A GESTURE- BASED TOOL FOR STERILE BROWSING OF RADIOLOGY IMAGES

(TEAM ID : PNT2022TMID34850)

PROJECT REPORT
Submitted by

MANIKANDAN.P - 510419104044

BALAGURU.S - 510419104017

ARAVIND.S – 510419104010

DHARANI.E - 510419104022

BACHELOR OF ENGINEERING

COMPUTER SCIENCE AND ENGINEERING

ARUNAI ENGINEERING COLLEGE, TIRUVANNAMALAI.

ANNA UNIVERSITY: CHENNAI 600 025

INDEX

CHAPTER 1

INTRODUCTION

Computer information technology is increasingly penetrating into the hospital domain. A major challenge involved in this process is to provide doctors with efficient, intuitive, accurate and safe means of interaction without affecting the quality of their work. Keyboards and pointing devices, such as a mouse, are today's principal method of human—computer interaction. However, the use of computer keyboards and mice by doctors and nurses in intensive care units (ICUs) is a common method for spreading infections. In this project, we suggest the use of hand gestures as an alternative to existing interface techniques, offering the major advantage of sterility. Even though voice control also provides sterility, the noise level in the operating room (OR) deems it problematic.

In this project, we refer to gestures as a basic form of non-verbal communication made with the hands. Psychological studies showed that young children use gestures to communicate before they learn to talk. Manipulation, as a form of gesticulation, is often used when people speak to each other about some object. Naturalness of expression, non-encumbered interaction, intuitiveness and high sterility are all good reasons to replace the current interface technology (e.g., keyboard, mouse, and joystick) with more natural interfaces.

1.1 Project Overview

Humans are able to recognize body and sign language easily. This is possible due to the combination of vision and synaptic interactions that were formed along brain development . In order to replicate this skill in computers, some problems need to be solved: how to separate objects of interest in images and which image capture technology and classification technique are more appropriate, among others.

In this project Gesture based Desktop automation, First the model is trained pre trained on the images of different hand gestures, such as a showing numbers with fingers as 1, 2, 3, 4 . This model uses the integrated webcam to capture the video frame. The image of the gesture captured in the video frame is compared with the Pre-trained model and the gesture is identified. If the gesture predicts is 1 then images is blurred; 2 image is resized; 3 image is rotated etc.

1.2 Purpose

Purpose of the project is used to browse the radiology images using hand gestures rather than using mouse, keyboard etc, to maintaining sterility inside the operation room.

CHAPTER 2

LITERATURE SURVEY

1. The recently developed depth sensors, e.g., the Kinect sensor, have provided new opportunities for human-computer interaction (HCI). Although great progress has been made by leveraging the Kinect sensor, e.g., in human body tracking, face recognition and human action recognition, robust hand gesture recognition remains an open problem. Compared to the entire human body, the hand is a smaller object with more complex articulations and more easily affected by segmentation errors. It is thus a very challenging problem to recognize hand gestures. This paper focuses on building a robust part-based hand gesture recognition system using Kinect sensor. To handle the noisy hand shapes obtained from the Kinect sensor, we propose a novel distance metric, Finger-Earth Mover's Distance (FEMD), to measure the dissimilarity between hand shapes. As it only matches the finger parts while not the whole hand, it can better distinguish the hand gestures of slight differences. The extensive experiments demonstrate that our hand gesture recognition system is accurate (a 93.2% mean accuracy on a challenging 10gesture dataset), efficient (average 0.0750 s per frame), robust to hand articulations, distortions and orientation or scale changes, and can work in uncontrolled environments (cluttered backgrounds and lighting conditions). The superiority of our system is further demonstrated in two real-life HCI applications.

