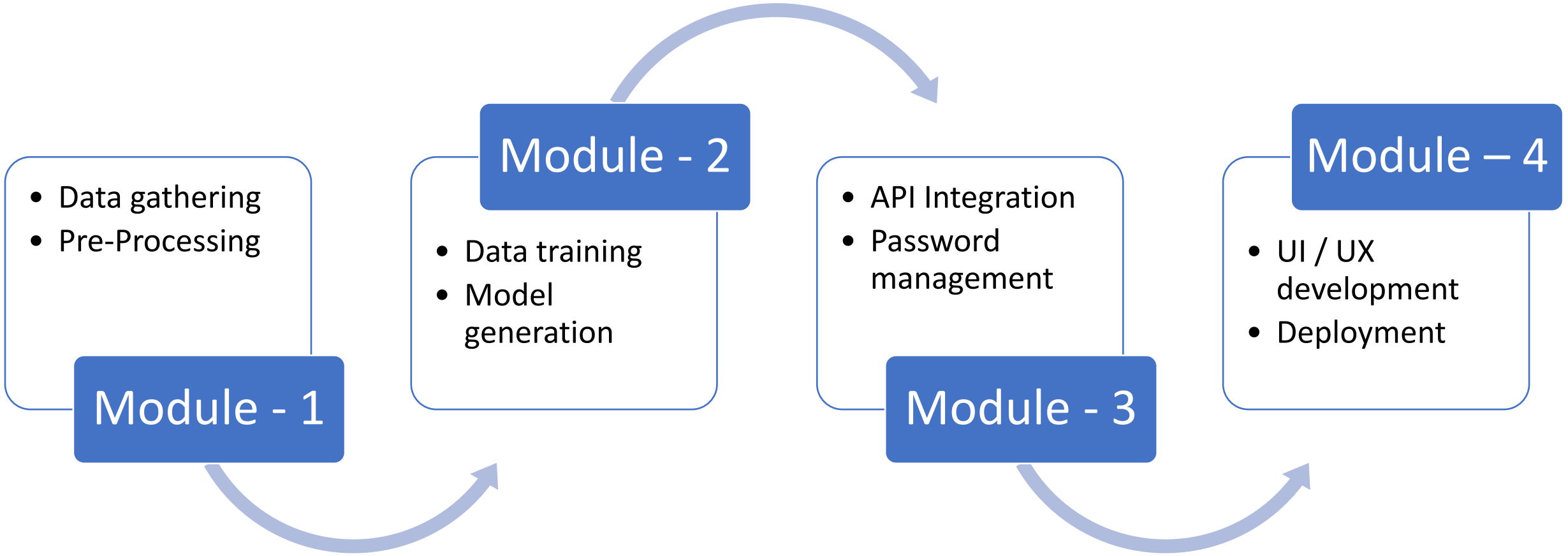
Advantages of Proposed System

- Secured chat history storage that does not involve personal information.
- Allows user to pay bills using local storage and secure payment gateway.
- Uses real time data of user to produce adaptive responsive answers according to the users.
- Stored chat history of user can be used in future to train the model.

