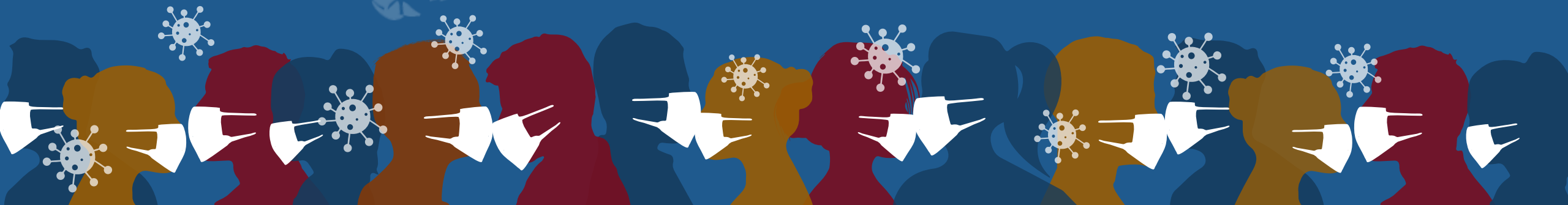




VITCare: make a vas deferens in lives

A DATA STRUCTURE AND ALGORITHM J-
COMPONENT





What is it?

***An All Purpose Medical
Facility Management System***

A J-COMPONENT FOR DATA STRUCTURES AND
ALGORITHM



Content

04

Team Details

Information on people behind the project.

06

Introduction

Inspiration behind making of project

07

Modules

Specifications of different parts of the J-component.

04

Progress

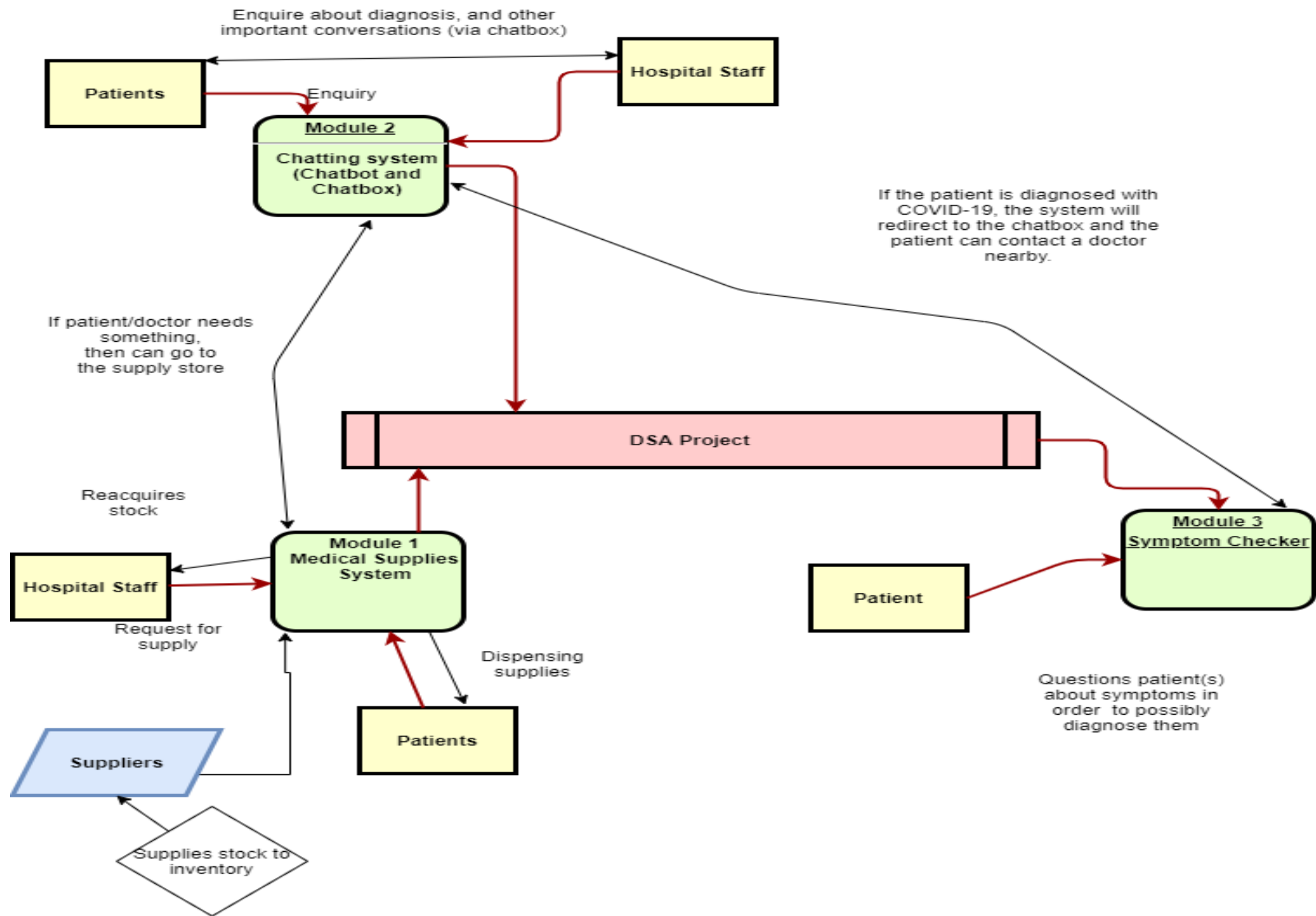
Detailed Information of development of idea.