2. In this paper, we address natural human-robot interaction (HRI) in a smart assisted living (SAIL) system for the elderly and the disabled. Two common HRI problems are studied: hand gesture recognition and daily activity recognition. For hand gesture recognition, we implemented a neural network for gesture spotting and a hierarchical hidden Markov model for context-based recognition. For daily activity recognition, a multi sensor fusion scheme is developed to process motion data collected from the foot and the waist of a human subject. Experiments using a prototype wearable sensor system show the effectiveness and accuracy of our algorithms.

3. The use of doctor-computer interaction devices in the operation room (OR) requires new modalities that support medical imaging manipulation while allowing doctors' hands to remain sterile, supporting their focus of attention, and providing fast response times. This paper presents "*Gestix*," a vision-based hand gesture capture and recognition system that interprets in real-time the user's gestures for navigation and manipulation of images in an electronic medical record (EMR) database. Navigation and other gestures are translated to commands based on their temporal trajectories, through video capture. "*Gestix*" was tested during a brain biopsy procedure. In the in vivo experiment, this interface prevented

   the surgeon's focus shift and change of location while achieving a rapid intuitive reaction and easy interaction. Data from two usability tests provide insights and implications regarding human-computer interaction based on nonverbal conversational modalities.

4. They have developed a system that uses computer vision to replace standard computer mouse functions with hand gestures. The system is designed to enable non-contact human-computer interaction (HCI), so that surgeons will be able to make more effective use of computers during surgery. In this paper, They begin by discussing the need for non-contact computer interfaces in the operating room. They then describe the design of our non-contact mouse system, focusing on the techniques used for hand detection, tracking, and gesture recognition. Finally, They present preliminary results from testing and planned future work.

5. A gesture interface is developed for users, such as doctors/surgeons, to browse medical images in a sterile medical environment. A vision-based gesture capture system interprets user's gestures in real-time to manipulate objects in an image visualization environment. A color distribution model of the gamut of colors of the users hand or glove is built at the start of each session resulting in an independent system. The gesture system relies on real-time robust tracking of the user's hand based on a color-motion fusion model, in which the relative weight applied to the motion and color cues are adaptively determined according to the state of the system. Dynamic navigation gestures are translated to commands based on their relative positions on the screen. A state machine switches between other gestures such as zoom and rotate, as well as a sleep state. Performance evaluation included gesture recognition accuracy, task learning, and rotation accuracy. Fast task learning rates were found with convergence after ten trials. A beta test of a system prototype was conducted during a live brain biopsy operation, where neurosurgeons were able to browse through MRI images of the patient's brain using the sterile hand gesture interface. The surgeons indicated the system was easy to use and fast with high overall satisfaction.

2.1 Existing problem

When the doctors inside the operation room use keyboard and mouse, there is a possibility of spreading infection. So, in this project we used hand gesture methodology to keep the doctor sterile.

2.2 References

1. A Fast Gesture Recognition Scheme for Real-Time HumanMachine Interaction Systems . Ching-Hao Lai* Smart Network System Institute Institute for Information Industry Taipei City, Taiwan , 2010.

2. Real-Time Hand Gesture Recognition System for Daily Information Retrieval fromInternet 1633 IEEE Fourth International Conference on Ubi-Media Computing.

3. Wearable Sensor-Based Hand Gesture and Daily Activity Recognition for Robot-Assisted Living IEEE Transactions on systems, Man , and Cybernetics Part A: Systems and humans, Vol .41, No. 3, May 2011.

4. A Hand Gesture Based Interactive Presentation System Utilizing Heterogeneous Cameras Bobo Zeng, Guijin Wang**, Xinggang Lin.