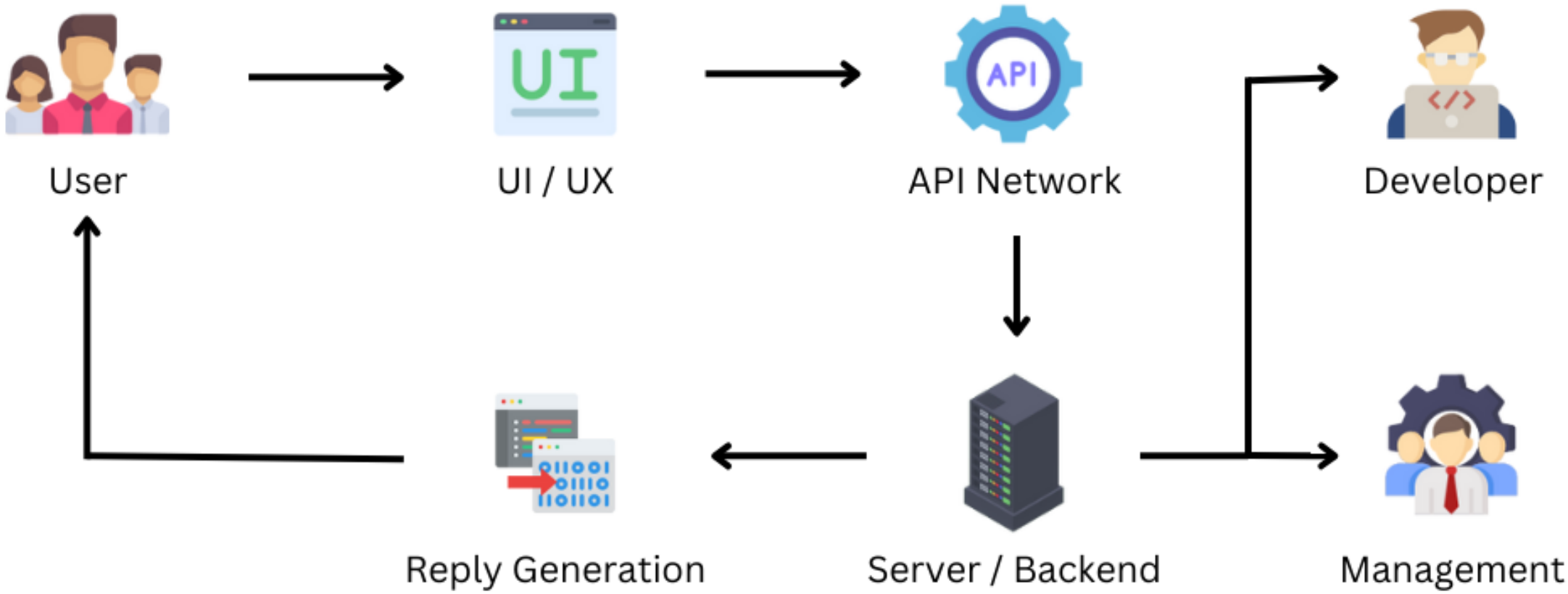
Block Diagram



Module Design



Data Flow Diagram

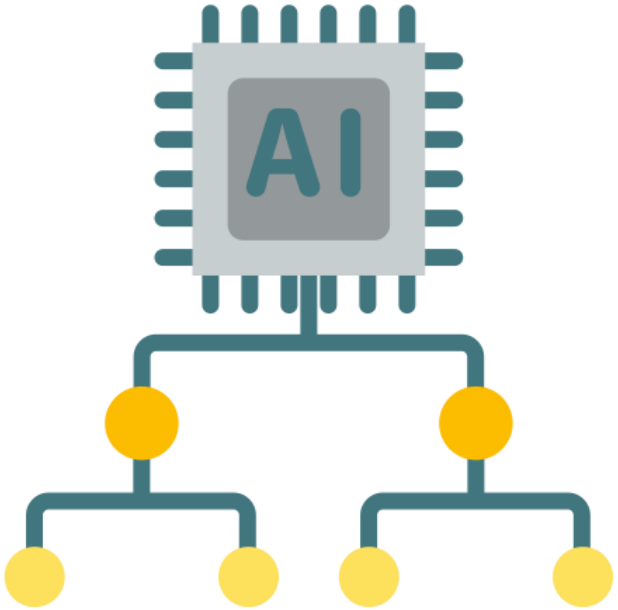


Module 1 – Data Gathering

- Collecting website details.
- Categorizing the websites.
- Users need and Requirements.
- Website usage statistics according to time and region.

Module 2 – Training Data

- Pre-processing the gathered data.
- Organizing the pre-processed data.
- Using Algorithms to train the data.
- Creation of training model using processed data.
- Processing data using NLP and Keras algorithms.



Module 3 – API Integration

- Integration of third party API’s

  - ❑ **IP- Geo Location** : To track users location (Only if required).
  - ❑ **Password Safe** : To store users personal data in their local storage.
  - ❑ **Razor pay Gateway** : Payment gateway to pay bills.

Module 4 – UI / UX Development

- Development of user interface.
- Integration of front-end and back-end.
- Application testing with various scenarios.
- Deployment of the application

Algorithms and Methodology

- **Natural Language Processing :** Can understand and reply human like answers.
- **Tensor Flow :** Used along with NLP to train the pre-processed dataset and create a base model to work with.
- **KERAS :** Keras is neural networking algorithm which defines relationship between multiple question.
- **NumPy :** Mathematical algorithm used to ID the datasets.