Our Team Infographic

COVID-19

1. ARYAMAN MISHRA -> Chatbot Designing.
2. ARJUN NAIR -> C program for medical supplies management system
3. CLIVIN JOHN GEJU -> ML for Symptom Checker
4. HARSANTH KP -> Creating a basic webpage design for user interaction and the C program





Abstract

COVID-19

“A man ate a bat and now I cannot go to my favourite restaurant,A man ate a bat and I can’t eat a pizza by home delivery.A man ate a bat and I have to stare at a screen for 12 hours a day and my sleep schedule has been utterly destroyed.”

-Anonymous Student

In India, the Covid-19 deaths per million population stand at 20.4 per million(subject to change) which is amongst the lowest in the world.Our Prime Minister enforced a lockdown which at a certain degree was the correct move.The patients being brought in to various medical facilities took a toll over the administration and the entire system in various systems just broke down.Lives were lost due to sheer recklessness on doctor’s part and some were made to suffer due to lack of resources.

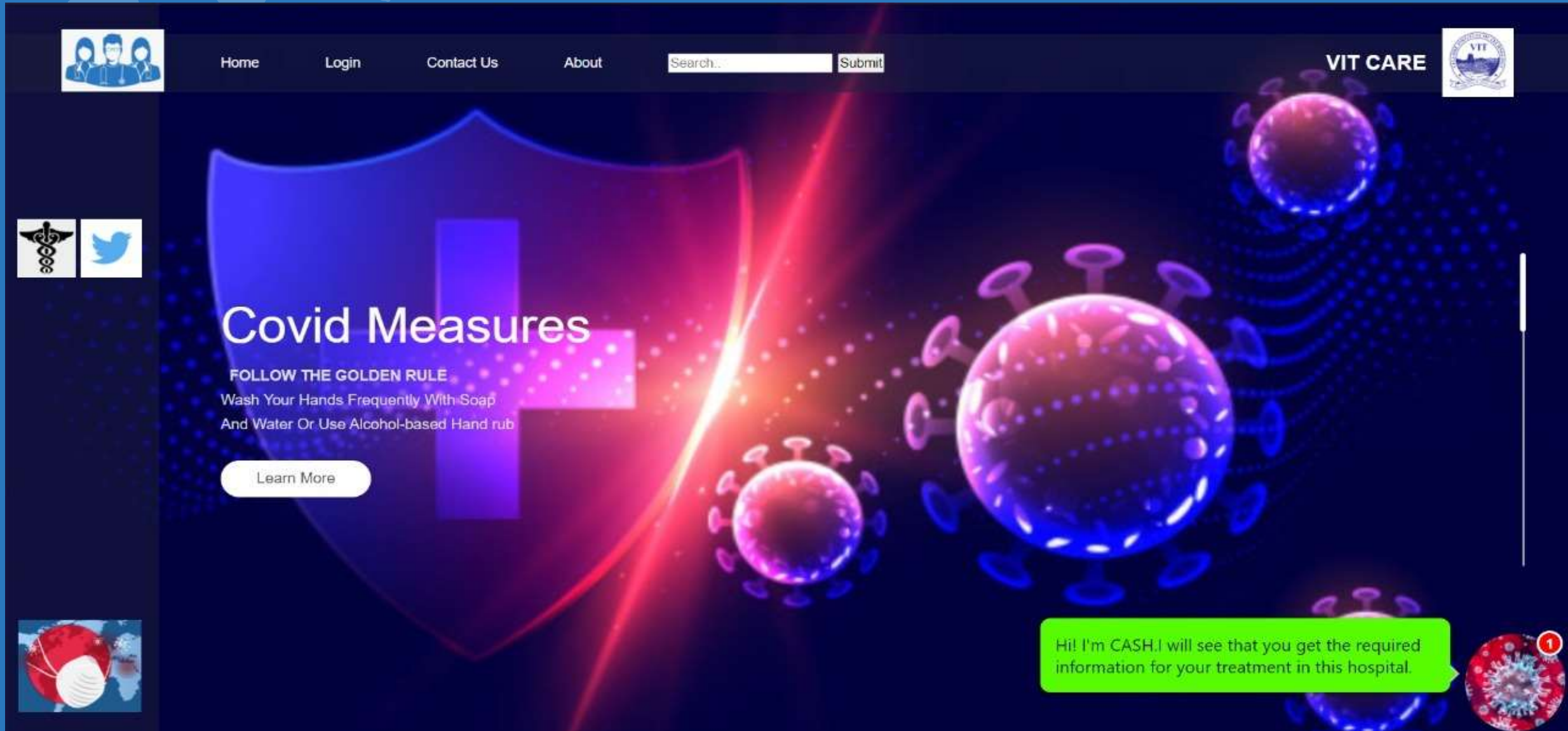
The reason for low supplies in hospital is that the system is too complex and also requires a person to go blind on paperwork related to supplies.Private hospitals which admits less patients have supplies in abundance which are not being used due to high admission rates and civil hospitals are being overcrowded with people who are not getting oxygen cylinders,attention or medicines to stabilize their pitiful conditions.