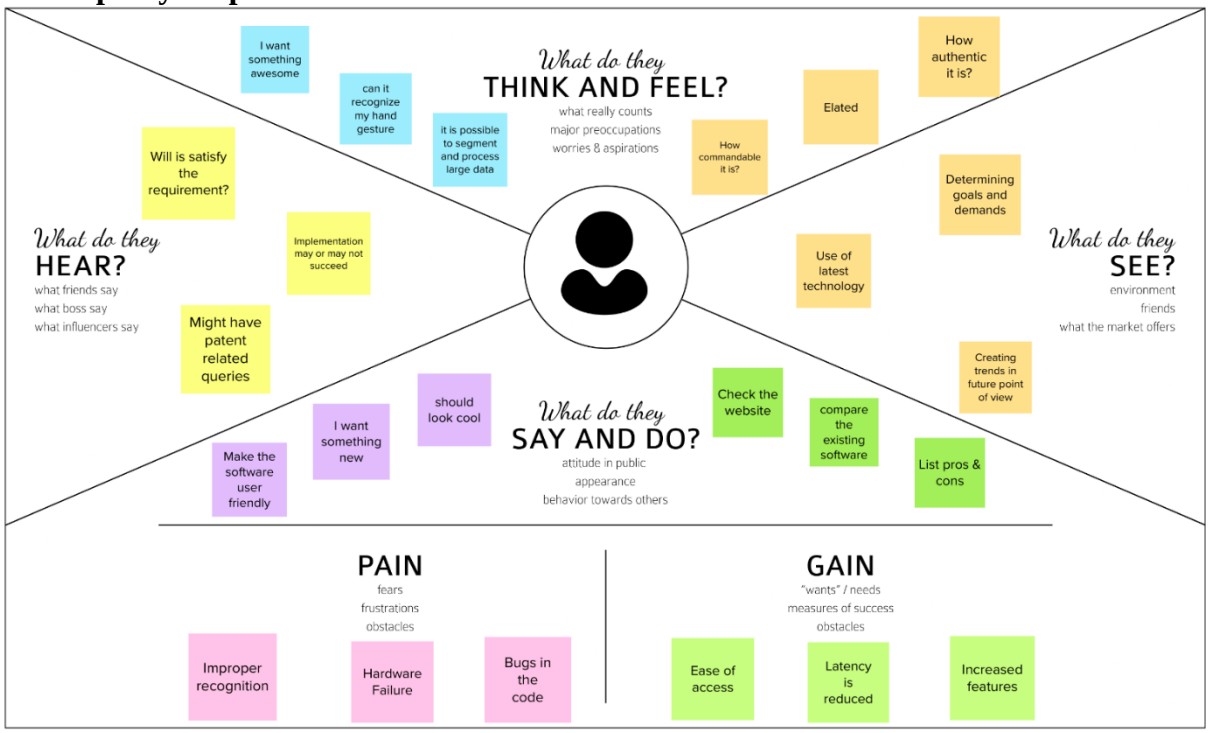
2.3 Problem Statement Definition

Problem Statement (PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
PS-1	Deaf and dumb person	To overcome my disabilities	Difficult to establish communication with others	To convey my message to a normal people is always a challenging task for me	Sad and frustrated
PS-2	Driver and passenger	To interact with vehicles using hand gestures	Unexpected input recognition and behaviour	Drivers don't need to take attention off from the road	Feel like the passenger and driver are safe