Data Sets

- **JSON** file is used to store the data set.
- **Total Dataset used** – 03
- **Current Data set count –** 217 possibilities

Data Type	Data set
Pre – Defined	Universal Dataset
	Intent Recognition
User - Defined	Custom Dataset

```
{"intents": [
    {"tag": "greeting",
     "patterns": ["Hi there", "How are you", "Hello"],
     "responses": ["Good to see you!", "Hi there, how can I help?"]
    }
]}
```

Source - https://github.com/Aravinth-S-731/E-Bot/blob/main/data.json

Implementation

FRONT-END DEV

HTML

CSS

JavaScript

BACK-END DEV

Python

DATASET

Kaggle

FRAMEWORK

Flask

ALGORITHMS

NumPY

NLTK

TensorFlow

API's

IP Geolocation

Password Safe

Razor Pay API

Sample Code

Training Dataset

```python
import nltk, pickle, json, random
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
import tensorflow as tf
import numpy as np
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout
from keras.optimizers import SGD

words, classes, documents, ignore_words = [], [], [], []
data_file, intents = open('data.json').read(), json.loads('data_file')

# Cleaning Dataset
def removing_intents():

# Lemmatizing Words, converting to lower case and removing duplicates
def word_lemmatization():

# Saves he keywowrds as dataset in '.pkl' format
def pickling_to_BinaryFile():

# Vocabulary sorting of words, based on their importance in the document
def training_data_BOW():

# Neural Networking Datasets and storing it in 'HDF5' format
def tensorFlow_KNN():
```

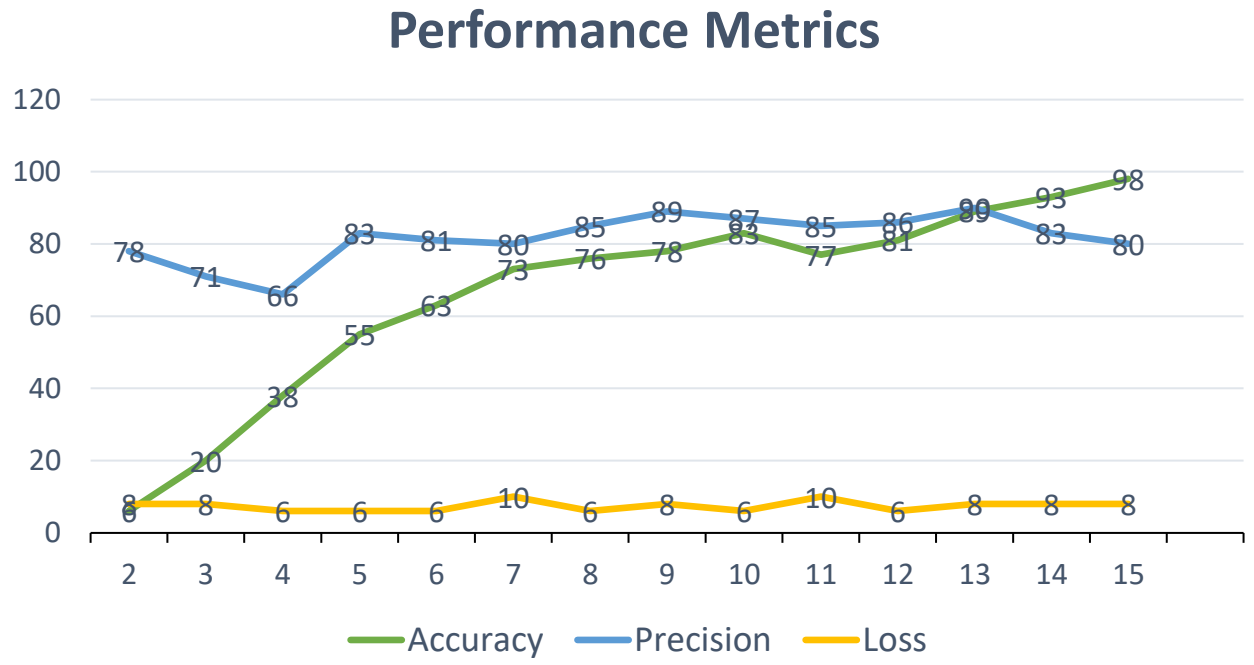
Server Application

```python
import nltk, pickle, json, random
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
import numpy as np
from keras.models import load_model
from flask import Flask, render_template, request

model = load_model('model.h5')
intents = json.loads(open('data.json').read())
words = pickle.load(open('texts.pkl','rb'))
classes = pickle.load(open('labels.pkl','rb'))

# Clean up the user 's input and return a normalized vector of words in Array
def clean_up_sentence(sentence):

# Bagging of words to be vocabularized
def bow(sentence, words, show_details=True):

# From the 'HDF5' file predict the class (intent) of the sentence given by the user
def predict_class(sentence, model):

# Using the 'HDF5' file match with dataset to retrieve intent if found
def getResponse(ints, intents_json):

# Send and Receive Response from Browser
def chatbot_response(msg):

app = Flask(__name__)
@app.route("/index")
@app.route("/about")
@app.route("/user-working")
@app.route("/bot-training")
@app.route("/get")
if __name__ == "__main__":
    app.run()
```