Our project which has functionalities to last beyond this ‘pandemic era’ allows for simple ordering of medical supplies and equipments using stacks and arrays for a single medical facility.In addition,we integrate an interactive UI which helps in easy establishment of communication between program and user.Our users include both medical staff and patients.In addition,we provide a medical staffer to update the list of items and provide payment options for purchasing with a calculated amount by way of material and quantity.

We have included a chatbot which caters to both users and doctors by giving appointments,assigning ambulances and providing details on driver,state and estimated time of arrival including other necessary hands-on information.

There’s a symptom checker built for particularly the novel COVID-19 which takes necessary details on patients and gives a probability of if a person has been infected with the virus and upto which level is he in danger of having contracted the virus.

Website to link our modules COVID-19



Chatbot

A Data Structures and Algorithm Module



Analysis

Trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute as shown in the above figure. This process is then repeated for the subtree rooted at the new node.



Chatbot

The Chatbot(NAME:CASH) uses responses stored in an array to respond to various queries. It also gives pre-defined options for smoother user interaction.

Introduction



This is the hospital bot. It will provide contacts, find doctors reception schedule, make or cancel an appointment with a doctor. It can also call an ambulance if needed,

Reception Schedule



The reception schedule goes beyond the pandemic and offers to make appointments with surgeons, pediatricians and even gynaecologists.

Contacts



You can get the work hours of a facility and get an option to call a reception or administrating officer..

Others



You can call an ambulance, get a quote to keep your spirits up, call up the symptom checker. More updates will be rolled out over the progress eventually.



Time Complexities

Assuming that each node of the tree requires $O(1)$ to execute, the complexity of the tree is bounded by its depth, e.g, longest route from the root to a leaf node.

Searching->Chatbot searches for function and traverses through all of them. Therefore, searching in binary tree has worst case complexity of $O(n)$.

If chatbot could be self-balanced; difference between height of left sub-tree and right sub-tree of any node can't be more than 1, complexity for executing a single function would be $O(h)$ where h is the height of the tree.



Implementation

Hospital would be directed into creating an account on Snatchbot. They will then import the .snatchbot extension file available in the folder. They can then enter the doctor's emails so that patient can reach them directly by chatbot through online consultation as shown in the video.



Features

Quotes->Keep yourself motivated in these troubled times.Pre-defines by using 2d matrix of characters in Snatchbot interface.

Call Ambulance->Use google maps to give present location which would be forwarded to assigned driver with the most efficient route.

Make appointment->Consult with a doctor online while you wait for emergency services to reach you and apply first-aid.

Call->Get ambulance, reception and administration number via chatbot with option to copy it directly to dialpad.



Plugins Available

- There are 5 types of plugins available:
- **eBay** - for a quick search of goods on eBay.
- **Giphy** - a service for searching gifs or stickers.
- **Weather** - a plugin that allows you to get a weather forecast for any city you want.
- **Calendar** - access to your Google calendar. Authorization required.
- **Jira** - using Jira via bot. Authorization required. Your domain must start with "<https://>".
- **Trello** - with this plugin, your users will be able to create, view, edit or remove the Boards that they own or have access to, Lists, Activities, Cards and Notifications.
- Google Maps API implemented manually.



