VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

CSE3001 : SOFTWARE ENGINEERING

TOPIC - POSBOT(A POSTURE CORRECTING BOT)

Winter Semester 2020-21

Sai Chaitanya ~ 19BCE0727
Mohammed Nadish ~ 19BCE2465
Soham Sengupta ~ 19BCE0857

TABLE OF CONTENTS

I. ABSTRACT

Our project aims to monitor the fellow human being's posture when he/she spends their time in front of the screens and notify the person when they are slouching or maintaining a bad posture. Moreover, we aim to increase the awareness among the people about maintaining a bad posture, and other harmful effects of spending too much time in front of their screens and how to avoid them. There are 5 major functionalities of are project that are listed below:

- **Build & Deploy a Classification Model:** We have created a Classification model on what is called as the “Good Posture” and “Bad Posture”, by training a large amount of data over an *Echo AR Model* using *TensorFlow.js* . We will mainly classify the posture into 3 main categories i.e. “Too Close to the Screen”, “Slouching”, ”Perfect Posture”.
- **Live Notifications:** Once the client maintains a bad posture over *a specific period of time(about 30-40 seconds)* a notification pops from the web app, as a reminder to correct the posture. This notification will suggest the appropriate correction(eg. Maintain distance from Screen, Keep your back upright, etc).
- **Forum based Practitioner Recommender:** If the client feels he/she should get proper guidance over their physique, then they may *search practitioner forum*, to get to know the well-known orthopedic practitioner that lives in his/her proximity.
- **Implement other Augmented Reality Models:** We believe it is important for the children to know the importance of maintaining proper physique over their nervous system. We plan to implement this using an *Augmented Reality Module of the human body* so as the children find it engaging and informative.
- **Easy Navigation using Chat-Bot:** Once the other features are properly tested and made sure is in proper use, we plan to create a chat-bot that could ease the user over navigation of the website. It also contains a questionnaire that can give proper suggestions based on some common symptoms shown.

II. INTRODUCTION

In these difficult times, where everyone is forced to work remotely and with the mode of schools and colleges going digital, students are spending time on the screen than ever before, it not only affects students but also employees who have to sit for hours in front of the screen. Prolonged exposure to a computer screen and sitting in a bad posture can cause severe health problems like postural dysfunction and affect one's eyes. Over time, poor posture can cause these natural curves to change shape, putting an excessive amount of pressure in the wrong position. Hence we believe these ill effects can be avoided by mere supervision that our project aims to provide.

III. RELATED WORKS

1. Posture-Corrector-Bot : This project has created an archetypal discord bot that, in the future, can help lots of people to be aware of their improper postures as we continue to improve its functions.
2. NagBot : NagBot is a discord bot written in Python using the discord.py API wrapper to remind users of water and posture checks. At the moment, it is intended to run on a single Discord server per bot instance.
3. Posture-Checker : A realistic time unconstrained posture checker that was implemented through discord. This bot was written in Python using the discord.py module.
4. Check & Improve Posture : This blog talks about the importance of maintaining a good posture and what it is exactly that is meant by the term “good posture”. This talks about a posture ratio that the distance from the eye to the length of the shoulder that should be maintained so that a good posture can be achieved.
5. Posture-Watch : Monitor Good Posture - with Artificial Intelligence. Ensure proper office ergonomics with Tensorflow, a webcam and openCV Python real time computer vision libraries.

IV. REAL-LIFE APPLICATIONS

This product can be used over all the platforms and at all the times when a person is working on is workspace over a long time. Especially in this pandemic era of COVID-19, we have increased digital life where it is important to take care of ourselves. Recent research suggest that postural imbalance in early stages of life may become more dangerous and incurable at the later ages.Sitting and standing with proper alignment improves blood flow, helps keep your nerves and blood vessels healthy, and supports your muscles, ligaments, and tendons. People who make a habit of using correct posture are less likely to experience related back and neck pain. Moreover it is necessary that people know the benefits of maintaining a good posture. For example, When your bones and joints are in correct alignment, it allows the muscles to be used as they're intended, so you'll have less fatigue and more energy. This is also another application of the POSBOT website i.e. to increase awareness about benefits of posture.

So now using POSBOT we aim to implement a way for people to monitor their posture, while they are crunching their livelihood in a digital era.

V. INDIVIDUAL CONTRIBUTIONS

A. Software Engineering Part:
   - Brainstorming & Performing done by all.

B. Implementation Part:
   - Fetching of Datasets : Soham Sengupta & Mohammed Nadish
   - Creation of Classification Model : Sai Chaitanya
   - Structural Frontend for Model testing : Soham Sengupta
   - Backend structure for main POSBOT : Sai Chaitanya
   - Backend - Database connections : Mohammed Nadish
   - Frontend : Sai Chaitanya & Soham Sengupta
   - UI/UX for POSBOT : Mohammed Nadish

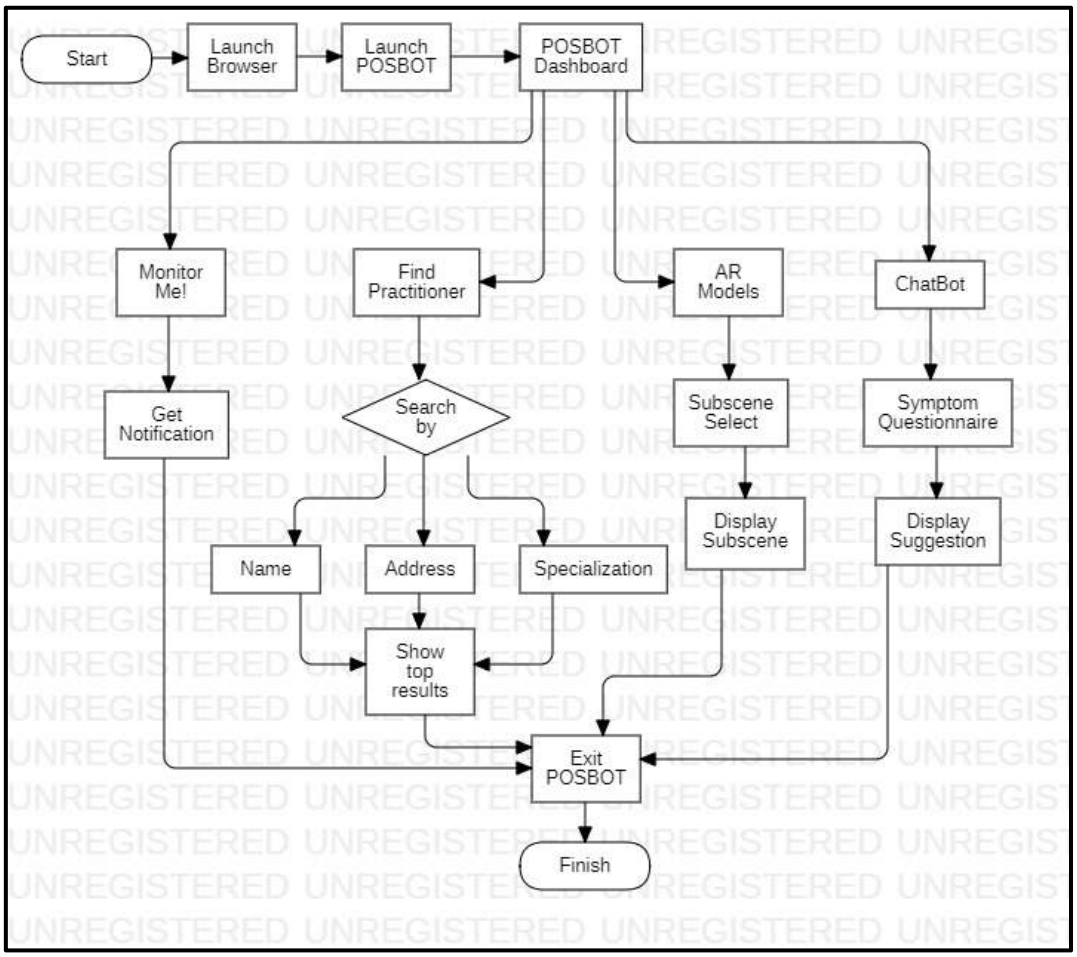
VI. TOOLS & TECHNOLOGIES USED

- **Backend - Posture Classification Model**
  - ✓ TensorFlow.js
  - ✓ Echo AR

- ✓ Teachable Machines

- **Backend - WebApp**
  - ✓ Php
  - ✓ MySQL
- **Frontend - WebApp**
  - ✓ HTML/CSS
  - ✓ Bootstrap
  - ✓ JavaScript
  - ✓ JQuery
- **UI/UX - WebApp**
  - ✓ Figma
- **Some Additional Tech Stacks**
  - ✓ LandBot (API used for the ChatBot)

VII. PROPOSED SYSTEM FLOW



VIII. WORKING METHODOLOGY

1. PROCESS MODEL:

The core functionality of this project i.e. the classification module of posture highly depends on the data we select and can be updated regularly on the basis of new datasets available. Due to this factor, we plan to implement our project in various versions, hence this project follows the Incremental Process Model.

This project is divided into 3 Main Components:

- Building the Classification Model.
- Building the structure of the Website.
- Building the UI/UX for the Website.

As we can see all three of these components act as stages towards building our project, i.e. we would require the Classification Model to run properly, in order to create a proper skeleton for our website. Then after that, we need an interactive UI to present it as a final product. As we can see the building of the classification model is the basis on which our website stands, hence it has the maximum priority. This way we can also see each of our components can be assigned specific priorities, which is yet another factor resulting in the choice of Incremental Process Model.

Once all these stages are ready as a base, we can start upgrading the product with specific changes and start adding new features that come along the way. For example, we found a bigger dataset with a higher number of categories for classification of posture, we may need to train our model on this new data, and hence, improve our efficiency and accuracy. It may also happen that we want to change the design of our website so as to incorporate more and more features, then we can do so without affecting the previous components.

As our data and features may be continuously changing and does not maintain a fixed structure, other models, like waterflow model, RAD model, prototype model may not be suitable for our project.

Hence through following this structure, we hope to test and debug each of our component with great ease, and in turn recognize errors with efficiency and flexibility.

2. SRS DOCUMENT

2.1 Product Perspective

This is a product which detects a bad posture of the user. This can be used be any day to day folks working daily on their screens. This is an excellent product for monitoring our posture while the user works day and night in front of the screens. This product has to have initially a live camera access to the user's webcam so that it can track his/her posture regularly, when a bad posture is maintained, this product is supposed to display a notification to the user to correct their posture. Moreover this product serves other features like, recommending top practitioners habituated in their vicinity and raise awareness to the school children about the harmful effects of maintaining a bad posture using interactive and attractive augmented reality models.

2.2 Product Functions

2.2.1 Deploy Classification Model to identify as to what is known as "bad posture".

2.2.2 Send Live Notification if a bad posture is maintained.

2.2.3 Recommend top practitioners near user, based on a forum.

2.2.4 Showcase the human nervous system to children using augmented reality models so that children find it interesting and attractive.

2.2.5 Implement a chat-bot that will ensure easy navigation over the website.

2.3 Operating Environment

2.3.1 Windows 7 or higher/ any linux based OS/ MacOS

2.3.2 Browser with which is javascript enabled(Chrome prefered)

2.3.3 System with either internal or external webcam.

2.4 Design and Implementation Constraints

2.4.1 Must not run the webcam over another application when enabled on this product.

2.4.2 Javascript/ Adobe Flash must be enabled in the background.

2.4.3 Desktop pop notification must be enabled for this website.

2.4.4 Javascript version must be v2.4.0 or higher.

2.5.5 New practitioner cannot be recommended if they are not present in the database.

2.5.6 A lot of traffic on the server may slow the website.

2.5 Assumptions and Dependencies

2.5.1 JS version of the browser to be v2.4.0 or higher.

2.5.2 Echo AR base models to be used.

2.5.3 User has a proper internet connection.

2.5.4 User must be knowledgeable enough to know about computers.

2.5.5 User must be aware of how an augmented reality models work.

2.5.6 User must have a good resolution webcam.

3. SYSTEM FEATURES

3.1 Classification Model

3.1.1 Description and Priority

Here the user is asked to enable their webcam permission and the live video feed is shown to in a small window. The shown feed also points the skeletal structure of the user's shoulder & forehead edges and compares this with optimal posture required. Priority: 8

3.1.2 Functional Requirements

REQ-3.1.2.1:

Input: "Allow Camera" option selected

Output: Small window shows their live video feed.

REQ-3.1.2.2:

Input: Live video feed

Output: Animates the Skeletal Posture of the user.

Processing: Send video query to Echo AR[(1)] to extract skeletal structure of each video frame.

3.2 Live Notifications

3.2.1 Description and Priority

In this section the user needs to grant permission to the website to push notifications through their browser. Once granted permission then the website will push

notification if he/she maintains improper posture for some time(30-40 secs). Priority: 7

3.2.2 Functional Requirements

REQ-3.2.2.1:

Input: “Allow Notification” option selected

Output: Message Acknowledgement “Processing Live Video Feed” REQ-3.2.2.2:

Input: Live video feed

Output: If categorized as “Slouching Posture” or “Too Close” then the website prompts a message to the user as “You are slouching!! please correct your posture.” and “Too close to the screen!! please maintain proper distance from the screen”.

Processing: The live feed is sent to the Tensorflow.js[(2)] model to classify each video frame feed as “Perfect Posture”,”Slouching Posture” or “Too Close”.

3.3 Practitioner Finder

3.3.1 Description and Priority

The user can search for top practitioners habituated in his/her vicinity, by filling a form about his/her location details or the user can search practitioner by designation, speciality or name. Priority: 7

3.3.2 Functional Requirements

REQ-3.3.2.1:

Input: “Search Practitioner” Selected

Output: User prompted to a form to fill details about their location REQ-3.3.2.2:

Here, in this page the user can find details of the practitioner based on Location, Name, Specialization as key.

REQ-3.3.2.2.1:

Input: “Search Practitioner by Name” selected & “Enter Name” is input.

Output: Shows practitioner details of top matches.

Processing: Retrieve top matches based on key=name from the database.

REQ-3.3.2.2.2:

Input: “Search Practitioner by Specialization” selected & “Specialization” selected.

Output: Shows practitioner details of top matches.

Processing: Retrieve top matches based on key=specialization from the database.

REQ-3.3.2.2.3:

Input:”Search Practitioner by Location” selected & “Location” is input.

Output: Shows practitioner details of top matches.

Processing: Retrieve top matches based on key=street_name from the database.

3.4 Interactive & Informative AR Models

3.4.1 Description and Priority

In this section the user can educate him/herself on different parts of the nervous system and how different types of poor posture can affect their nervous system, using interactive Augmented Reality Models.

Priority: 6

3.4.2 Functional Requirements

REQ-3.4.2.1:

Input: ”Educate Me!!” selected.

Output: Prompts user to Echo AR Models REQ-3.4.2.2:

Here in this page the user can select 8 different Human Body scenes to learn about i.e. Nervous System, Human Brain, Nerves in the Body, Neuron Forest, Neurotransmitters, Cell Membranes & Migraines. REQ-3.4.2.2.1:

Input: “Select Scene” is selected from the given 8 Scenes

Output: The specific scene AR Model is shown for navigation.

3.5 Chat-Bot Helper

3.5.1 Description and Priority

A Chat-Bot is implemented using which the user can easily navigate over the website. Priority: 5

3.5.2 Functional Requirements

REQ-3.5.2.1:

Input: ”Help Me” button is selected

Output: Chat-Bot window is opened.

REQ-3.5.2.2:

Here the user can ask various questions to the chat bot, like navigation to a specific section, or contact information about the developer of the project, in case of crucial queries.

REQ-3.5.2.2.1:

Input: User types “Connect me to Developers!”

Output: Displays Github and Email-Ids of the developers of this product.

REQ-3.5.2.2.2:

Input: User types “Help Me!”

Output: Displays Instructions on how to use the website through a demo video.

REQ-3.5.2.2.3:

Input: User types ”Navigate Me!”

Output: Displays the hyperlinks to the different sections of the website.

REQ-3.5.2.2.4:

Input: User selects a hyperlink Output:

Prompts to the selected section.