Performance Metrics

- **Accuracy:** 0.9709 = ~ 97%
- **Loss:** 0.0890 = ~ 8%

```
21/21 [==============================] - 1s 2ms/step - loss: 3.5791 - accuracy: 0.0194
Epoch 2/200
21/21 [==============================] - 0s 2ms/step - loss: 3.5215 - accuracy: 0.0000e+00
Epoch 3/200
21/21 [==============================] - 0s 2ms/step - loss: 3.4543 - accuracy: 0.0485
Epoch 4/200
21/21 [==============================] - 0s 2ms/step - loss: 3.4531 - accuracy: 0.1068
Epoch 5/200
21/21 [==============================] - 0s 2ms/step - loss: 3.4274 - accuracy: 0.1262
Epoch 6/200
21/21 [==============================] - 0s 2ms/step - loss: 3.3871 - accuracy: 0.0680
Epoch 7/200
21/21 [==============================] - 0s 2ms/step - loss: 3.2780 - accuracy: 0.1068
Epoch 8/200
21/21 [==============================] - 0s 2ms/step - loss: 3.1946 - accuracy: 0.1359
Epoch 9/200
21/21 [==============================] - 0s 2ms/step - loss: 3.2690 - accuracy: 0.1165
Epoch 10/200
21/21 [==============================] - 0s 2ms/step - loss: 3.1386 - accuracy: 0.1942
Epoch 11/200
21/21 [==============================] - 0s 2ms/step - loss: 3.0066 - accuracy: 0.2136
Epoch 12/200
21/21 [==============================] - 0s 2ms/step - loss: 3.0022 - accuracy: 0.2039
.
.
.
Epoch 199/200
21/21 [==============================] - 0s 2ms/step - loss: 0.0963 - accuracy: 0.9806
Epoch 200/200
21/21 [==============================] - 0s 2ms/step - loss: 0.0820 - accuracy: 0.9903
```