Introducing Ambulance route feature

Google would never disclose their data structures for their various products but it takes a DSA Student to know the workings behind map applications.

Graphs are awesome **data structures too**. They are **used** to represent elements that share connections. The elements in the graph are called Nodes and the connections between them are called Edges.

Graphs can be **directed**, when their edges have a specific orientation, similar to one-way streets, or **undirected**, when their edges don't have a specific orientation, similar to two-way streets.

Edges can have a value associated with them, called **weight**.

If a graph has many edges, it's called a **dense** graph. Otherwise, if it has few edges, it's called a **sparse** graph.

A series of connections can form a **cycle** if they create a path that lets you to return to the same node.

R-trees are tree data structures used for spatial access methods, i.e., for indexing multi-dimensional information such as geographical coordinates, rectangles or polygons.



Traversal through a Graph

2 ways possible->Breadth First and Depth-First Search

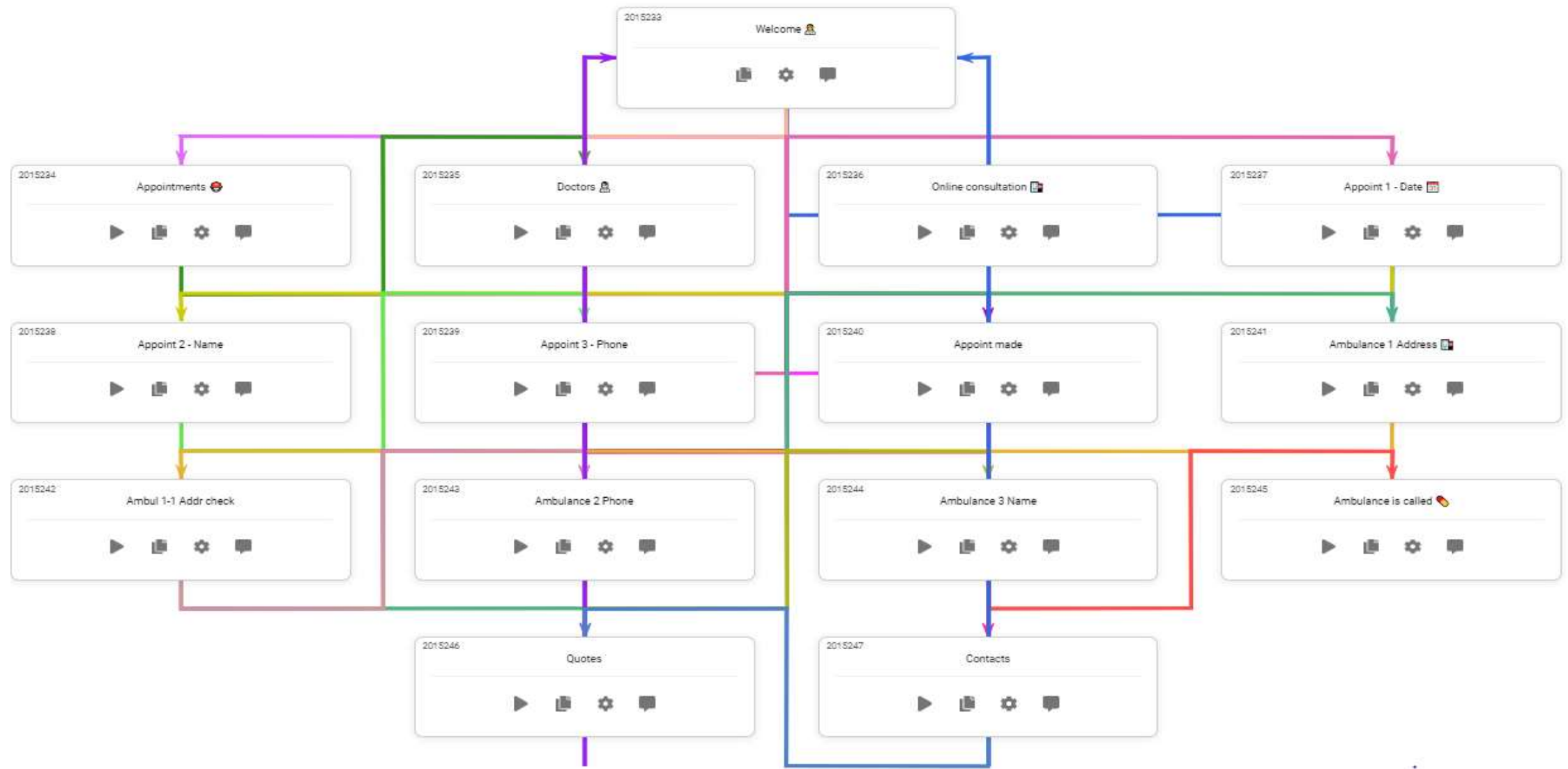
The Time complexity of both BFS and DFS will be $O(V + E)$, where V is the number of vertices, and E is the number of Edges. This **depends on the data structure** that we use to represent the graph.