CHAPTER 3

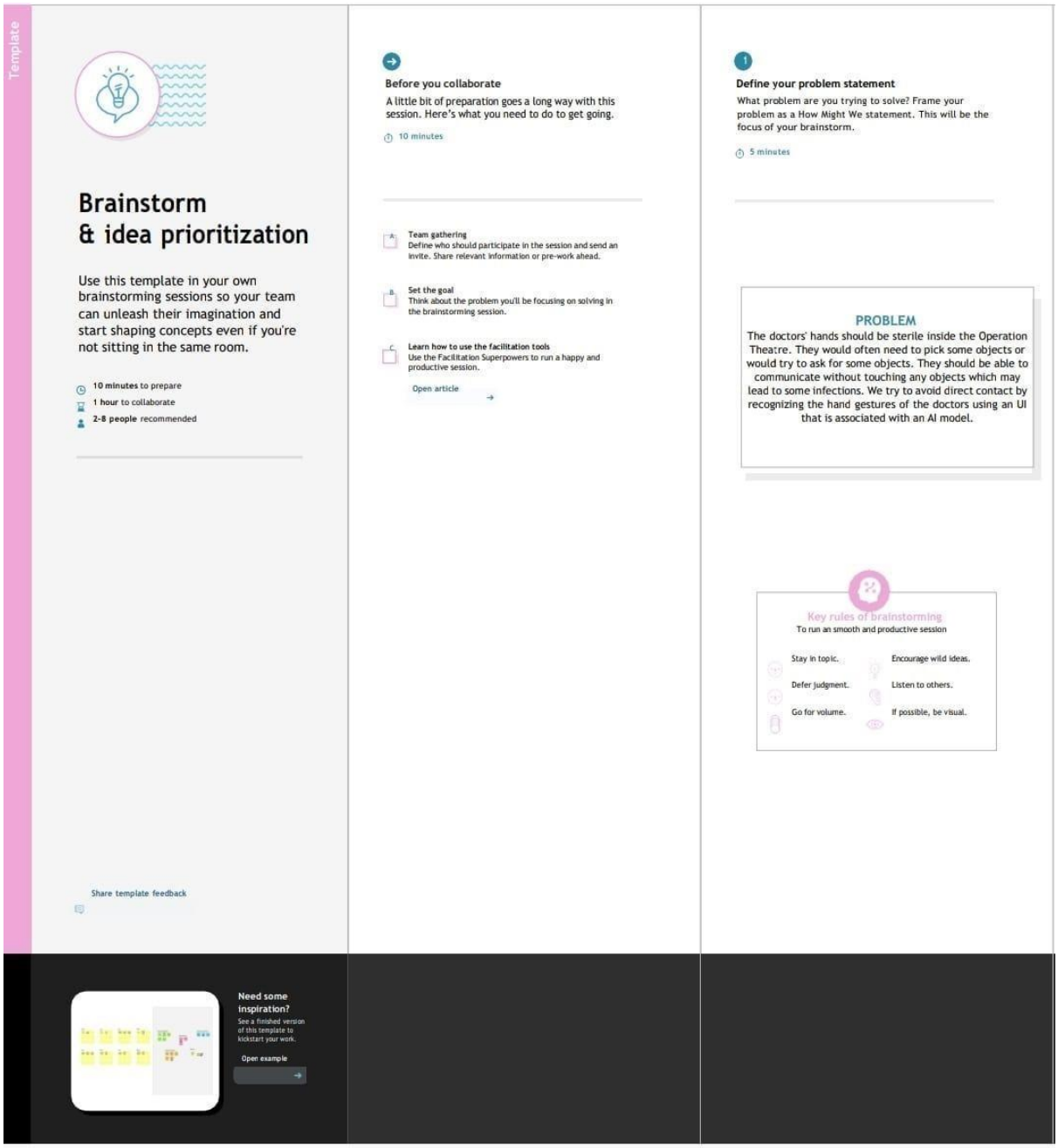
IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