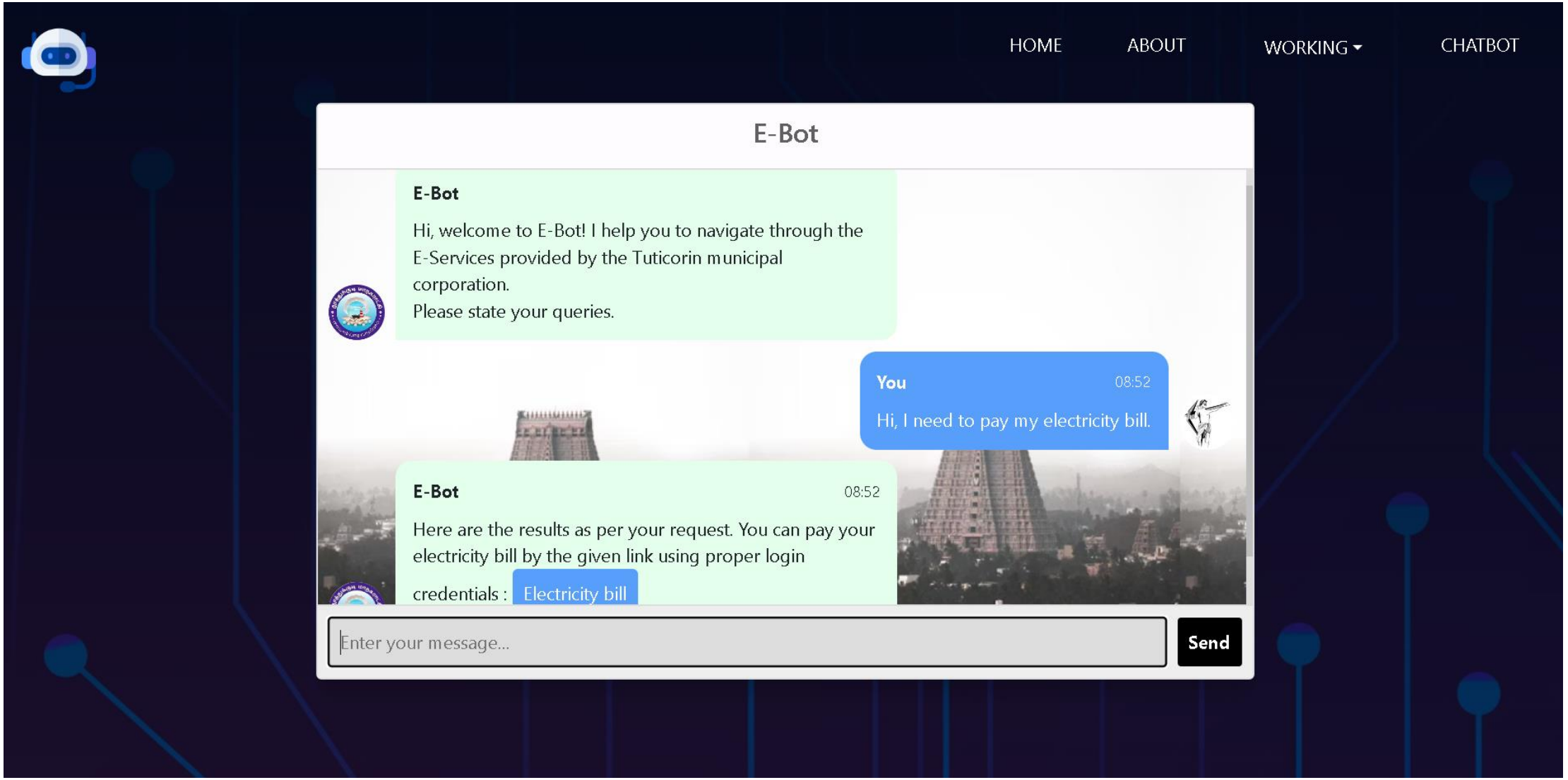


- **Precision:** 85%
- **Interference Time:** 0.02 Sec
- **Training Time:** 144 mins (~2.15 hrs.)
- **Memory Used:** 6 GB (± 400 MB)

Comparison with Existing Solution

Characteristics	Existing Solution	Proposed Solution
Logic	Decision Tree	Natural Language (NLP)
User Interaction	Menu / Buttons	Queries / User Input
Interaction Flow	Corresponding to Inquiry	Context-Aware Response
Personalization	NA	As Required
Use Case	FAQ’s / Pre-defined	Dynamic Conversation
Advantages	Lower Development Cost	Scalable

Result / Output



Scope for Improvement

- Expanding ***Dataset*** modules.
- ***Training data*** with real-time queries.
- ***Payment gateway*** to avoid Third party applications.
- Enable application to ***function offline***.
- Using ***Big-data tools*** to reduce processing time.

References

- *Improving Chatbot Performance using Hybrid Deep Learning Approach.* **Palanisami Naveen[1], Sue-Cheng Haw[2], Devakumar Nadathan[3] -** https://www.aasmr.org/jsms/Vol13/No.3/Vol.13.3.34.pdf

- *A comparative study of retrieval – based and generative – based chat bot using Deep Learning.* **Sumit Pandey[1], Srishti Sharma[2] -** https://www.sciencedirect.com/science/article/pii/S2772442523000655

- *Artificial Intelligence based Chat bot: A Case Study.* **Nidhi Singh Kushwaha[1], Pawan Singh[2] -** https://jmss.a2zjournals.com/index.php/mss/article/view/15/15

- *AI-Based Interactive Agent for Health Care Using NLP and Deep Learning.* **Hemavathi U[1], Ann C. V. Medona[2] -** https://link.springer.com/chapter/10.1007/978-981-19-0095-2_2

THANK YOU

Mr. V. Rajesh Kumar B.E, M.E,
Assistant Professor
Department of CSBS

Aravind S – 2020113002
Aravinth S – 2020113003
Palaniappan M – 2020113306