The Breadth First Search (BFS) traversal is an algorithm, which is used to visit all of the nodes of a given graph. In this traversal algorithm one node is selected and then all of the adjacent nodes are visited one by one. After completing all of the adjacent vertices, it moves further to check another vertices and checks its adjacent vertices again.


The Depth First Search (DFS) is a graph traversal algorithm. In this algorithm one starting vertex is given, and when an adjacent vertex is found, it moves to that adjacent vertex first and try to traverse in the same manner.




Bot Scheme




 CASH:Clivin Aryaman Sunil Harsanth



Hi! Thanks for getting in touch with us on our website. Please send us any question you may have and we will try to get back to you as quickly as we can.



Healthcare Bot

Welcome to Hospital bot! Here you can see our contacts, find doctors reception schedule, make or cancel an appointment with a doctor, or call an ambulance if needed. Choose an option below. 

Oct 12, 2020, 3:55:07 PM



Reception schedule

Contacts



Call an ambulance!


Quote of the day;)


Manage appointments

type something...


 Powered by SnatchBot



**Harsanth PK-
Ophthalmologist**
Monday - Friday, 8:00 - 20:00
Saturday, 8:00 - 20:00
Sunday, 10:00 - 18:00

Online consultation

Make an appointment



**Aryaman Mishra-ENT
specialist**
Monday - Friday, 8:00 - 20:00

Online consultation

Make an appointment



CASH:Clivin Aryaman Sunil Harsanth



Quote of the day;)



Oct 12, 2020, 3:58:41 PM



Healthcare Bot

After dinner rest a while; after supper walk a mile.
T. Cogan

Oct 12, 2020, 3:58:41 PM

More quotes



Oct 12, 2020, 3:58:45 PM



Healthcare Bot

If you need medical advice, let these three things be your physicians;
a cheerful mind, relaxation from business, and a moderate diet.
Schola Salern



type something...



CASH:Clivin Aryaman Sunil Harsanth



Healthcare Bot

You can call for an ambulance by pressing "Call us" button. If all operators are busy or you can not use your phone now, enter your address instead.

If you don't know for sure where you are or our system failed to distinguish a correct address in your answer, you may press No and proceed without entering an address. You will have to say it by phone or in online-conversation with our human operator.

Please, remember that the false call of an ambulance causes not only problems with medical care obtaining for those who need it but also a prosecution and punishment by law!

Call us

+0000000000





Medical Management System

Program Code

COVID-19

Please find code for Medical Management System where we have used Singly Linked List for updating product supplies list:



In case of corrupted file, please refer to the code given below: (Blurred lines is code in font size 1)

```
#include <iostream>
using namespace std;
// Node structure
struct Node {
    int data;
    Node* next;
};
// Singly Linked List
Node* head = NULL;
Node* tail = NULL;
// Insertion at the end
void insertAtEnd(int data) {
    Node* newNode = new Node;
    newNode->data = data;
    newNode->next = NULL;
    if (head == NULL) {
        head = tail = newNode;
    } else {
        tail->next = newNode;
        tail = newNode;
    }
}
// Deletion from the list
void deleteFromList(int data) {
    Node* temp = head;
    while (temp != NULL) {
        if (temp->data == data) {
            if (temp == head) {
                head = temp->next;
            } else if (temp == tail) {
                tail = temp->prev;
            } else {
                temp->prev->next = temp->next;
            }
            delete temp;
            return;
        }
        temp = temp->next;
    }
}
// Display the list
void displayList() {
    Node* temp = head;
    while (temp != NULL) {
        cout << temp->data << " ";
        temp = temp->next;
    }
    cout << endl;
}
// Main function
int main() {
    insertAtEnd(10);
    insertAtEnd(20);
    insertAtEnd(30);
    displayList();
    deleteFromList(20);
    displayList();
    return 0;
}
```

Medical Supplies Management System

COVID-19

We will be making a C program to support the medical supplies management system which will be implemented using stack and SLL data structure/ADT . We will provide a program in which medical supplies with their stock and cost will be displayed for the user from which any of the items of any quantity(until stock runs out) can be bought and the payment can be done through cash or credit card . The stock and prices can be updated by the admin by accessing the admin panel.