4. OTHER NONFUNCTIONAL REQUIREMENTS

4.1 Safety/Security Requirements

4.1.1 Description and Priority

The product provides, proper anonymity to the user, that means the data given to the website is stored in an encrypted md5 format, so that the personal and sensitive information is kept secure from the administrator of the website themselves.

4.1.2 Non-Functional Requirements

REQ-4.1.2.1:

In this segment the data that the user provides to the website stays completely anonymous using different encryption algorithms.

REQ-4.1.2.1.1:

Input: User Enters his/her location details.

Output: Encrypted values in database

Processing: Actual data encrypted into md5 format.

REQ-4.1.2.1.2:

Input: Live user video feed.

Output: Encrypted image files in database

Processing: AES algorithm used to encrypt each video frame.

4.2 Software Quality Attributes

4.2.1 Description and Priority

This product results are based on live day to day data change. The machine learning algorithm is also flexible enough to incorporate on a new data-set with much wide based categories.

4.2.2 Non-Functional Requirements

REQ-4.2.2.1:

The algorithm can have a change in data-set at any point of time and new practitioner details can be added b the product admin at anytime.

REQ-4.2.2.1.1:

Input: Admin adds practitioner details Output:

Modified Database.

REQ-4.2.2.1.2:

Input: Admin removes practitioner Output:

Modified Database.

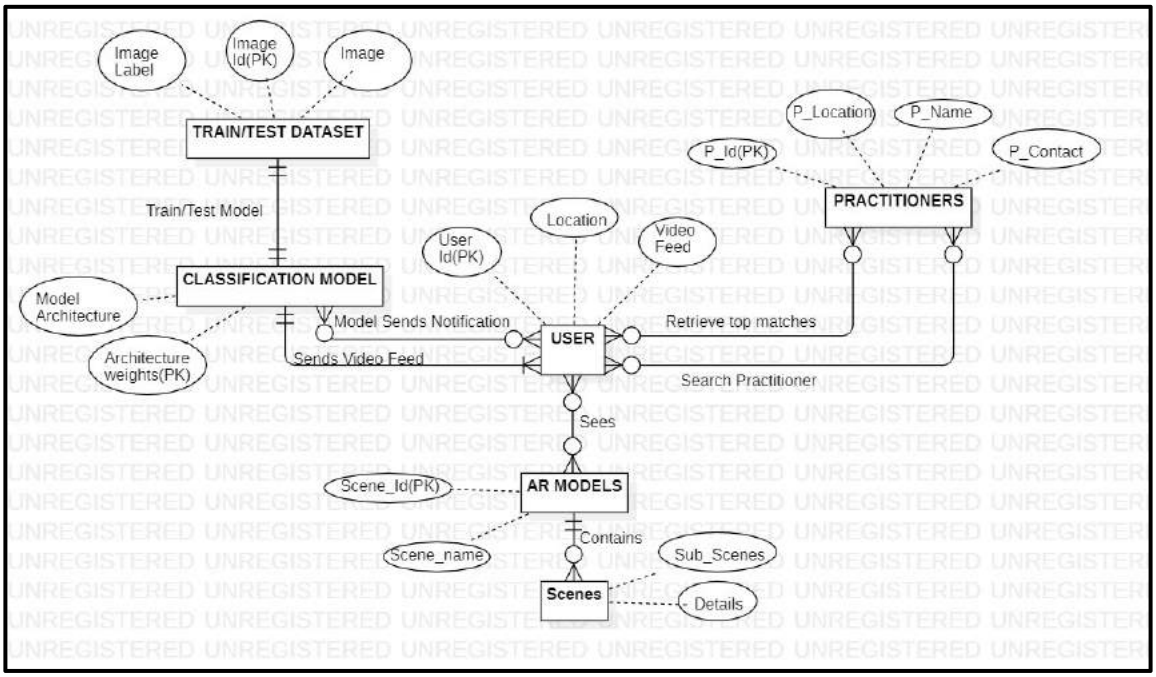
REQ-4.2.2.1.3:

Input: New Classification Data-set used.

Output: Model weights modified.

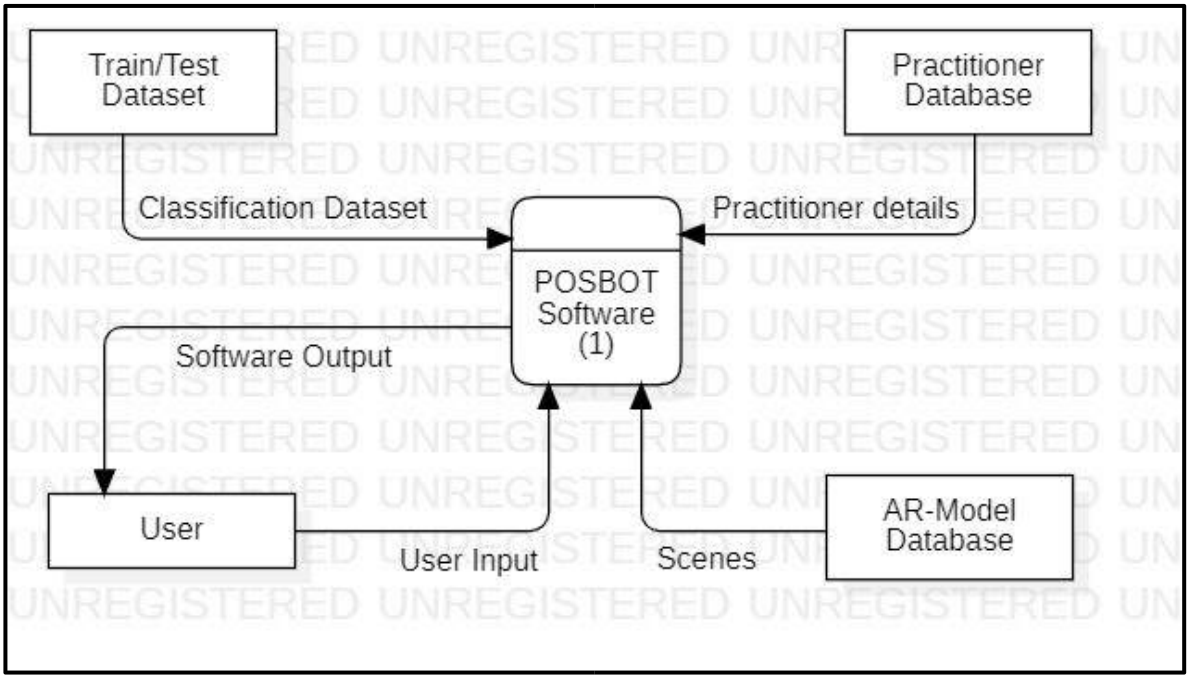
Processing: Trained model on new data-set.

5. ER- DIAGRAM

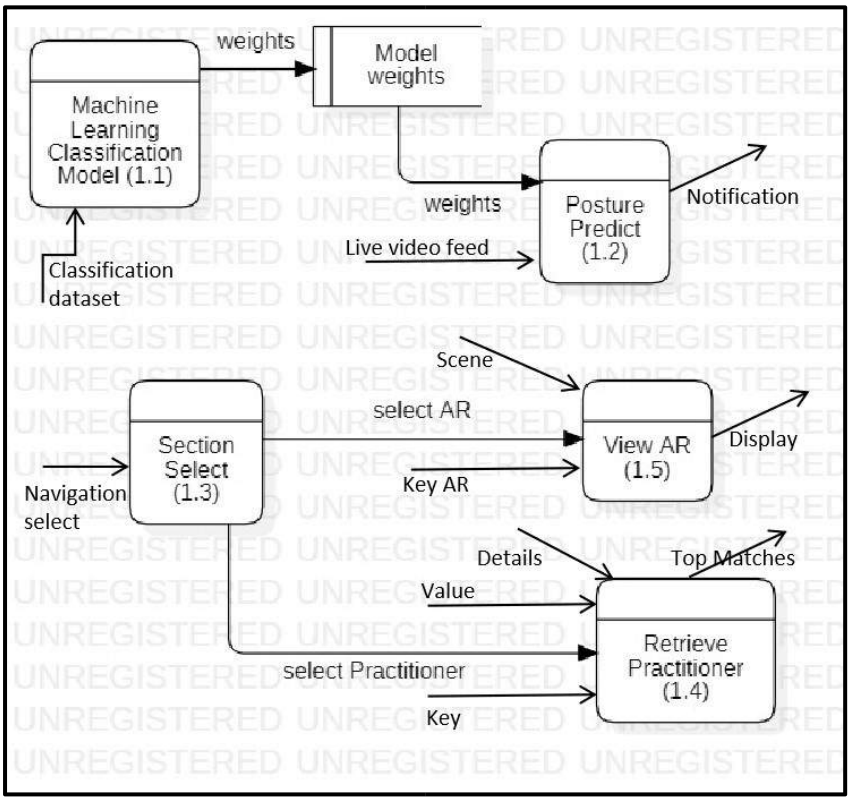


6. DFD & DATA DICTIONARY

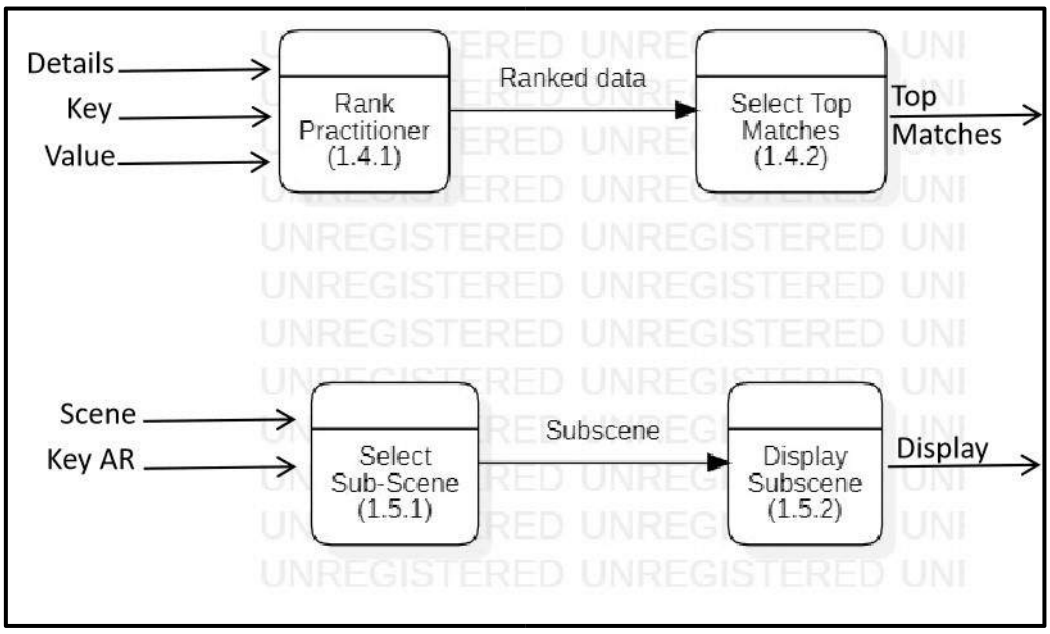
6.1 Context Diagram



6.2 Level-1 DFD



6.3 Level-2 DFD



6.4 Data Dictionary

Classification Dataset : {images}*300

Practitioner Details: {integer + string + string + integer}*1000

Software Output: [string , {ARimage , string} , {integer + string + string + integer}*20]

User Input: [string , video]

Scenes: {ARimage, string}*8

Live video feed: video

Scene: {ARimage, string}*8

Scenes = Scene

Navigation select: string

Key AR: string

Value: string

Key: string

Details: {integer + string + string + integer}*1000

Practitioner Details = Details

User Input=[Live video feed , Navigation select, Key AR , Key , Value]

Select AR: integer

Select Practitioner: integer

Weights: .bin file

Notification: string

Display: {ARimage, string}

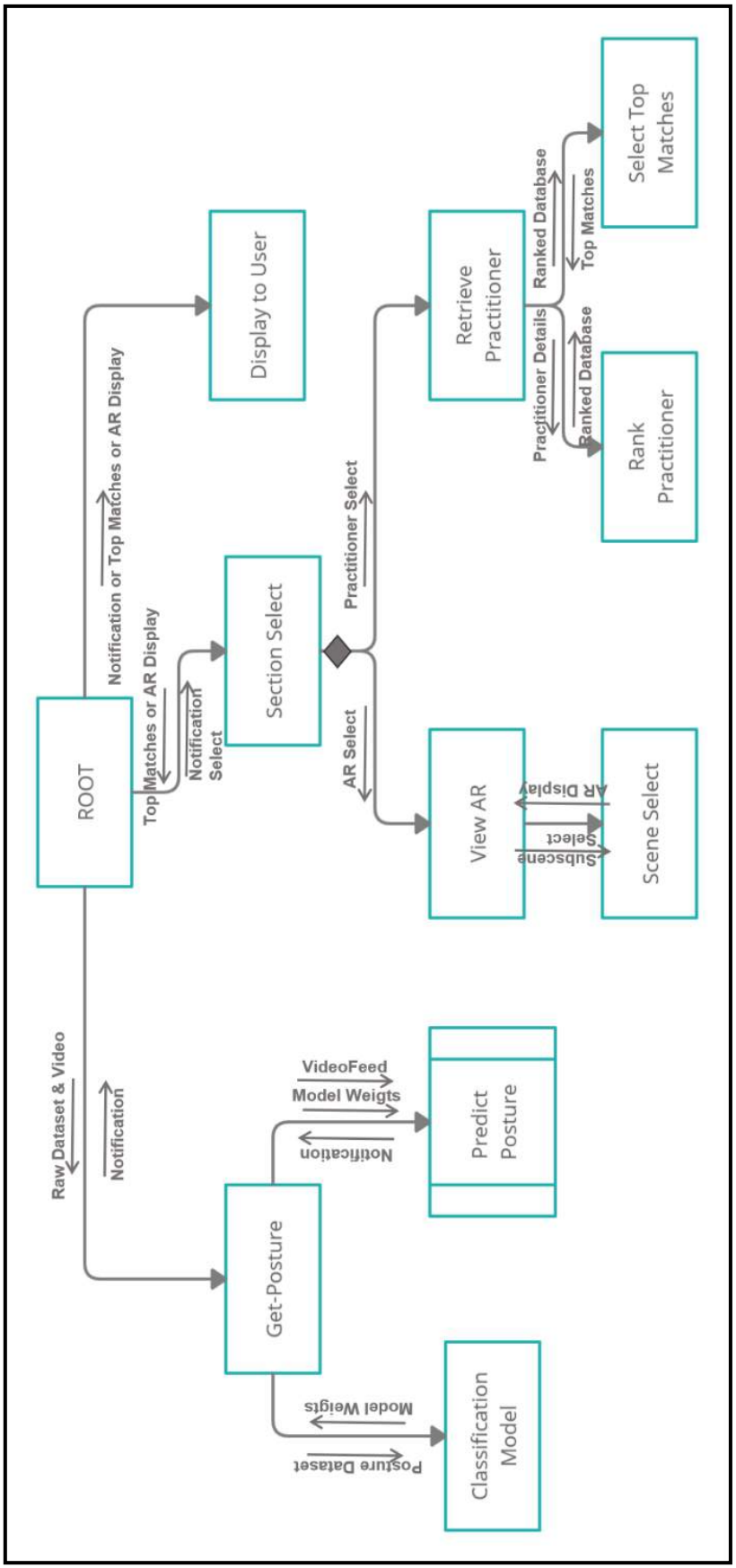
Top Matches: {integer + string + string + integer}*20

Software Output: [Notification , Display , Top Matches]

Ranked Data: {integer + string + string + integer}*1000

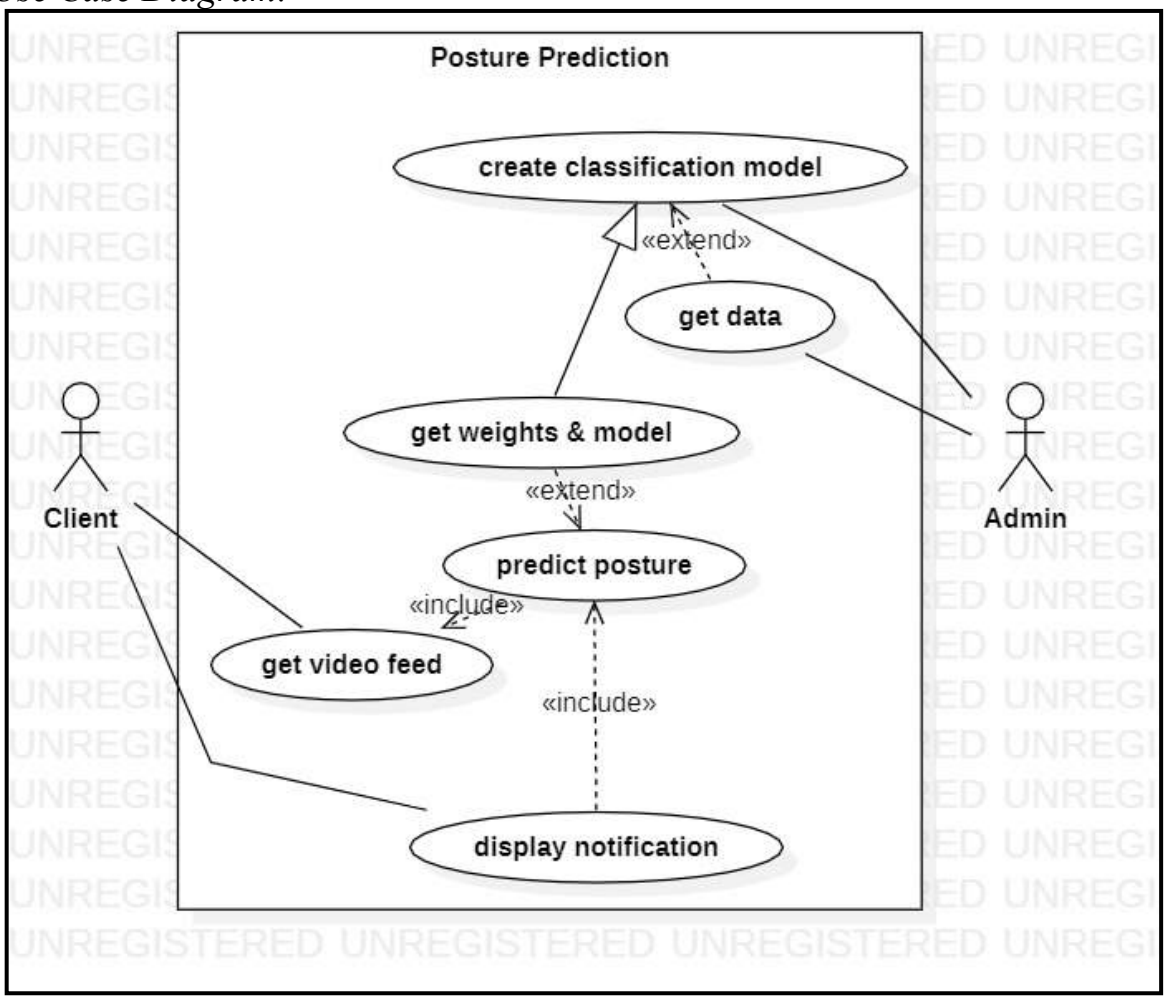
Subscene: {ARimage, string}

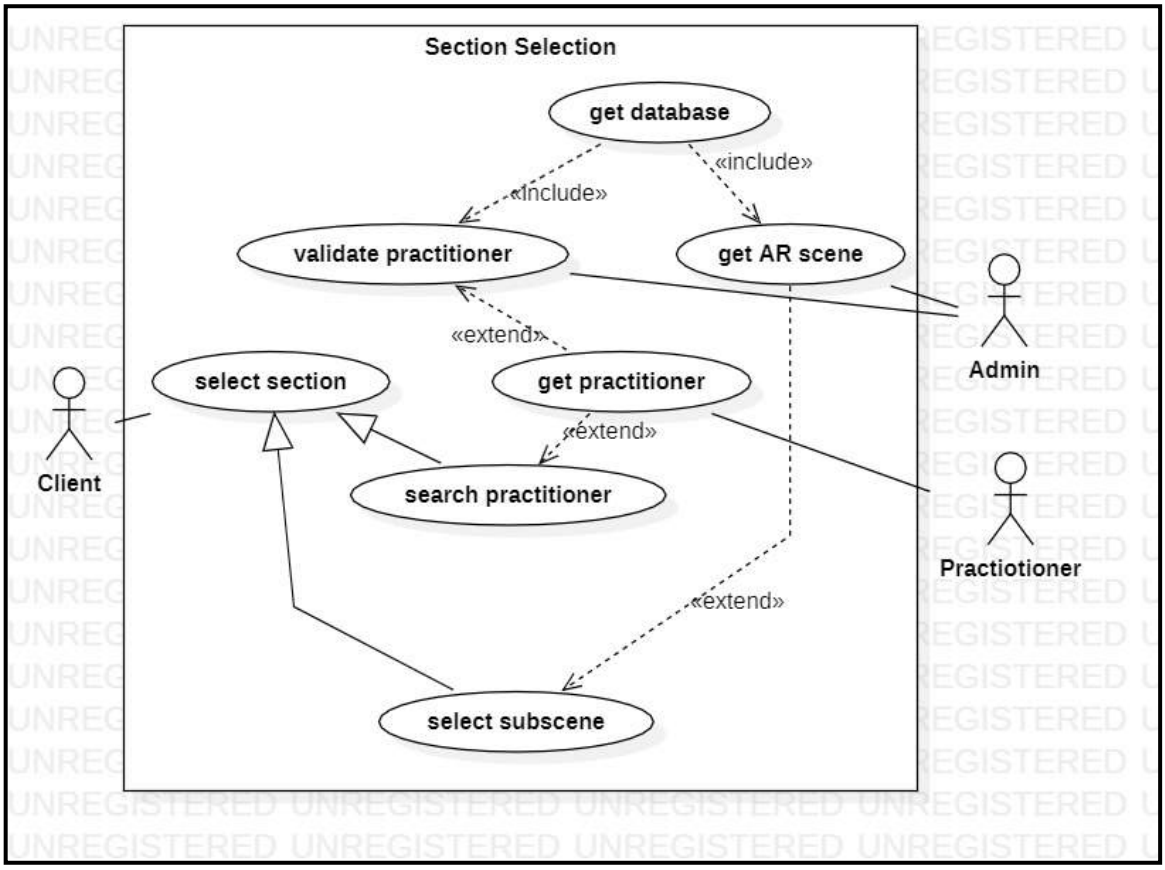
7. STRUCTURE CHART:



8. UML DIAGRAMS:

- *Use Case Diagram:*





➢ Use Case Description:

➢ Use Case ID:	1		
Use Case Name:	Create Classification Model		
Created By:	Soham Sengupta	Last Updated By:	Soham Sengupta
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	Admin
Description:	Creates a classification model from the input values given for training as provided by the admin
Preconditions:	Admin must be logged into the program
Postconditions:	Classification Model is created
Priority:	high
Frequency of Use:	nil
Normal Course of Events:	1) Admin logs into their account 2) Admin provides input data to train the model 3) Classification model is created.
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	Teachable Machine
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	2		
Use Case Name:	Upload Weights and Model		
Created By:	Sai Chaitanya Pentapati	Last Updated By:	Sai Chaitanya Pentapati
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	Admin
Description:	Provides the weights and model required to predict the posture of the user
Preconditions:	Admin must be logged in
Postconditions:	Weights and Model is provided
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) Admin logs into their account 2) Clicks on upload weights and model button 3) Finds the location of the weights and model files 4) Clicks on upload button 5) The files are uploaded to the app
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	3		
Use Case Name:	Get Video Feed		
Created By:	Mohammed Nadish	Last Updated By:	Mohammed Nadish
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	User
Description:	Provides access to their webcam to get the video feed for posture correction
Preconditions:	User must be logged into the app
Postconditions:	Video Feed access is provided
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) User logs into the app 2) Clicks on “Start Posture Detection” button 3) User allows notification by the app to access the video feed 4) Video Feed access is provided
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	4		
Use Case Name:	Predict Posture		
Created By:	Soham Sengupta	Last Updated By:	Soham Sengupta
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	User
Description:	The posture is predicted by the app based on the video feed provided by the user
Preconditions:	User must be logged into the app
Postconditions:	The posture of the user is predicted
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) User logs into the app 2) Clicks on “Start Posture Detection” button 3) User allows notification by the app to access the video feed 4) Video Feed access is provided 5) The app retrieves the weight and training model of the program 6) The app start detecting the posture for the user
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	5		
Use Case Name:	Display Notification		
Created By:	Sai Chaitanya Pentapati	Last Updated By:	Sai Chaitanya Pentapati
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	User
Description:	The posture is predicted by the app based on the video feed provided by the user
Preconditions:	User must be logged into the app
Postconditions:	Notification is provided by the app based on the posture predicted by the app
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) User logs into the app 2) Clicks on “Start Posture Detection” button 3) User allows notification by the app to access the video feed 4) Video Feed access is provided 5) The app start detecting the posture for the user 6) Based on the posture detected by the app, it provides various notifications to the user to improve/correct/encourage the user
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	6		
Use Case Name:	Create Practitioner Database		
Created By:	Mohammed Nadish	Last Updated By:	Mohammed Nadish
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	Admin
Description:	The database used for storing practitioner details is created
Preconditions:	Admin must be logged into the app
Postconditions:	The practitioner database is created based on the data provided by the admin
Priority:	high
Frequency of Use:	nil
Normal Course of Events:	1) Admin logs into the app 2) Admin creates the database of practitioners 3) Admin uploads the database to the app 4) Admin validates the practitioner database
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	Nil

Use Case ID:	7		
Use Case Name:	Fetch Practitioner Database		
Created By:	Soham Sengupta	Last Updated By:	Soham Sengupta
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	Admin
Description:	The database is fetched by the admin
Preconditions:	Admin must be logged into the app
Postconditions:	The practitioner database is fetched (based on the data required by the user) by the admin
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) Admin logs into the app 2) Admin gets the AR Scene 3) Admin validates the practitioner required by the user 4) Admin fetches the database for the required data
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	8		
Use Case Name:	Select Section		
Created By:	Sai Chaitanya Pentapati	Last Updated By:	Sai Chaitanya Pentapati
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

Actor:	User
Description:	The user selects section to open either Posture Predictor or Practitioner finder
Preconditions:	User must be logged into the app
Postconditions:	The required section is opened by the app
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) User logs into the app 2) User selects the desired section 3) The app opens the section to the user
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	9		
Use Case Name:	Search Practitioner Details		
Created By:	Mohammed Nadish	Last Updated By:	Mohammed Nadish
Date Created:	04-04-2021	Date Last Updated:	04-04-2021

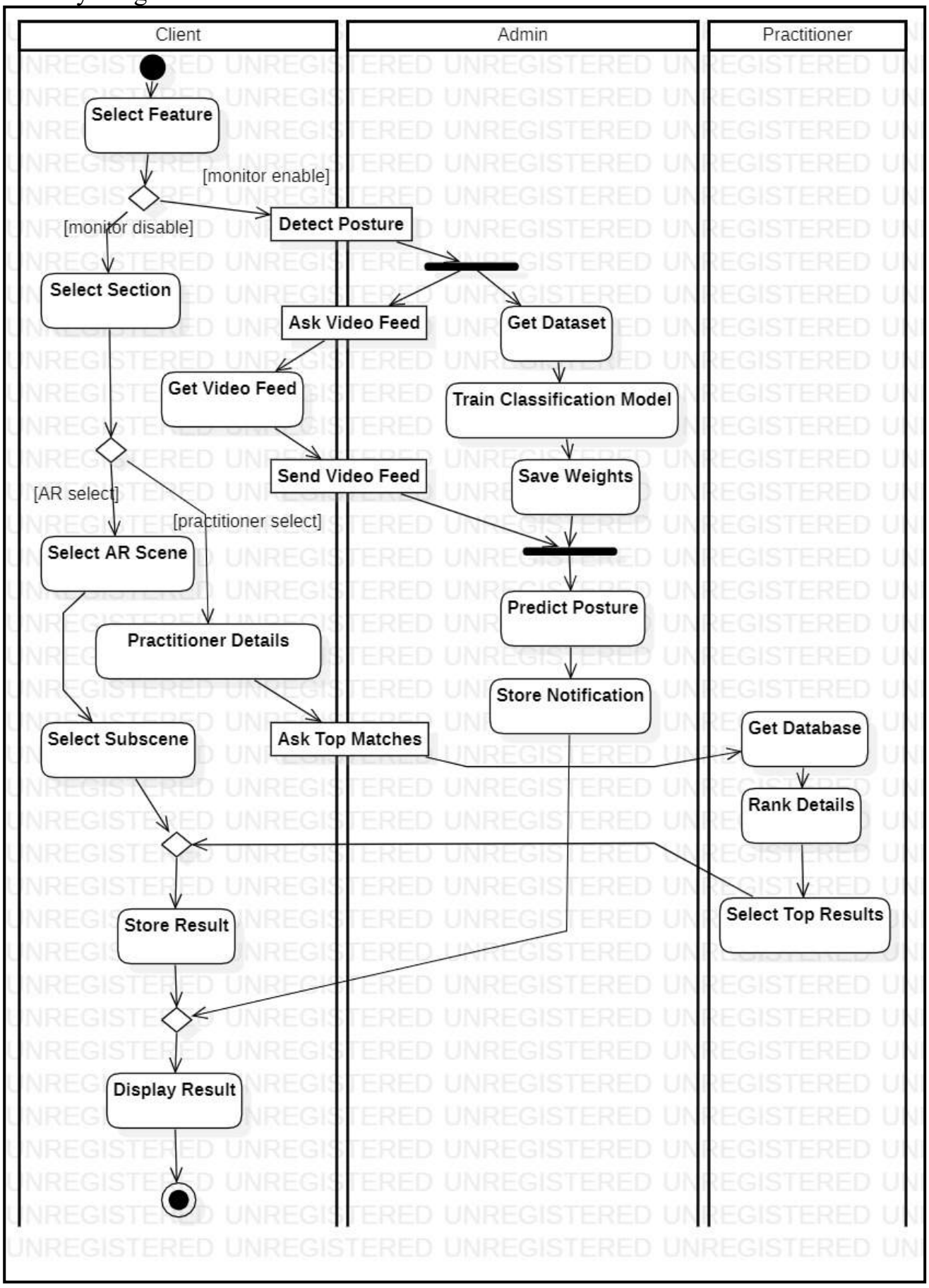
Actor:	User
Description:	The user searches for a practitioner according to his/her requirements
Preconditions:	User must be logged into the app
Postconditions:	The practitioner details is searched by the user in the app
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) User logs into the app 2) User selects the Practitioner section 3) User provides details to filter his/her requirements 4) User clicks the search button to search for the practitioner
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

Use Case ID:	10		
Use Case Name:	Get Practitioner Details		
Created By:	Soham Sengupta	Last Updated By:	Soham Sengupta