This system is very helpful for patients to buy medicines and supplies such as masks and medical personnel such as doctors to refer the stock to prescribe required medicines to the patients and to order inventory for the hospital from the list.

This program can also be integrated with the chat bot so that the doctor or medical personnel can redirect a patient to buy the prescribed medicine or to look over the stock of medical supplies present in the hospital.

The SLL ADT is used to store the medicines and medical items in the list from which the user can select the products/

Data structure

- We have used singly linked list for building this module.
- The main was to reduce the amount of pointers that to be maintained as compared to DLL, eg: in case of insert we need to maintain 2 pointers next and previous so to reduce the extra space we used SLL.

Processes involved

In singly linked list;

- Insertion (where new medical related stuffs are added by using next pointer and storing it)
- Deletion (here we delete the product)
- Updation (when a customer buys a product the content already in the list needs to be updated)
- Searching (here the customer is provided with various options for searching there product).

Time complexity

- Time complexity of the complete module is dependent on the user, as the user can decide when to terminate the module.
- In case of insertion involved , we have 'n' as the size or total number of products, so $\theta(1)$ will be for insertion at average case and for worst case it would be $O(1)$.
- In case of search , $\theta(n)$ in average case and in worst case it would be $O(n)$.
- In case of deletion ,average time complexity is of $\theta(1)$ and in worst case it is $O(1)$.

Screenshots

COVID-19

=> 0. Main Menu

supply No.	supply Name	Price	In Stock
5	Masks	120.23	23
6	Syringes	100.67	13
1	Cough Syrups	720.83	8
2	First Aid Kits	70.23	46
3	Hazmat Suits	70.23	46
4	Stethoscope	60.23	34
7	Thermometer	520.29	7
8	Antibiotics	35.13	121
9	Antiseptics	20.13	73

Place Your Order:

You are on Admin Pannel

1. Total Cash Today
2. View Card Pay
3. Add supply
4. Delete Supply
5. Instant supply List
6. Item Counter
7. Backup System
8. Instant Order Preview
0. Main Menu

Symptom Checker

COVID-19

The Symptom Checker helps in diagnosing the user for COVID 19 with the help of Machine Language. The user will be asked several question related to finding about the existence of the above mentioned symptoms, and based on the input, the system will decide if the user has COVID 19 or not. The user input will be taken in as binary for easier comparison and will display the probability based on the input.



Untitled - Jupyter Notebook

localhost:8888/notebooks/Untitled.ipynb

Other bookmarks

Logout

FileEditViewInsertCellKernelWidgetsHelp

TrustedPython 3

Run

Code

[5.1, 4.4, 1.3, 0.4],

[5.4, 3.9, 1.3, 0.4],

[5.1, 3.5, 1.4, 0.3],

[5.7, 3.8, 1.7, 0.3],

[5.1, 3.8, 1.5, 0.3]

In [3]:

```
x, y = load_iris(return_X_y = True)
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.25, random_state = 0)
print(x_train.shape)
print(y_test.shape)
```

```
(112, 4)
(38,)
```

In [4]:

```
cnb = CaltegoricaNB()
```

In [5]:

```
cnb.fit(x_train, y_train)
y_pred = cnb.predict(x_test)
print(y_pred)
```

```
[2 1 0 2 0 2 0 1 1 1 2 1 1 1 1 0 1 1 0 0 1 1 0 0 1 0 0 1 1 0 2 1 0 1 2 1 0
 2]
```

In [6]:

```
print("Number of mislabeled points out of a total %d points: %d"%(x_test.shape[0], (y_test != y_pred).sum()))
```

```
Number of mislabeled points out of a total 38 points: 4
```