Date Created:	04-04-2021	Date Last Updated:	04-04-2021

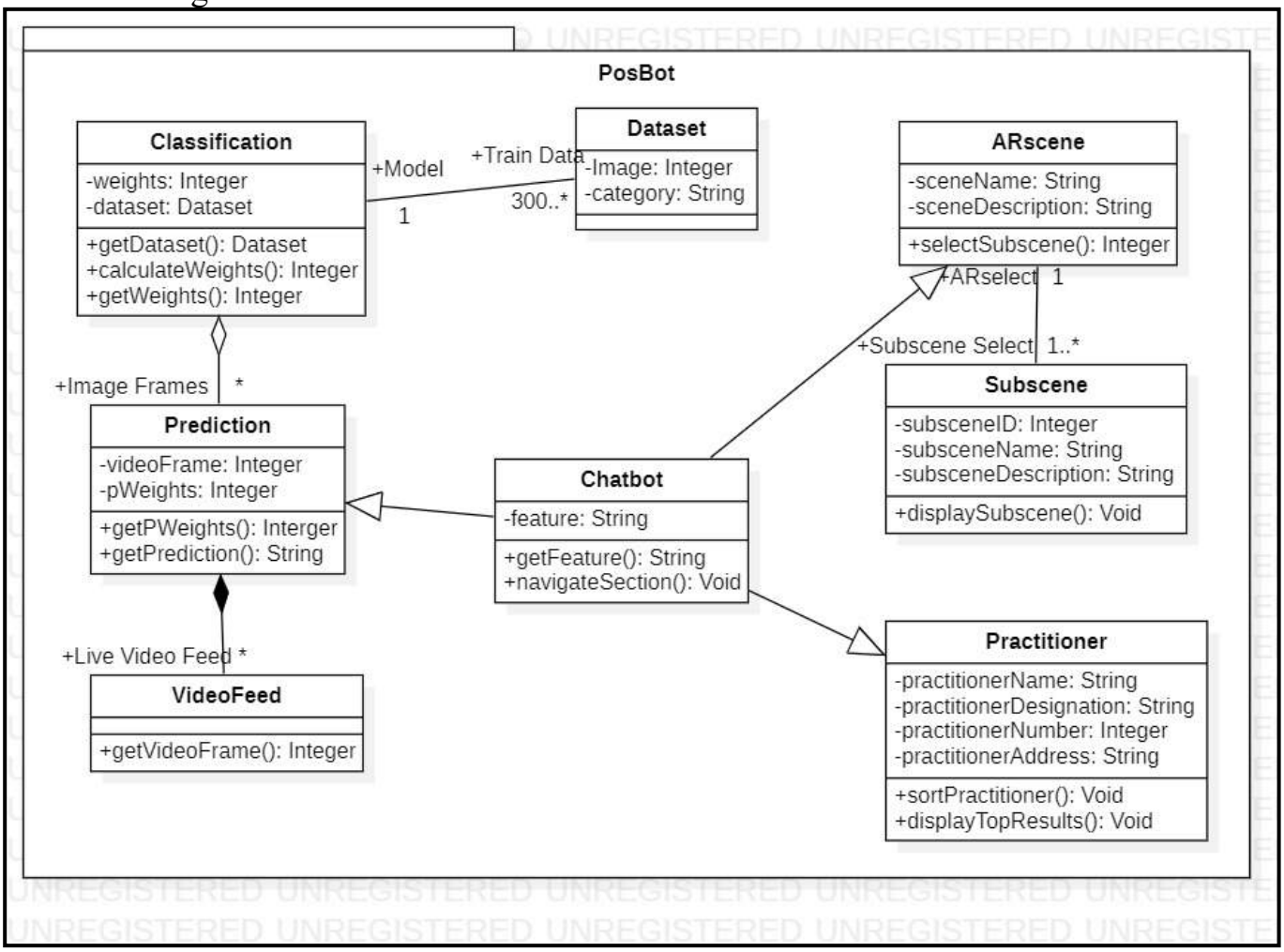
Actor:	User
Description:	The user gets details for a practitioner according to his/her requirements
Preconditions:	User must be logged into the app
Postconditions:	The practitioner details is provided by the app for the user
Priority:	medium
Frequency of Use:	nil
Normal Course of Events:	1) User logs into the app 2) User selects the Practitioner section 3) User provides details to filter his/her requirements 4) User clicks the search button to search for the practitioner 5) App validates practitioner from the database 6) App displays the practitioner details to the user
Alternative Courses:	nil
Exceptions:	nil
Includes:	nil
Special Requirements:	nil
Assumptions:	nil
Notes and Issues:	nil

- Activity Diagram:

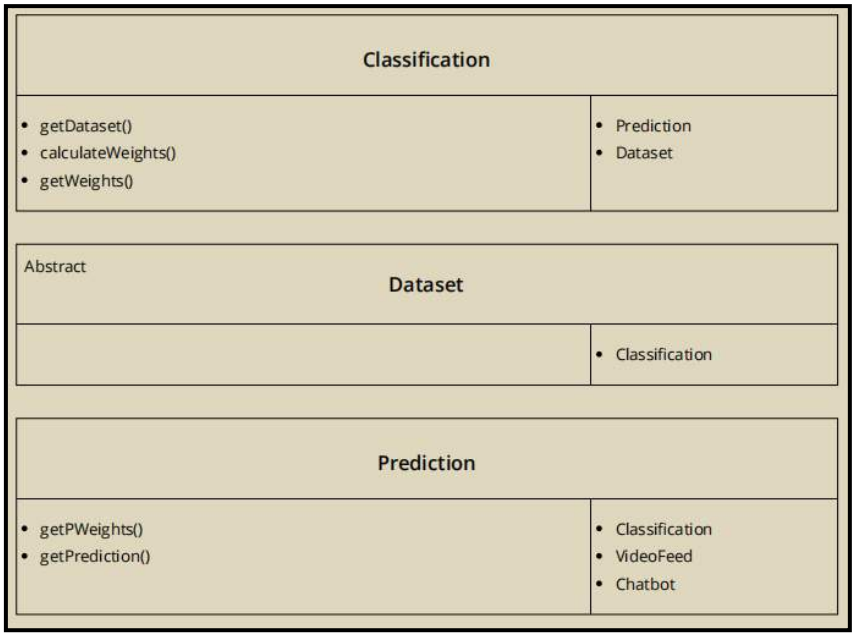


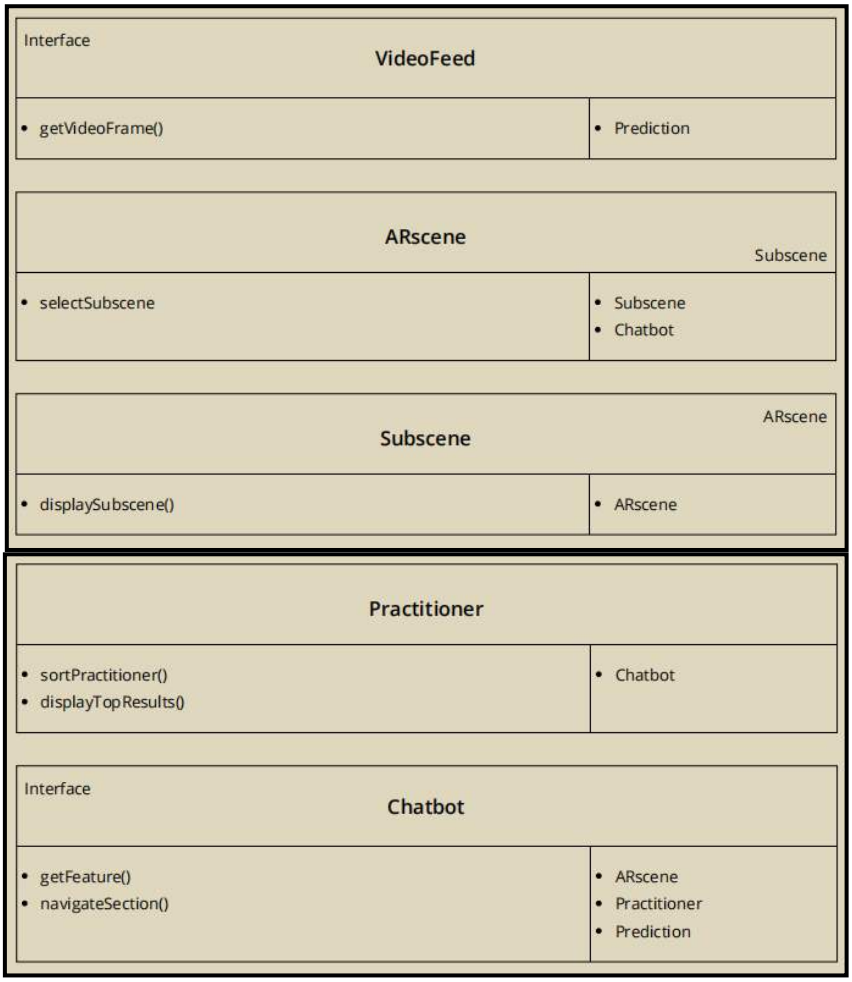
-

- Class Diagram:

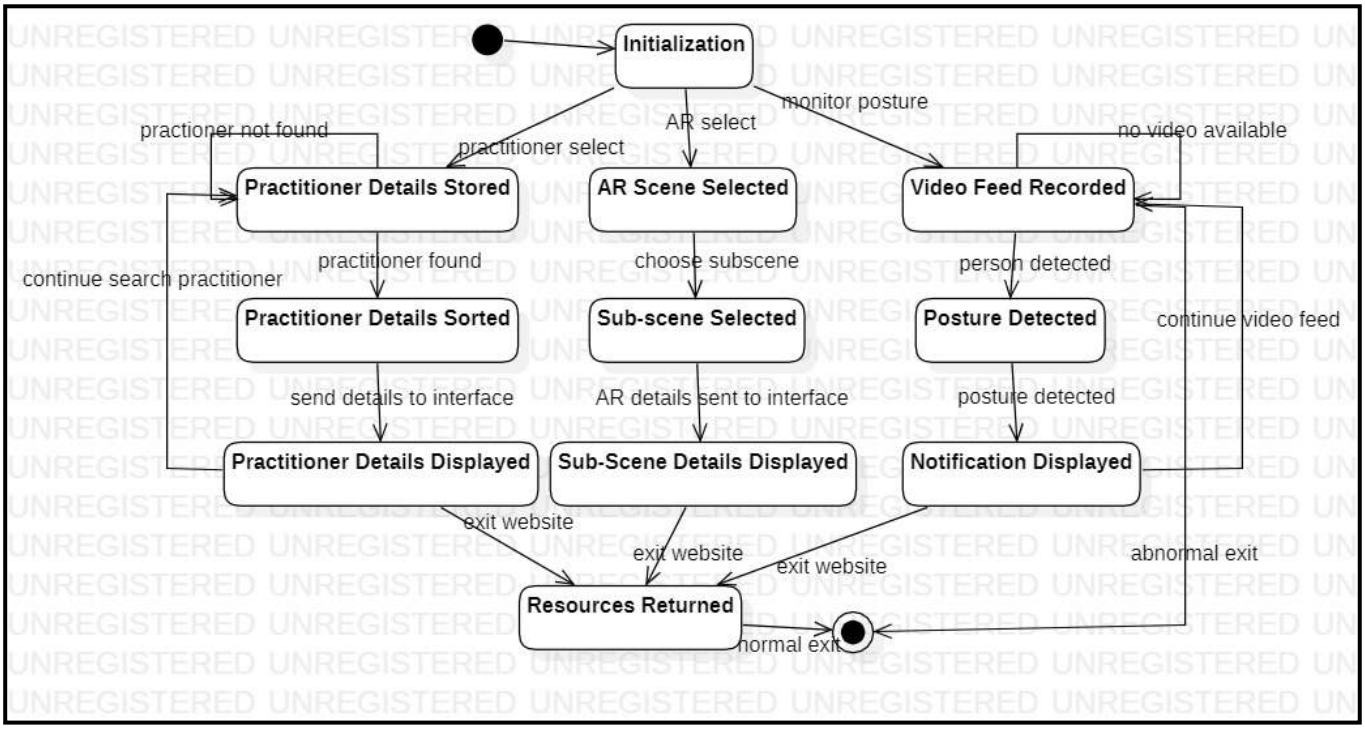


CRC Card:

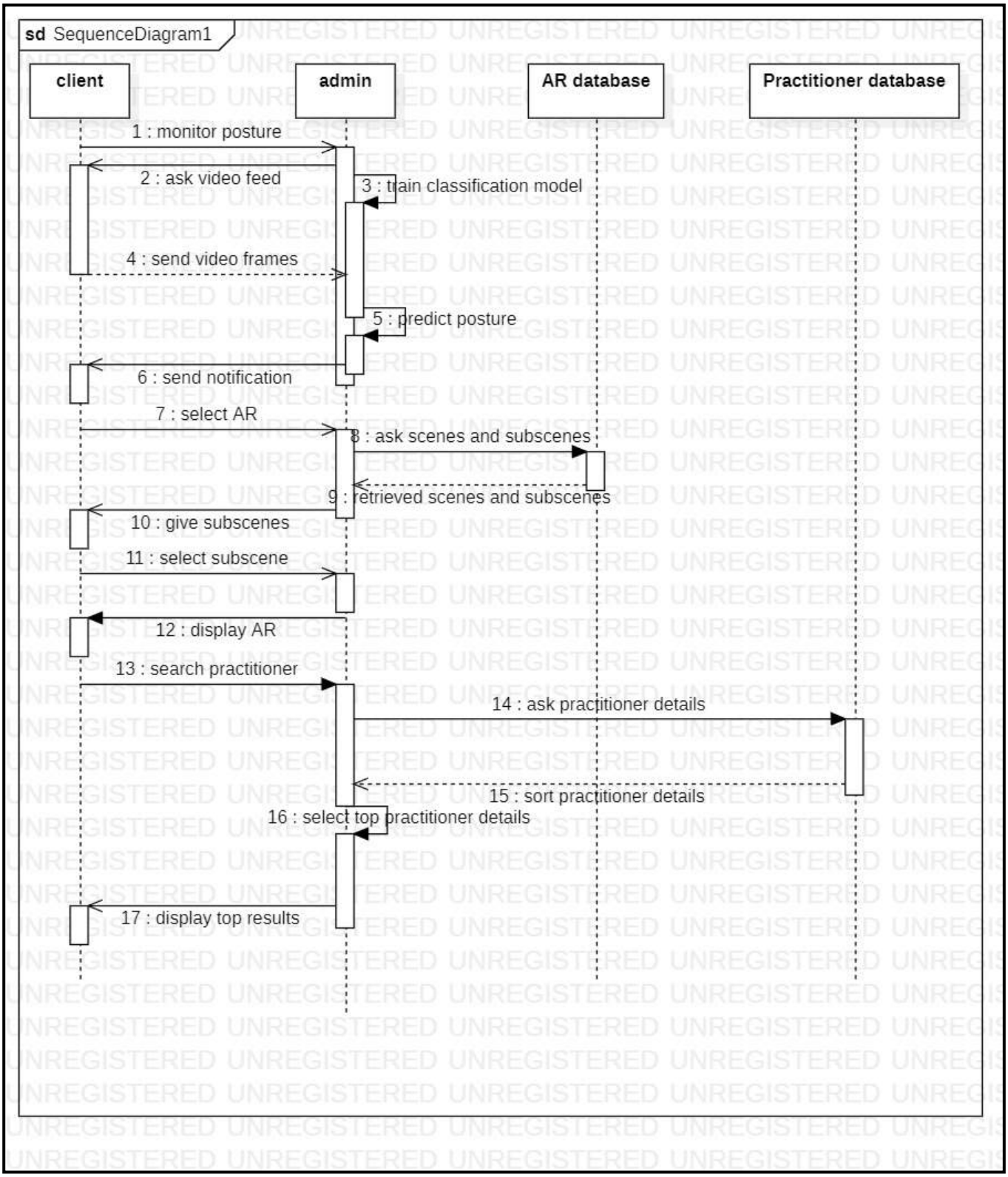




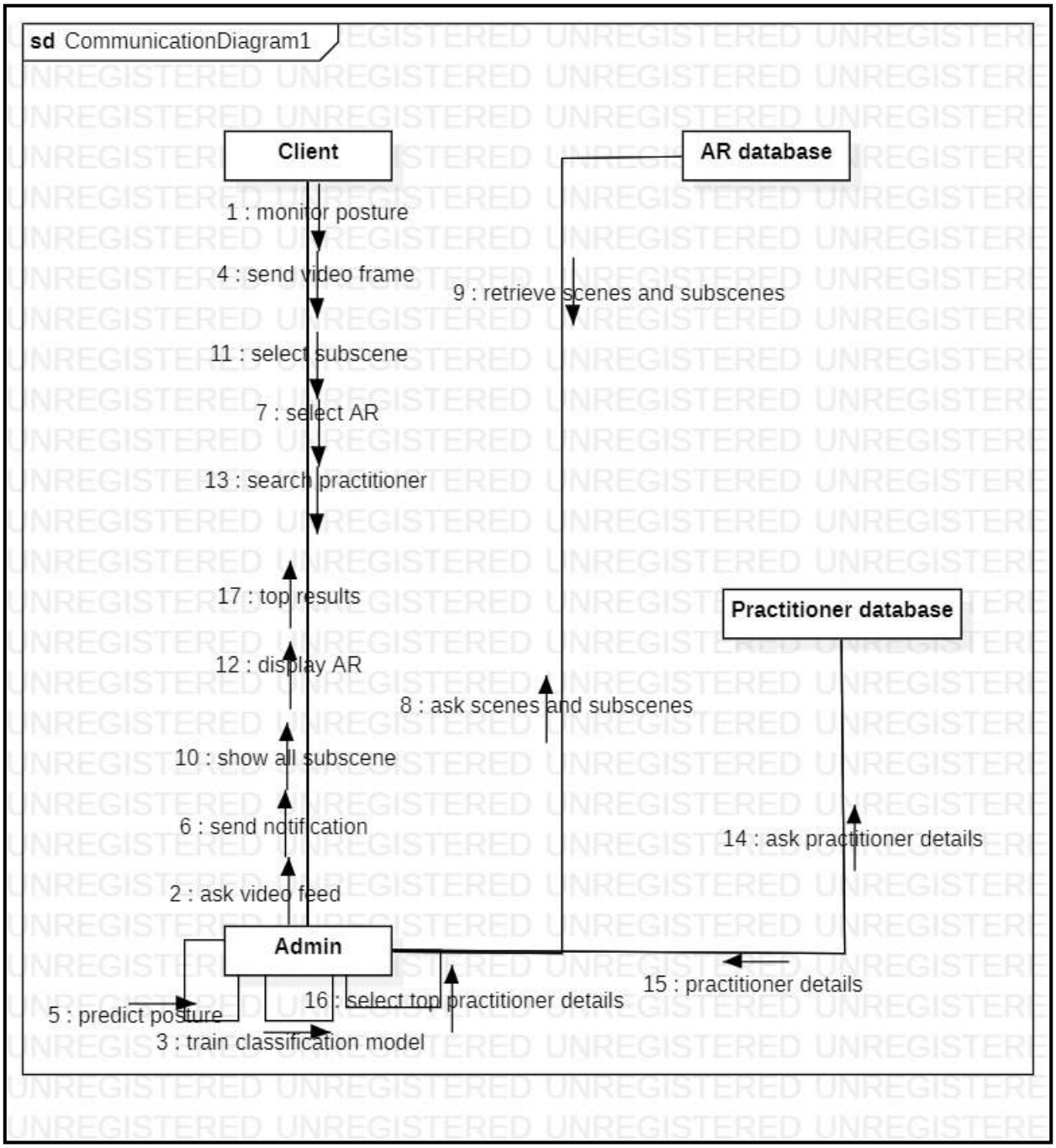
State Chart Diagram:



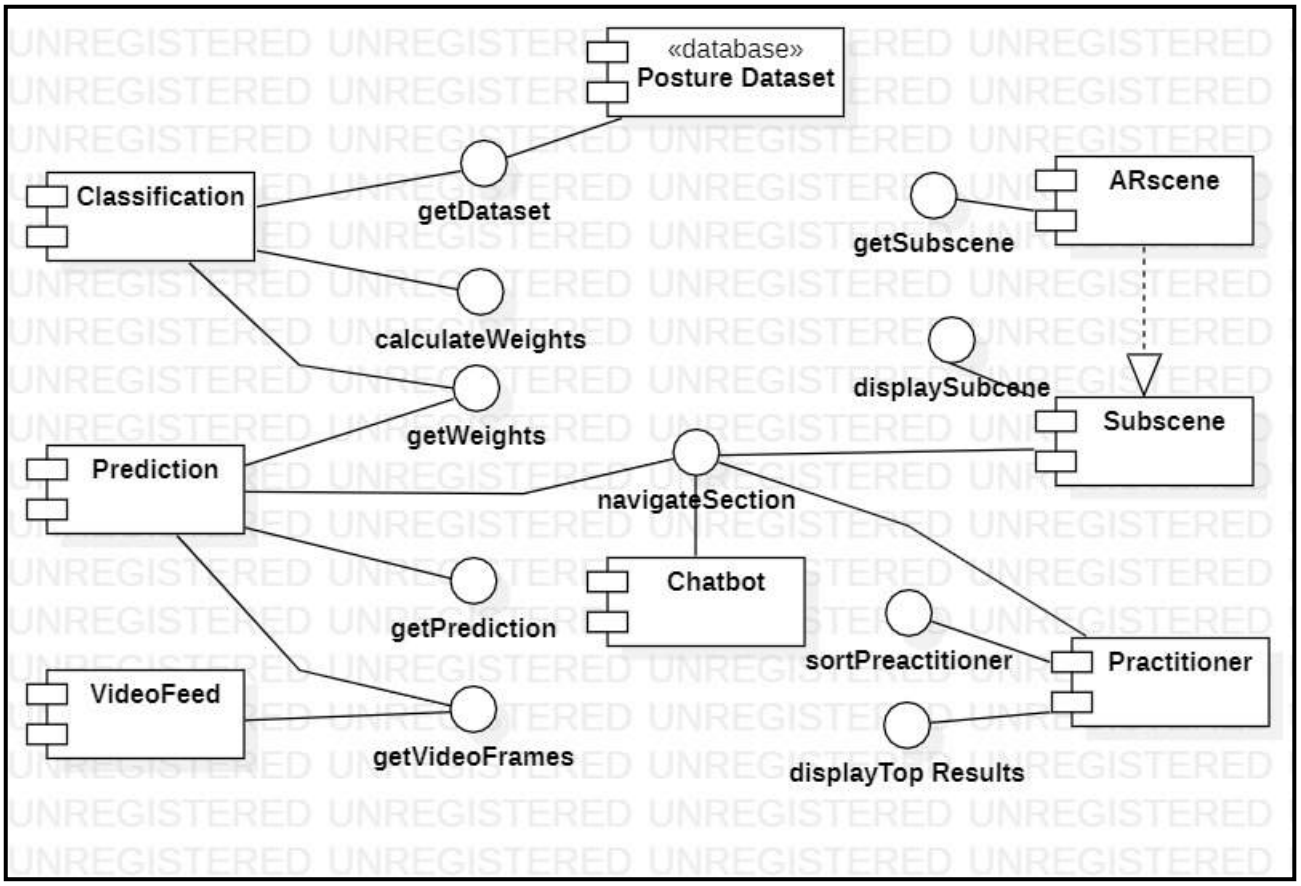
Sequence Diagram:



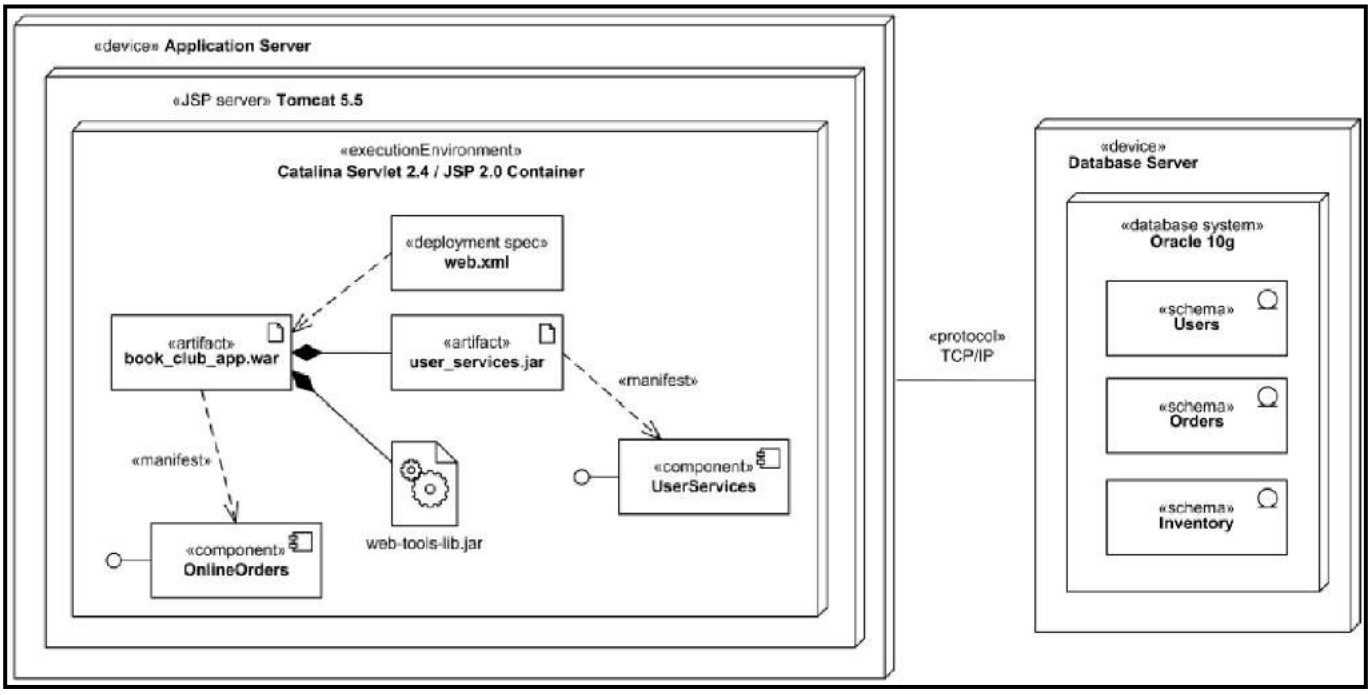
Communication Diagram:



Component Diagram:



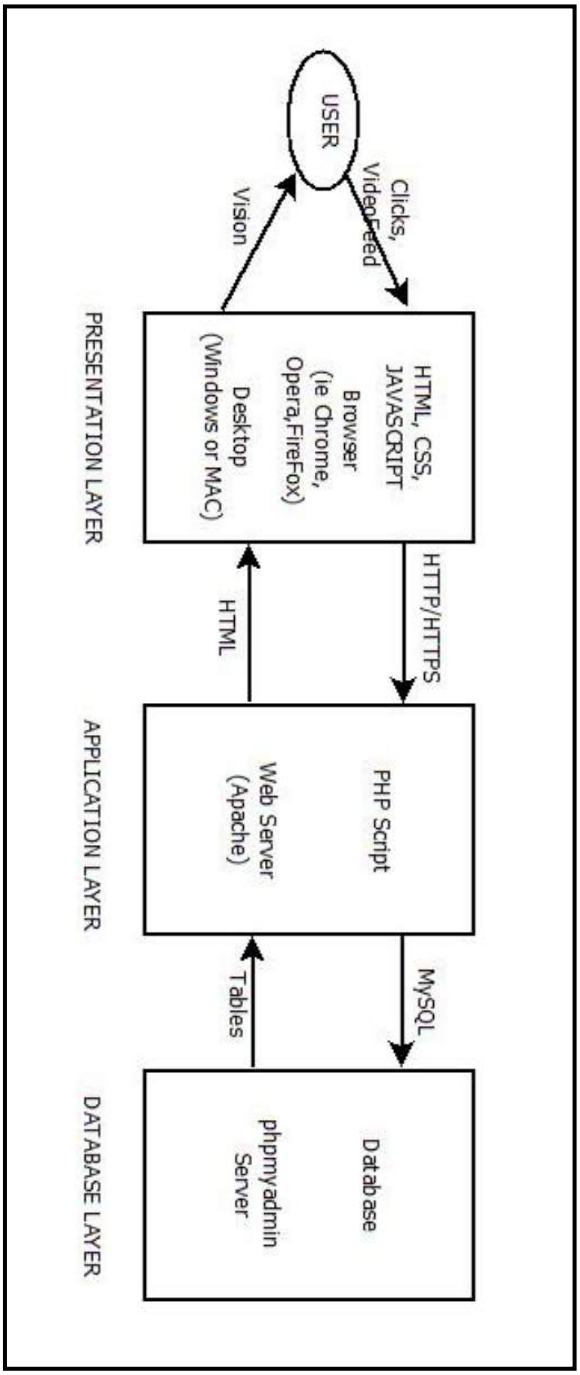
Deployment Diagram:



SOFTWARE ARCHITECTURE DESIGN:

The 3-Client Architecture best suits and satisfies our project. This architecture was chosen because in our project we had to concentrate separately on different aspects of the software more in compared to the functional/component modules of the project.

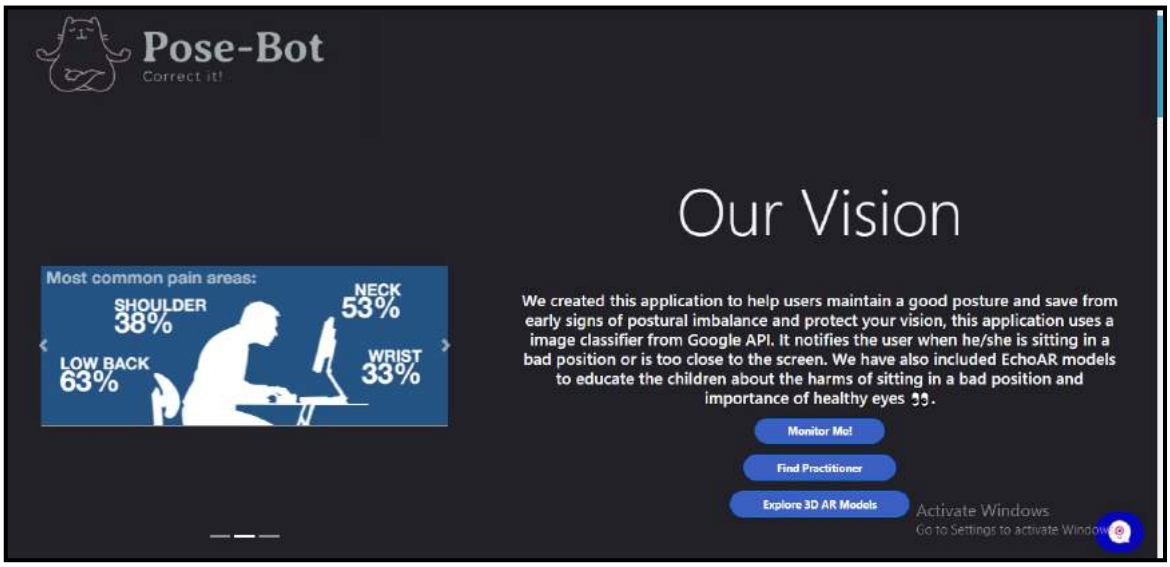
Other than this architecture the Data-Flow Architecture or Component Based Architecture were also considered but the 3-Client Architecture was most suitable.



INTERFACE DESIGN & DESCRIPTION:

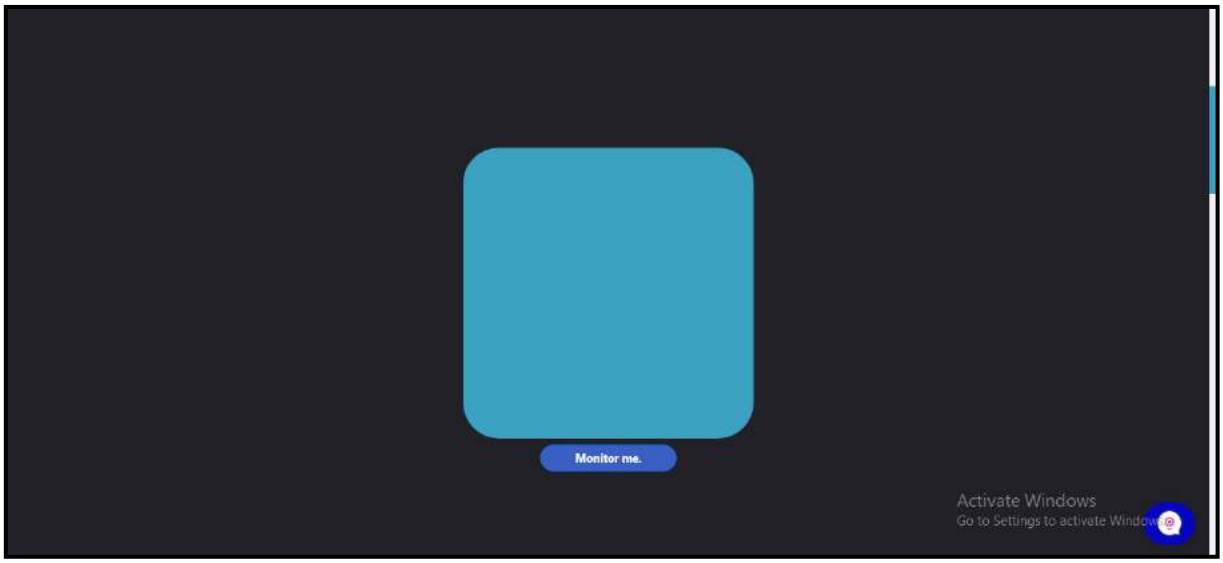
- Home-Page:

This is the dashboard part of the website where a brief introduction and navigation to different sections of the software is provided.



- Monitor Me:

Through this part of the interface the user can grant permission to the software to fetch the user's video feed, and predict the user's posture then recommend corrective measures for the same.



❖ Find Practitioner:

From this section of the software the user can search for practitioners based on either name, specialization or the locality of preference.

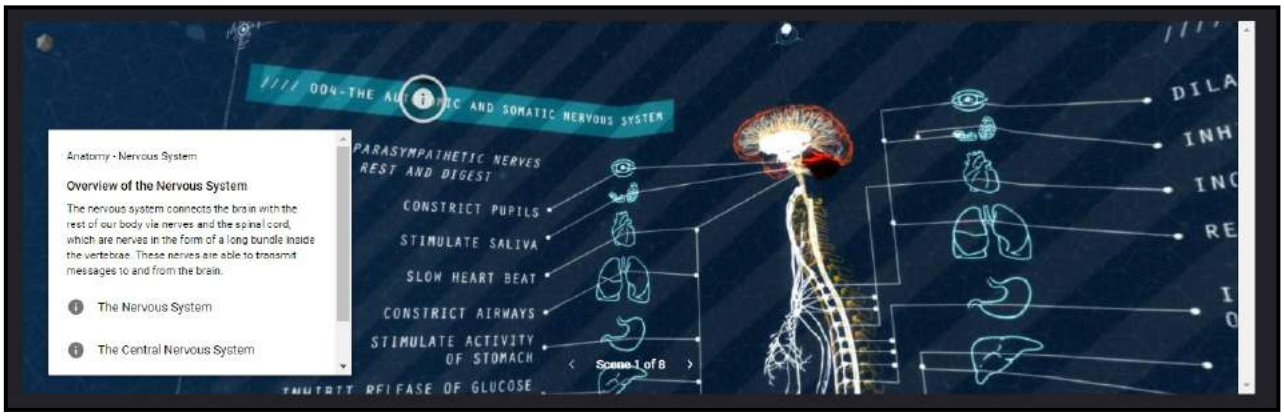
❖ AR Models:

Here is the interface, which is basically the “knowledge centre" of the website. This feature exists to educate a fellow user about the 8 different parts of the body that may be affected due to maintaining a poor posture. The following are the 8 scenes of the AR Models:

✓ Overview of nervous system:

This is the first scene of the AR Model which further has the following sub-scenes:

i. The Nervous System
ii. The Central Nervous System
iii. The Autonomic Nervous Systems



✓ Brain:

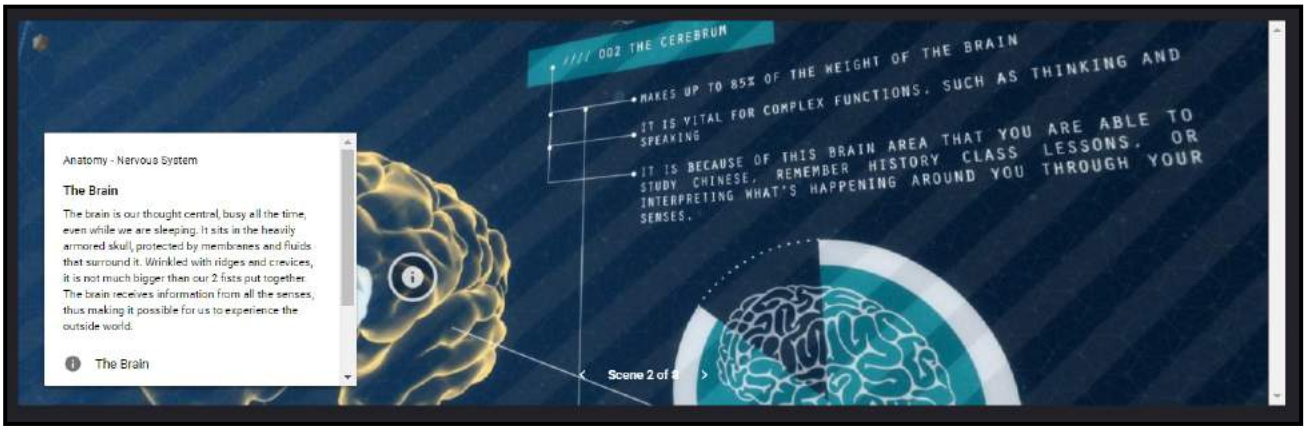
This scene explains the different components of the brain.

The following are its sub-scenes:

i. The Brain

ii. The Cerebrum

iii. The Cerebellum and Brainstem



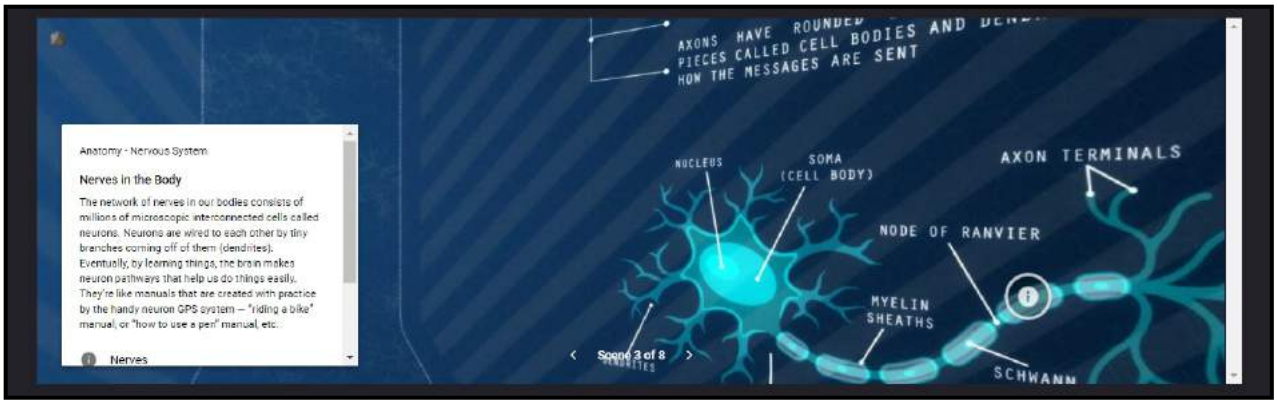
✓ Nervous flow:

This shows the network of nerves in our body, and the following are its sub-scenes:

i. Nerves

ii. Axons

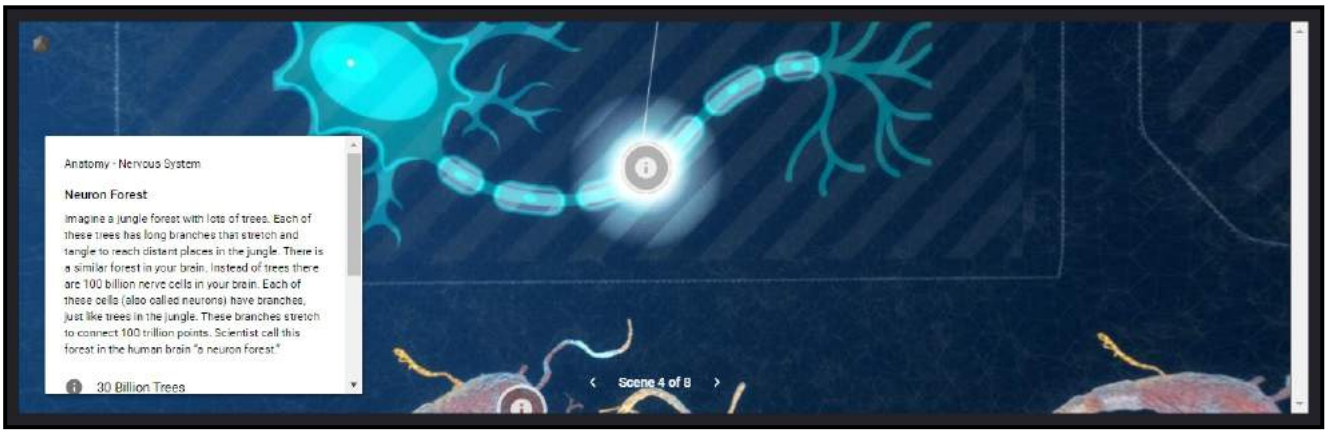
iii. Cell bodies and dendrites



✓ Neuron forest:

There are a millions of neurons in our brain that are connected in the form of trees. This section allows user to learn how these neurons function with each other.

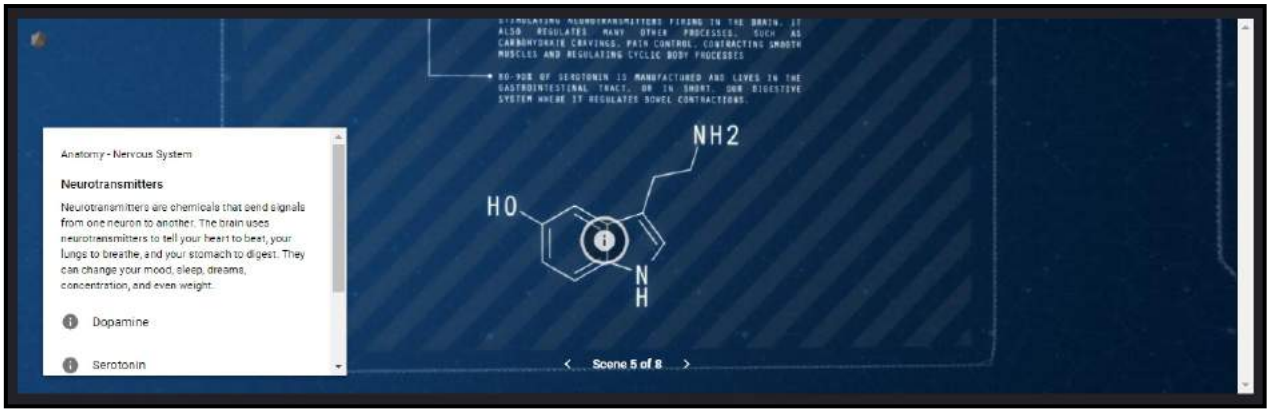
i. 30 Billion Trees
ii. Signals
iii. Synapses
iv. Neurotransmitter Keys
v. Receptor Locks



✓ Neurotransmitters:

Sub-scenes involved:

i. Dopamine
ii. Serotonin
iii. Melatonin
iv. Dimethyltryptamine



- ✓ Cell membrane and ion channels:

  Sub-scenes involved:

  i. Impulses
  ii. Ions
  iii. Membrane Potential
  iv. Polarization
  v. Depolarization

- ✓ Migraine:

  Sub-scenes involved:

  i. Triggers
  ii. Causes
  iii. Mixture of Errors
  iv. Trigeminal Nerve
  v. Eye Symptoms
  vi. Nose Symptoms
  vii. Ear Symptoms

- ✓ Sympathetic vs. Parasympathetic Nervous System:

  Sub-scenes involved:

  i. Sympathetic Nervous System
  ii. Fight or Flight
  iii. Parasympathetic Nervous System

- ❖ Chat-Bot:

Through this interface the user can gain a better introspection of his/her symptoms. The Chat-Bot navigates the user through a set of questionnaire and predicts the seriousness of his/her symptoms. And along with prediction it also suggests helpful measures that can be followed by the user.

SOME INTERFACE DESIGN PRINCIPLES MAINTAINED:

- Anticipation: The Chat-Bot implemented allows a sequential access through the website and smooth walk-through towards symptom estimator.
- Communication: The software is equipped with proper JavaScript Animations to show the loading state of the software.
- Consistency: A proper color contrast/format is maintained over the navigation center to achieve attention of the user.
- Efficiency: The website is equipped with proper shortcuts, that are easy to run so that the user can reach the desired destination/section efficiently.
- Focus: Through the process of posture detection, it is always made sure that the camera used is not being used by any other functionality, hence certifying complete resource allocation to given task at hand.

- Readability: All the information provided in the AR Model section has been drafted in a way that may impart knowledge easily over all different age groups.

RISK ASSESSMENT:

PROJECT RISK ASSESSMENT

Project Name: POSBOT	**Project Size: Medium**
Project Manager: SAI CHAITANYA	**Date: 19/05/2021**
Risk Score: 265	**Risk Rating: 115 to 290: MEDIUM RISK**

Project Risk Score & Risk Rating Guidelines
0 to 110: LOW RISK
115 to 290: MEDIUM RISK
300 or higher: HIGH RISK

#	Risk Category	Score	Comments
	BUSINESS RISK	**55**	**Overall business risk is medium**
1	**Implementation complexity and impact to processes and business areas:**	**20**	
	- Implementation is not complex	0	Implementation of creating an API to fetch the model requests makes our system more complex.
	- Implementation is complex, though team has prior experience	20	
	- Implementation is complex with many areas impacted and new architectures introduced	0	
2	**Business Dependencies that might impact implementation:**	**25**	
	- No impact, No business dependencies exist	0	The current business model gives the user ability to find the posture but cannot predict future behaviors
	- Yes, some business dependencies exist but are low risk	25	
	- Yes, there are key business dependencies that might impact project implementation	0	
3	**Outages required during the implementation:**	**10**	
	- None required	10	
	- Outages to be planned after business hours	0	
	- Outages to be planned during business hours	0	
	TECHNOLOGY RISK	**180**	**Overall technology risk is upper bound of Medium severity**
4	**Type of Project:**	**80**	
	- Maintenance (e.g., correct problems)	5	There are two major issues in the type of project, as the technology gets more advances our product would require new data about the practitioners to be inputted.These details must be updated on a daily basis, to keep the project running. The current model uses sequential neural network for training but when in due course a new and more efficient
	- Enhancement (e.g., add new features)	5	
	- Product introduction	5	
	- Process improvement/re-engineering	0	
	- Upgrade existing software or applications	25	
	- New development – replace existing system	0	
	- New development – develop system to support new business	0	

	- Implementation of software package in-house	0	architecture is formulated we would have to create a new system to fetch the prediction data.
	- Outsourcing to external vendor	40	
5	**External Vendor Involvement:**	**45**	
	- No external or internal vendor required	0	The deployment services needs to be outsourced to a cloud service provider to host our Deployment model easily and outsource the service of providing posture database to a third party API to gain more dynamic data.
	- No, an internal partner will be used for some project activities	0	
	- Yes, an existing external vendor will be used	15	
	- Yes, a known external vendor will be used but a new contract or PO is required	30	
	- Yes, and external vendor is required but not yet identified	0	
6	**Level of vendor support necessary for the technology after implementation:**	**30**	
	- N/A (no vendors involved)	0	Maintaining the deployment services and API calls need external support
	- No subsequent vendor support necessary	0	
	- Minor support necessary (e.g., maintenance only)	30	
	- Major support necessary (e.g., programming, upgrades, etc.)	0	
7	**Newness of Technology (hardware, systems, databases, communication, etc.) used in the project:**	**10**	
	- N/A (no technology involved)	0	
	- Technology is not new and has been proven or tested	10	
	- New to the business line	0	
	- New to University Services	0	
	- New to the education/learning services industry	0	
8	**Availability of Project Team Resources**	**15**	
	- We have the resources with the appropriate skill sets	5	
	- We have resources but they will need to be trained	10	
	- The project will require external resources	0	
	IMPLEMENTATION RISK	**30**	**N/A**
9	**Will the Office for General Counsel or Purchasing be used during the negotiation, creation, or modification of the contract?**	**10**	
	- No contract is required for project.	0	
	- No, the existing contract will be used and no modifications or addendum will be required.	10	
	- Yes, the OGC and Purchasing will be highly involved with contract negotiation, creation, or modification.	0	
	- Yes, the OGC and Purchasing will be involved, but in a limited capacity	0	
	- No, the OGC and Purchasing will not involved with the contract negotiation, creation, or modification.	0	
10	**Has the Business Case been developed, reviewed and approved:**	**20**	
	- Yes, the Business Case/justification is documented and fully approved	0	The business assumptions are modulated and discussed. The stake holders are estimated, although the service deliverables has some risk in case of generalizing the prediction results for almost every type of posture as the current business model can only promote identification of good posture.
	- Yes, but there are some unknowns or assumptions the business case, stakeholders or product/service deliverables	20	
	- No, there are many unknowns or assumptions around the business case, stakeholders, strategy or product/service deliverables.	0	
11	**Additional project risks not previously addressed [1]:**	**0**	
	-	0	
	-	0	