In [9]:

```
from sklearn import metrics
print('Accuracy: ', metrics.accuracy_score(y_test, y_pred))
print('Confusion Matrix\n', metrics.confusion_matrix(y_test, y_pred))
```

```
Accuracy: 0.8947368421052632
Confusion Matrix
[[13  0  0]
 [ 0 15  1]
 [ 0  3  6]]
```

```
In [1]: from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import CategoricalNB
from sklearn.datasets import load_iris
```

```
In [2]: from sklearn import datasets
iris = datasets.load_iris()
print("Shape: ", iris.data.shape)
print("Features: ", iris.feature_names)
print("Output: ", iris.target_names)

Shape: (150, 4)
Features: ['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']
Output: ['setosa' 'versicolor' 'virginica']
```

```
In [11]: load_iris()
```

```
Out[11]: {'data': array([[5.1, 3.5, 1.4, 0.2],
                           [4.9, 3. , 1.4, 0.2],
                           [4.7, 3.2, 1.3, 0.2],
                           [4.6, 3.1, 1.5, 0.2],
                           [5. , 3.6, 1.4, 0.2],
                           [5.4, 3.9, 1.7, 0.4],
                           [4.6, 3.4, 1.4, 0.3],
                           [5. , 3.4, 1.5, 0.2],
                           [4.4, 2.9, 1.4, 0.2],
                           [4.9, 3.1, 1.5, 0.1],
                           [5.4, 3.7, 1.5, 0.2],
                           [4.8, 3.4, 1.6, 0.2],
                           [4.8, 3. , 1.4, 0.1],
                           [4.3, 3. , 1.1, 0.1],
                           [5.8, 4. , 1.2, 0.2],
                           [5.7, 4.4, 1.5, 0.4],
                           [5.4, 3.9, 1.3, 0.4],
                           [5.1, 3.5, 1.4, 0.3],
                           [5.7, 3.8, 1.7, 0.3],
                           [5.1, 3.8, 1.5, 0.3])}
```

Progress

COVID-19

Project Review 1:

- 1) Idea confirmed.
- 2) Revision on Machine Learning done.

Project Review 2:

- 1) Chatbot has been added to the website to test modules linking with all functions and interactions operational. No plugins needed. Google Key API to be added 15 days before final review so as to last the trial period
- 2) Website has been made to accommodate access to all modules.
- 3) Program for our Medical Management System has been successfully implemented. Addition of command prompt to run program in html yet to be added.
- 4) Machine Learning Notebook still in testing phase with other subjects for our Symptom Checker.

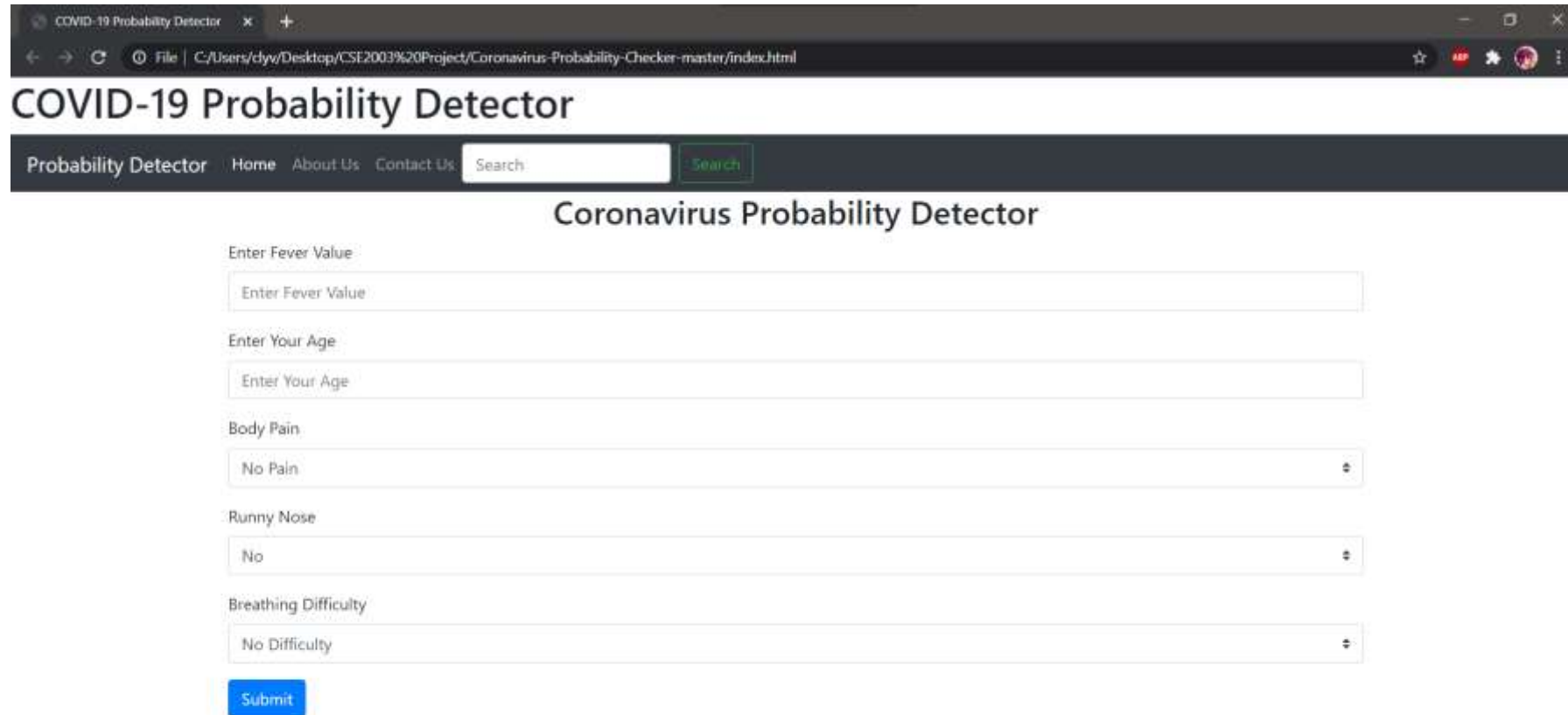
Project Review 3:

- 1) Stable Online IDE not found. Cannot run conio.h and windows.h package.
- 2) Chatbot implemented with API linking ambulance extraction to google maps.
- 3) Symptom Checker functional.
- 4) Website functional offline-no hosting yet.
- 5) Code fully functional along with admin privileges.
- 6) Chatbot ready.

Data Structure is 2-dimensional array

[illegible]

Time Complexity: $O(1)$



The screenshot shows a web browser window with the title "COVID-19 Probability Detector". The address bar displays the file path "C:/Users/dlyw/Desktop/CSE2003%20Project/Coronavirus-Probability-Checker-master/index.html". The page has a dark header with the title "COVID-19 Probability Detector" and a navigation bar with links "Probability Detector", "Home", "About Us", and "Contact Us". A search bar with a "Search" button is also present. The main content area is titled "Coronavirus Probability Detector" and contains a form with the following fields:

- Enter Fever Value:** A text input field with the placeholder "Enter Fever Value".
- Enter Your Age:** A text input field with the placeholder "Enter Your Age".
- Body Pain:** A dropdown menu with the selected option "No Pain".
- Runny Nose:** A dropdown menu with the selected option "No".
- Breathing Difficulty:** A dropdown menu with the selected option "No Difficulty".

At the bottom of the form is a blue "Submit" button.

Time Complexity: $O(1)$



Future Enhancements

- This was supposed to be a system for the pandemic but seeing the scale of it's module can expanded to form a universal system.

Upcoming features:

- Medicine Recommendation System(No need for meeting a doctor for prescribing)
- Prescription Generator
- Video Calling feature in Chatbot
- Forum for Website for Trainees and Patients
- Symptom Checker for more diseases

Experience

COVID-19

Aryaman->To me,it was a project to create something never done before by using data structures like trees,linked list and graphs to form a prototype of a chatbot to aid in the battle against the pandemic.I was lucky enough to get a team in these times which were more than able to stand up on this oppurtunity.I hope to keep developing unique stuff in the coming future.Our 2nd review went a bit awry but Sir's guidance and kindness came through to inspire us to finish what we started.

Clivin->At first it was tough since I was yet to catch up on my ML algorithms but with the help of Arjun,Aryaman and Harsanth I was able to create a working symptom checker and learn even newer things in the field of ML and Deep Learning.

Harsanth->I loved this team which kept me motivated especially Aryaman whom I relentlessly disturbed at times so that he could tell me the ropes behind the development of these modules.It was a symbiotic relation in our team which helped us be better at understanding the concepts of Data Structures used in the project.I loved debugging these modules and creating a working website for this team.

Arjun->It was such an awesome experience with this team creating such a great project.It's even more unbelievable that we may get a patent on our project.Well....another feather on our hats then.Thank you Sir for giving us this oppurtunity in developing such a great system.

Acknowledgements

COVID-19

Thanks to Sir Oswald for inspiring and guiding us through this project. We couldn't have got the chance on working on this if it not been for you.

Thanks go out to our Friends and Family for giving us tips to use in our projects.

Thanks Clivin for working yourself out over the symptom checker which is the most complex module in this project.

Thanks Harsanth for helping us in the coding and website.

Thanks Aryaman for developing the chatbot from scratch which works to great extents.

Thanks Arjun for coding and helping us clear errors in our modules.



THANK YOU

Sir for hearing our idea and we will welcome any suggestions, including possible additions to the J-Component.