	-	0	
	TOTAL PROJECT RISK SCORE	**265**	**FINAL PROJECT RISK RATING: MEDIUM RISK**

Project Risk Score & Risk Rating Guidelines
0 to 110: LOW RISK
115 to 290: MEDIUM RISK
300 or higher: HIGH RISK

Risk Assessment Template - Top Five Risks

#	Risk statement		(Scale 1-100%)	(Scale 1-10)	D*E/100	Mitigation	Contingency	Triggers	Assignee
	Condition	**Consequence**	**Probability**	**Impact**	**Exposure**				
1	The current business model gives the user ability to predict diseases related to only tomato plants	Affects the business model significantly, can become difficult to market the product	80%	7	5.6	Search for more generalized dataset for training the deep learning model	Prioritize the training of the model and improving the accuracy of the model and making it viable for practical use	When we have to finalize the posture categories	Project manager
2	As the technology gets more advances our product would require new architecture or tech stack for the application to be implemented	The current model uses CNN architecture for training but when in due course a new and more efficient architecture is formulated we would have to create a new system to fetch the prediction data from firebase.	60%	8	4.8	Keeping updates of new architecture	Develop, test, and regularly review recovery procedures for the model training and architecture building.	Start of project	Development

3	As the complexity of the project increases.	Reduced ability of the system admin to access and manipulate the data as required by the product.	50%	6	3	If necessary, develop scripts to standardize the format of data to the service provider.	It might be possible to develop code to access the nonstandard-format data. If not, a migration sub-project will be necessary.	When fining practitioner stats functionality using dynamic location based search	Product manager
4	Practitioner data might require conversion. None is currently planned for.	Reduced ability of the system to access and manipulate and display the data as required.	40%	8	3.2	Determine extent of problem. If necessary, develop scripts to standardize the format of data.	It might be possible to develop code to access the nonstandard-format data.	When developing database architecture	Product manager
5	The deployment services needs to be outsourced to a cloud service provider to host our Deployment model easily	If the deployment collapses, the complete product architecture will be in risk.	50%	10	5	Planning a deployment architecture before hand and test it	Since, the model is huge it needs a external deployment service.	After creating the model	Product manager

MANUAL TESTING:

Test Scenario ID	Monitor Me	**Test Case ID**	Monitor Me - A
Test Case Description	Monitor Me - No Face Found	**Test Priority**	Medium
Pre-Requisite	NA	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The posture detection feature is activated	'Moniter Me' selected	User Video Feed	User Video Feed	Google Chrome	Pass

3.	No person is sitting in front of webcam	User Video Feed	No Outputs Given	No Outputs Given	Google Chrome	Pass

Test Scenario ID	Monitor Me	**Test Case ID**	Monitor Me - B1
Test Case Description	Monitor Me - Face Found : Good Posture	**Test Priority**	High
Pre-Requisite	Normal Human Being	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The posture detection feature is activated	'Moniter Me' selected	User Video Feed	User Video Feed	Google Chrome	Pass
3.	Person sitting in front of camera maintains good posture	User Video Feed	No Outputs Given	No Outputs Given	Google Chrome	Pass

Test Scenario ID	Monitor Me	**Test Case ID**	Monitor Me - B2
Test Case Description	Monitor Me - Face Found : Bad Posture	**Test Priority**	High
Pre-Requisite	Normal Human Being	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result

1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The posture detection feature is activated	'Moniter Me' selected	User Video Feed	User Video Feed	Google Chrome	Pass
3.	Person sitting in front of camera is slouching	User Video Feed	"You are slouching. Please correct your posture"	"You are slouching. Please correct your posture"	Google Chrome	Pass

Test Scenario ID	Monitor Me	**Test Case ID**	Monitor Me - B3
Test Case Description	Monitor Me - Face Found : Very Close to Webcam	**Test Priority**	High
Pre-Requisite	Normal Human Being	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The posture detection feature is activated	'Moniter Me' selected	User Video Feed	User Video Feed	Google Chrome	Pass
3.	Person sitting in front of camera is slouching	User Video Feed	"You are slouching. Please correct your posture"	"You are slouching. Please correct your posture"	Google Chrome	Pass

Test Scenario ID	Practitioner Search	**Test Case ID**	Practitioner Search - A
Test Case Description	Practitioner Search - Invalid Practitioner Details	**Test Priority**	Medium
Pre-Requisite	NA	**Post-Requisite**	NA

Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The practitioner search feature is activated	'Find Practitioner' selected	Search Forum	Search Forum	Google Chrome	Pass
3.	Select parameters to search	'Name'	Search by Name Forum	Search by Name Forum	Google Chrome	Pass
4.	Enter Invalid Details	'Doctor Strange'	No Details Found	No Details Found	Google Chrome	Pass

Test Scenario ID	Practitioner Search	**Test Case ID**	Practitioner Search - B1
Test Case Description	Practitioner Search - Practitioner Details : by Name	**Test Priority**	High
Pre-Requisite	Practitioner must exist	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The practitioner search feature is activated	'Find Practitioner' selected	Search Forum	Search Forum	Google Chrome	Pass

3.	Select parameters to search	'Name'	Search by Name Forum	Search by Name Forum	Google Chrome	Pass
4.	Enter Invalid Details	'Rakesh Shrama'	Top Matches of 'Rakesh Shrama'	Top Matches of 'Rakesh Shrama'	Google Chrome	Pass

Test Scenario ID	Practitioner Search	**Test Case ID**	Practitioner Search - B2
Test Case Description	Practitioner Search - Practitioner Details : by Specialization	**Test Priority**	High
Pre-Requisite	Practitioner must exist	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The practitioner search feature is activated	'Find Practitioner' selected	Search Forum	Search Forum	Google Chrome	Pass
3.	Select parameters to search	'Specialization'	Search by Specialization Forum	Search by Specialization Forum	Google Chrome	Pass
4.	Enter Invalid Details	'Surgeon'	Top Matches of 'Surgeon'	Top Matches of 'Surgeon'	Google Chrome	Pass

Test Scenario ID	Practitioner Search	**Test Case ID**	Practitioner Search - B3
Test Case Description	Practitioner Search - Practitioner Details : by Location	**Test Priority**	High
Designed By: Chaitanya	**Designed Date:**	**Executed By:** Soham &	**Execution Date:**

	22/05/2021	Nadish	22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The practitioner search feature is activated	'Find Practitioner' selected	Search Forum	Search Forum	Google Chrome	Pass
3.	Select parameters to search	'Location'	Search by Location Forum	Search by Location Forum	Google Chrome	Pass
4.	Enter Invalid Details	Street:Sector90 City: Gurgaon State: Haryana Country: India	Top Matches found	Top Matches found	Google Chrome	Pass

Test Scenario ID	AR Models	**Test Case ID**	AR Models
Test Case Description	AR Model - finding specific information centre	**Test Priority**	Low
Pre-Requisite	NA	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass

2.	The AR model feature is activated	'AR Models' selected	Scene Select Section	Scene Select Section	Google Chrome	Pass
3.	Selecting scene of the human body	'The Brain'	Sub-Scene Select Option	Sub-Scene Select Option	Google Chrome	Pass
4.	Selecting sub-scene	'Cerebrum'	Information Gain using AR model of 'Cerebrum'	Information Gain using AR model of 'Cerebrum'	Google Chrome	Pass

Test Scenario ID	Chat-Bot	**Test Case ID**	Chat-Bot - A
Test Case Description	Chat-Bot - Know More	**Test Priority**	Low
Pre-Requisite	NA	**Post-Requisite**	NA
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The Chat-Bot feature is activated	'Chat-Bot' selected	Chat-Bot Window	Chat-Bot Window	Google Chrome	Pass
3.	Selecting 'know more'	'Know More'	Display Information about Website	Display Information about Website	Google Chrome	Pass

Test Scenario ID	Chat-Bot	**Test Case ID**	Chat-Bot - B
Test Case Description	Chat-Bot - Predict Solution	**Test Priority**	Medium
Pre-Requisite	NA	**Post-Requisite**	NA

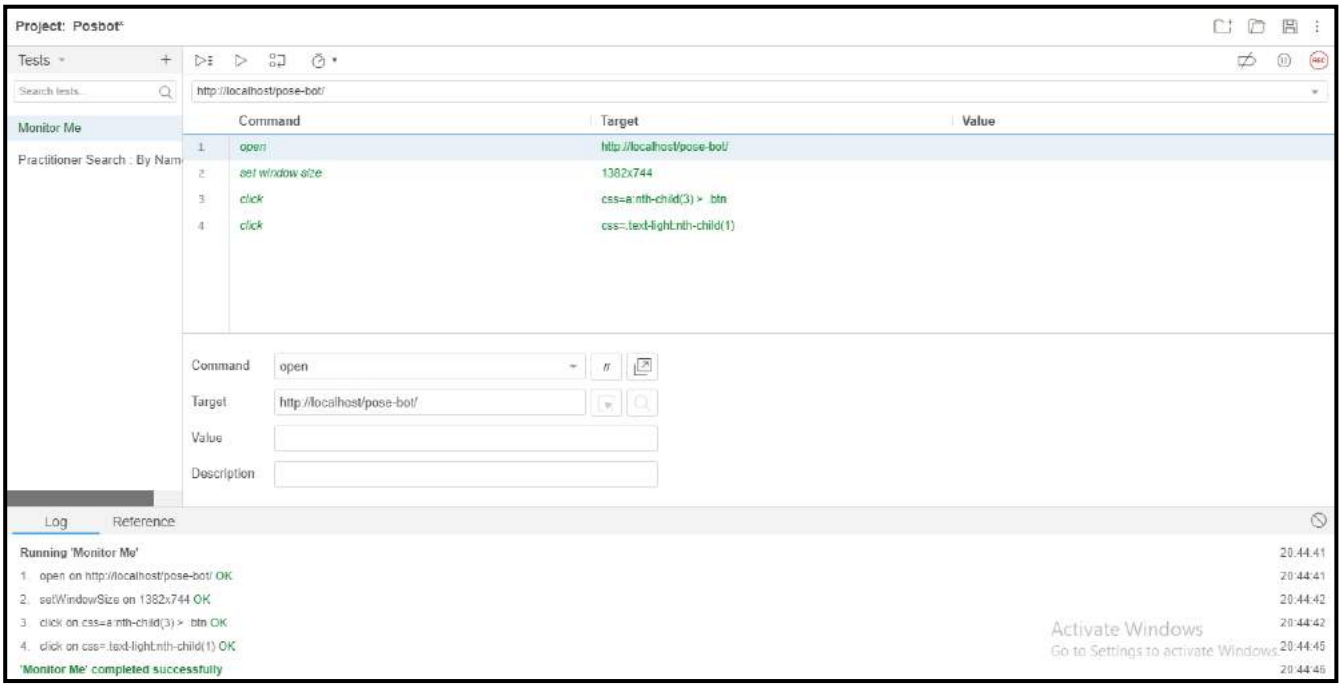
Designed By: Chaitanya	**Designed Date:** 22/05/2021	**Executed By:** Soham & Nadish	**Execution Date:** 22/05/2021

Test Execution Steps:

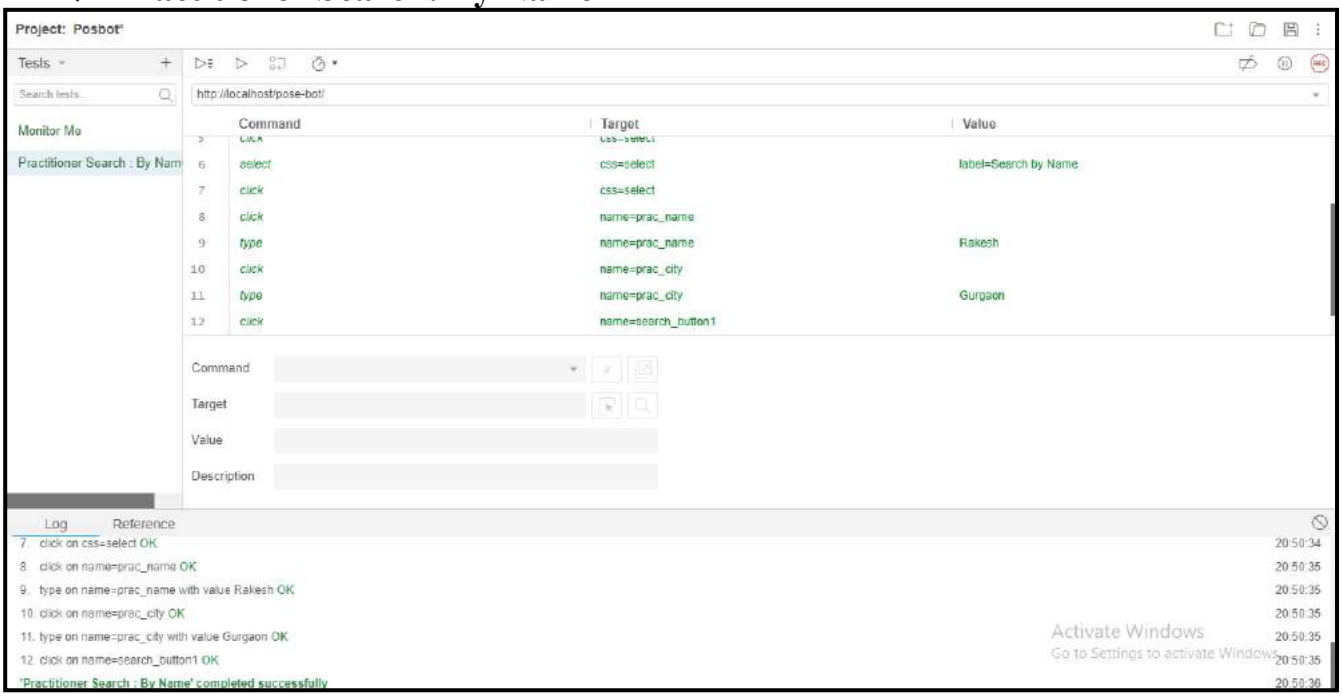
S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result
1.	Launch Application	'POSBOT' URL in browser search	'POSBOT' Dashboard	'POSBOT' Dashboard	Google Chrome	Pass
2.	The Chat-Bot feature is activated	'Chat-Bot' selected	Chat-Bot Window	Chat-Bot Window	Google Chrome	Pass
3.	Selecting option to find solutions	'Find Solution for my Symptoms'	Display Symptom questionnaire	Display Symptom questionnaire	Google Chrome	Pass
4.	Answer the questionnaire	Answer Radio-Based questionnaire	Display Solutions	Display Solutions	Google Chrome	Pass

AUTOMATED TESTING:

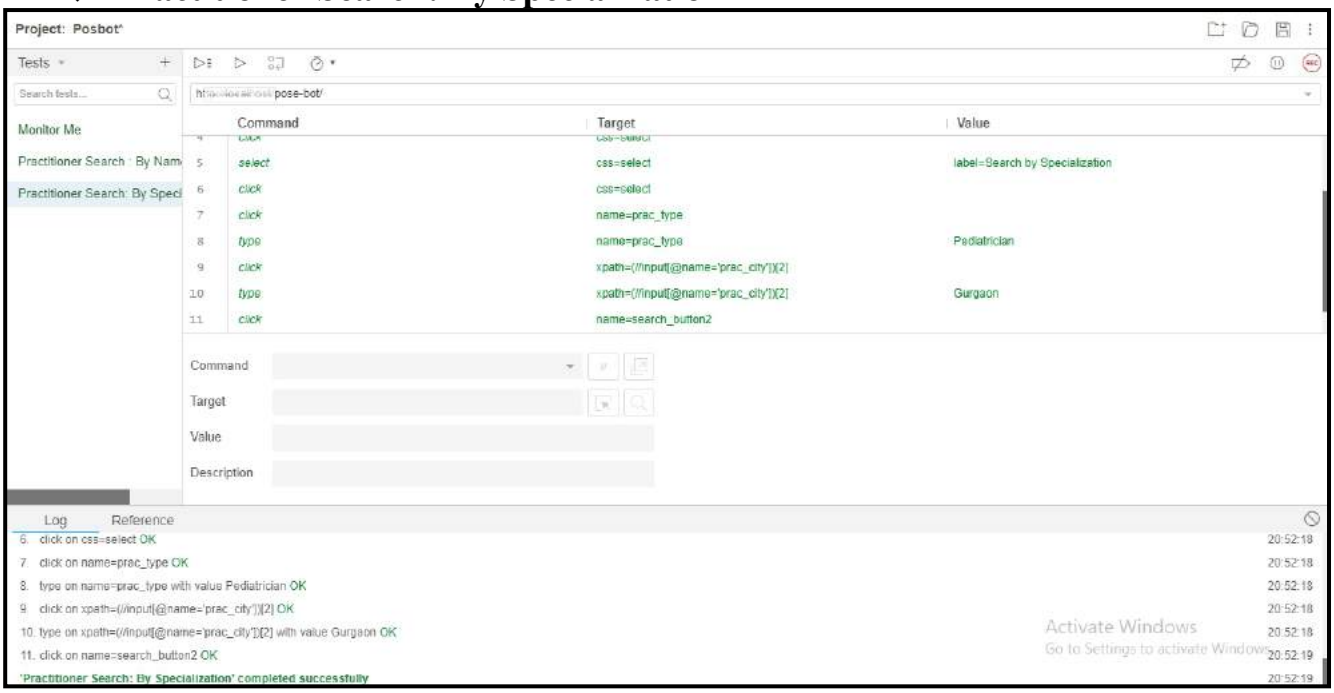
- **Monitor Me**



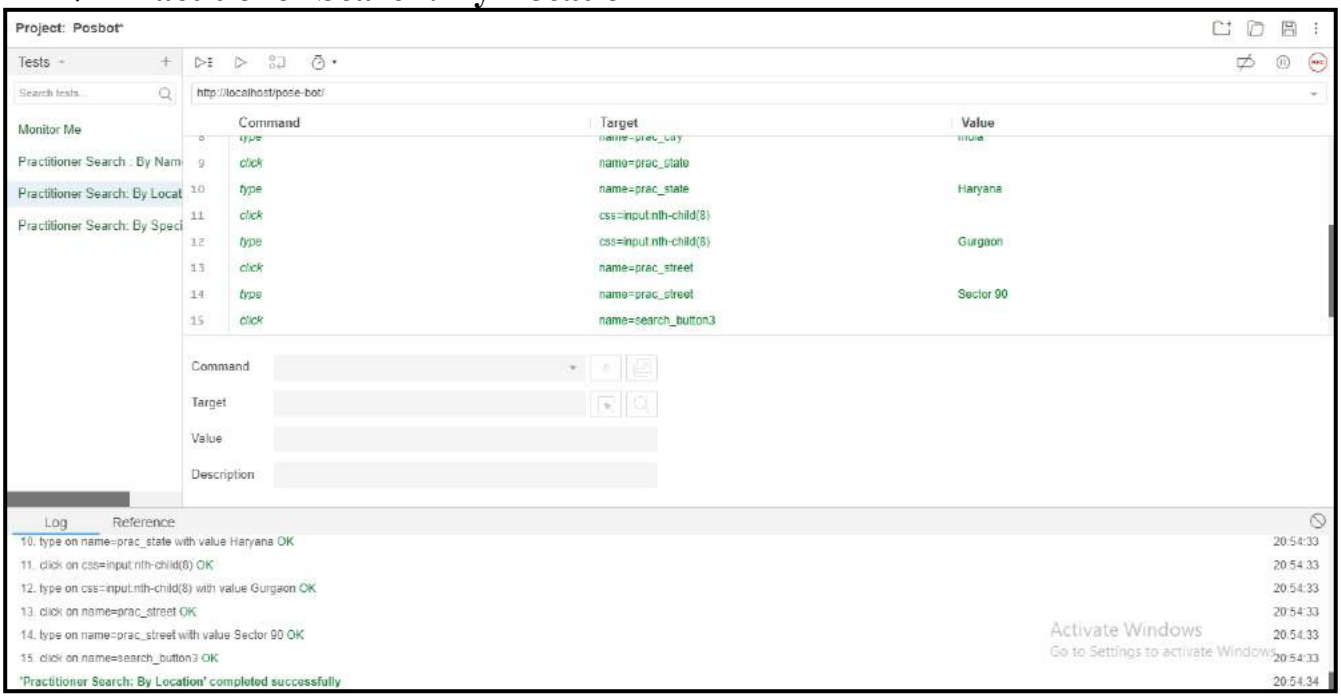
- **Practitioner Search: By Name**



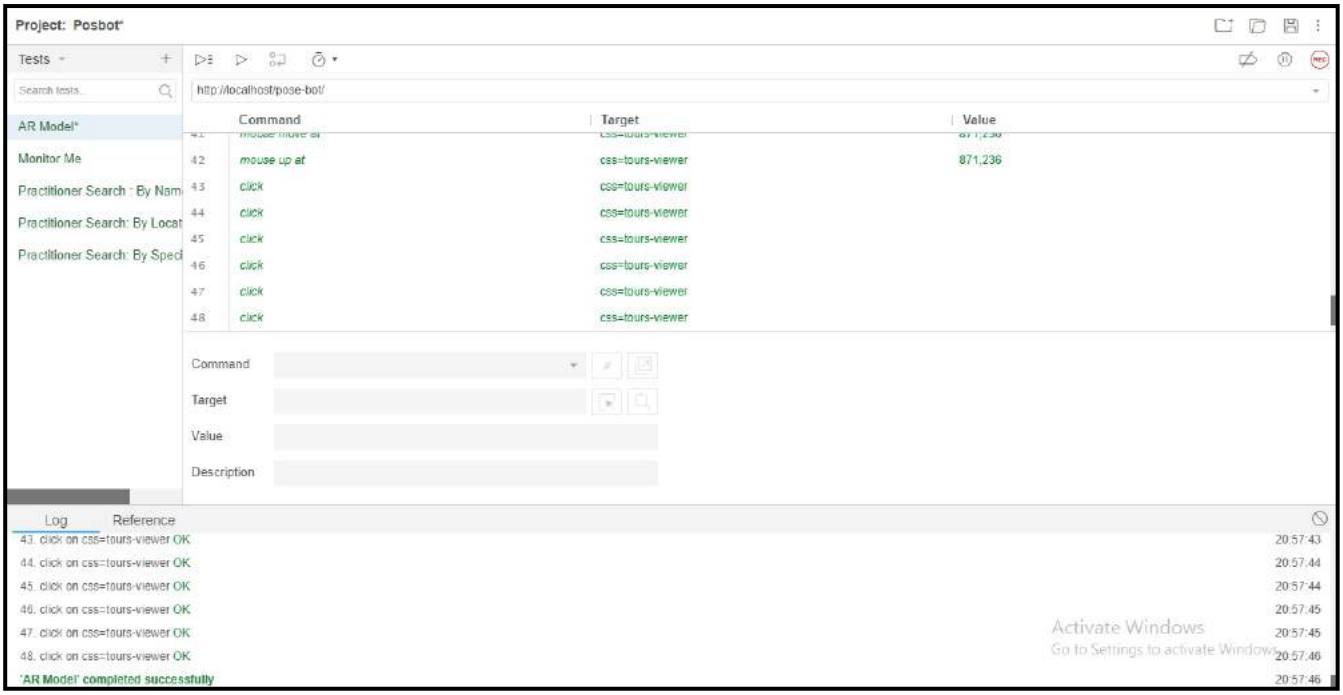
- **Practitioner Search: By Specialization**



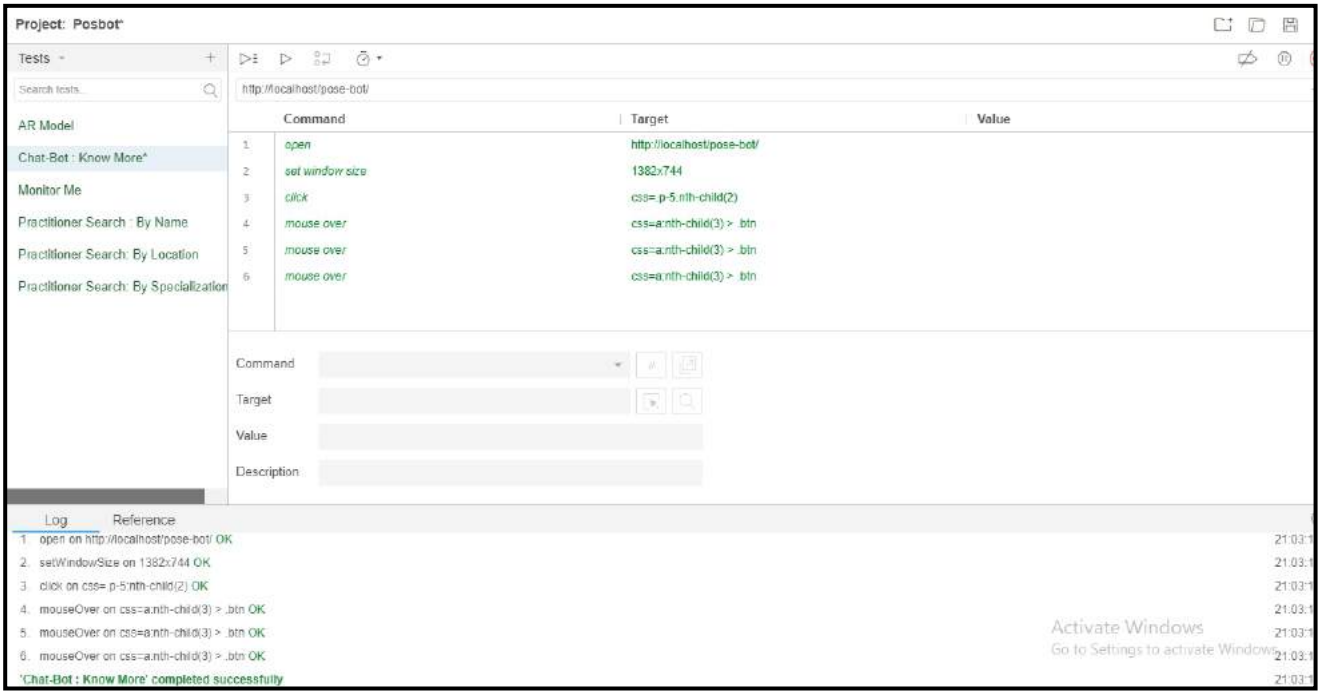
◆ Practitioner Search: By Location



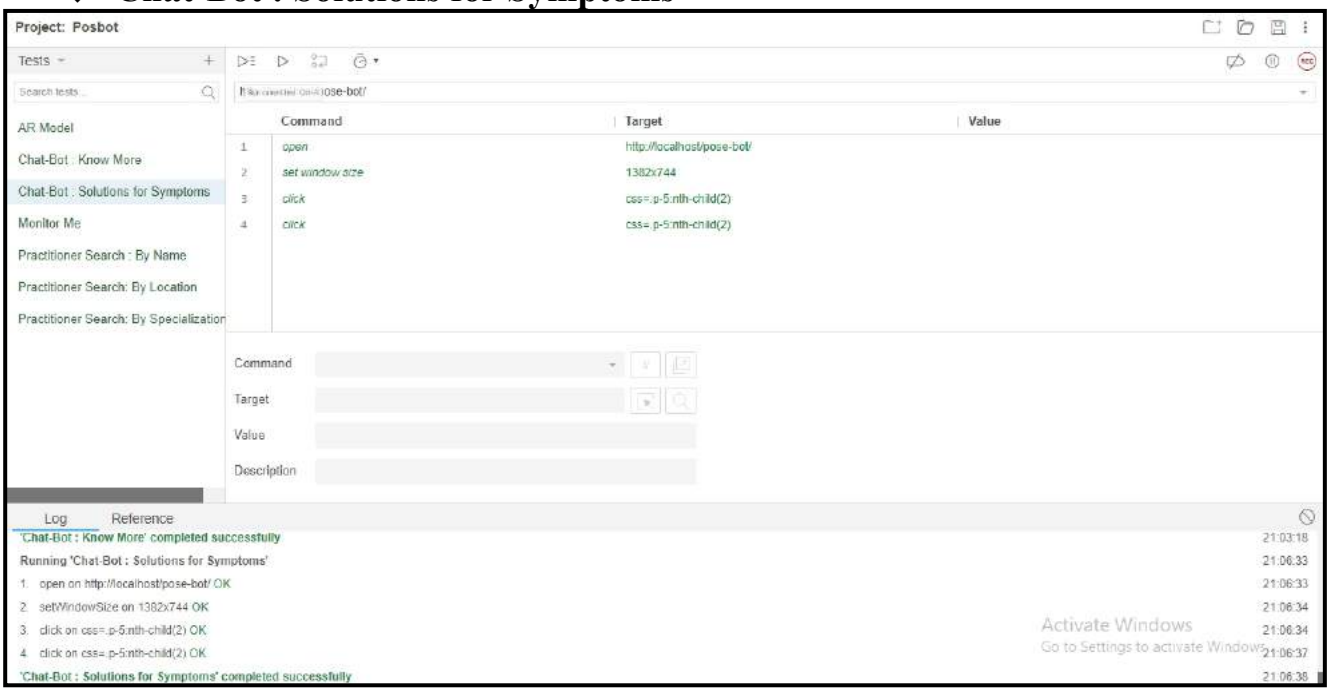
◆ AR Model



◆ Chat-Bot : Know More



◆ Chat-Bot : Solutions for Symptoms



IX. COMPARITIVE STUDY WITH EXISTING TECHNOLOGIES

The projects ‘Posture-Corrector-Bot’ and ‘Posture-Checker’ are en user interactive bots made in python. This used other modules like openCV for face and posture detection. The classification models developed were from scratch using complete raw data. This was then imported as a discord bot using discord.py python modules. As these were specifically created as discord bots, these features cannot run independently as a separate app or website.

Similarly in the project ‘NagBot’ also being a discord bot, yet provides more number of features. Along with the posture correction they put forward a reminder app that, reminds the user periodically to drink water. This allows the user to remain hydrated throughout the day.

‘Posture-Watch’ is yet another attempt towards the same objective. This project is also independent of relying on third party domains like discord or telegram. This a user can run anywhere from his/her local machine. Although the drawback of this is that the python script should always be open while working. And as this project includes continuous query transfers in form of image frames it might make the user’s workstation much slower and might even hinder with the performance of other applications on the workstation.

Hence in our project we have created a website that implements this feature. Moreover, instead of running a convolutional classification neural network from raw data, we have trained our model over Echo AR pre-processed data, hence POSBOT guarantees much higher accuracies. As this is hosted over the internet the user no longer need to worry about the this product hindering with the performance of other applications on his/her workstation, all the processing is taken care by the server.

X. CONCLUSION & FUTURE WORKS

This product proves to be an excellent way for different users to correct their postures in the early ages, rather than facing severe symptoms in the future. Moreover the information center(AR Models) are also attractive & interactive manner for school children to be hooked up with this website. In this way they can be educated of the benefits of maintaining proper postures and other good habits that they can learn, that will in turn help them remain healthy and strong for a long time.

We even plan to extend this project in the future by creating a chrome extension that can be more handy for the users and much more easier to use. Our project works on the basis of how accurate the classification model is, hence the dataset can be increased with more number of

categories to be classified into, so that the boundaries of the classification model can be extended and more number of problems can be identified by just the user's live video feed.

XI. REFERENCES

i. https://www.healthline.com/health/fitness-exercise/posture-benefits#3.-Increased-energy-levels-
ii. https://github.com/saubury/posture-watch
iii. https://github.com/minhduccao/NagBot
iv. http://doc.aldebaran.com/1-14/naoqi/motion/alrobotposture.html
v. https://www.hackster.io/justin-shenk/posture-pal-with-walabot-e2bf3d
vi. https://towardsdatascience.com/real-time-head-pose-estimation-in-python-e52db1bc606a

XII. APPENDIX

- Link to PPT : https://docs.google.com/presentation/d/1sqUo44RG09_mNsq3_V1xibve5l6g0tpgarUDnqKLl-U/edit?usp=sharing

- Link to Demo Video : https://drive.google.com/file/d/1fapSXi8q5vmsSFWVAJB1Nu6wnlH7m4SZ/view?usp=sharing

- Link to Source Code and Setup Instructions : https://github.com/saivit11/POSBOT