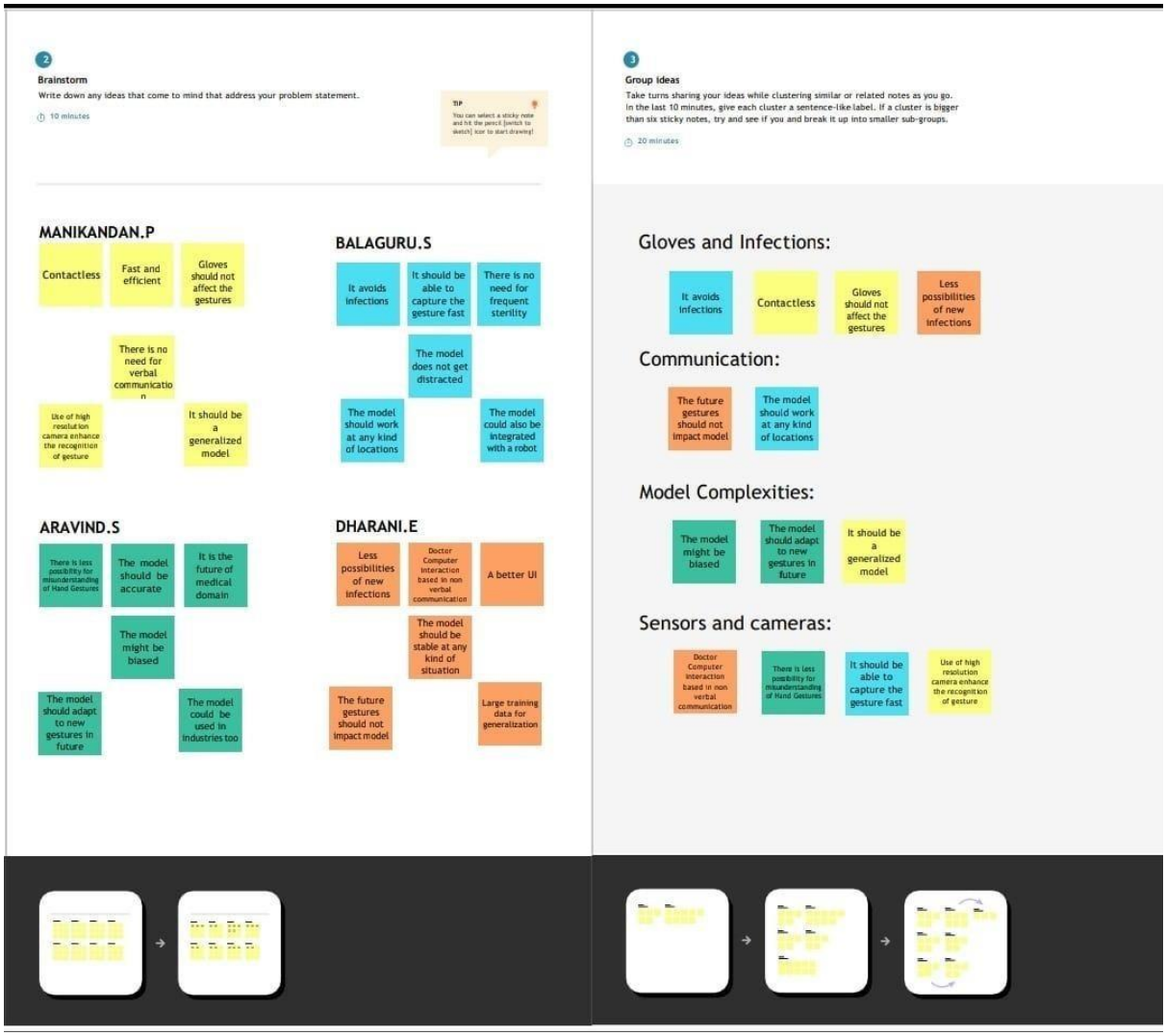


3.2 Ideation & Brainstorming

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3:Idea Prioritization

3.3 Problem Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	"Hand Gesture Recognition Using Camera" is based on concept of Image processing. In recent year there is lot of research on gesture recognition using kinect sensor on using HD camera but camera and kinect sensors are more costly. This project is mainly focused on to reduce cost and improve robustness of the proposed system using simple web camera.
2.	Idea / Solution description	Most gesture recognition methods usually contain three major stages. The first stage is the object detection. The target of this stage is to detect hand objects in the digital images or videos. Many environment and image problems are needed to solve at this stage to ensure that the hand contours or regions can be extracted precisely to enhance the recognition accuracy. Common image problems contain unstable brightness, noise, poor resolution and contrast. The better environment and camera devices can effectively improve these problems.

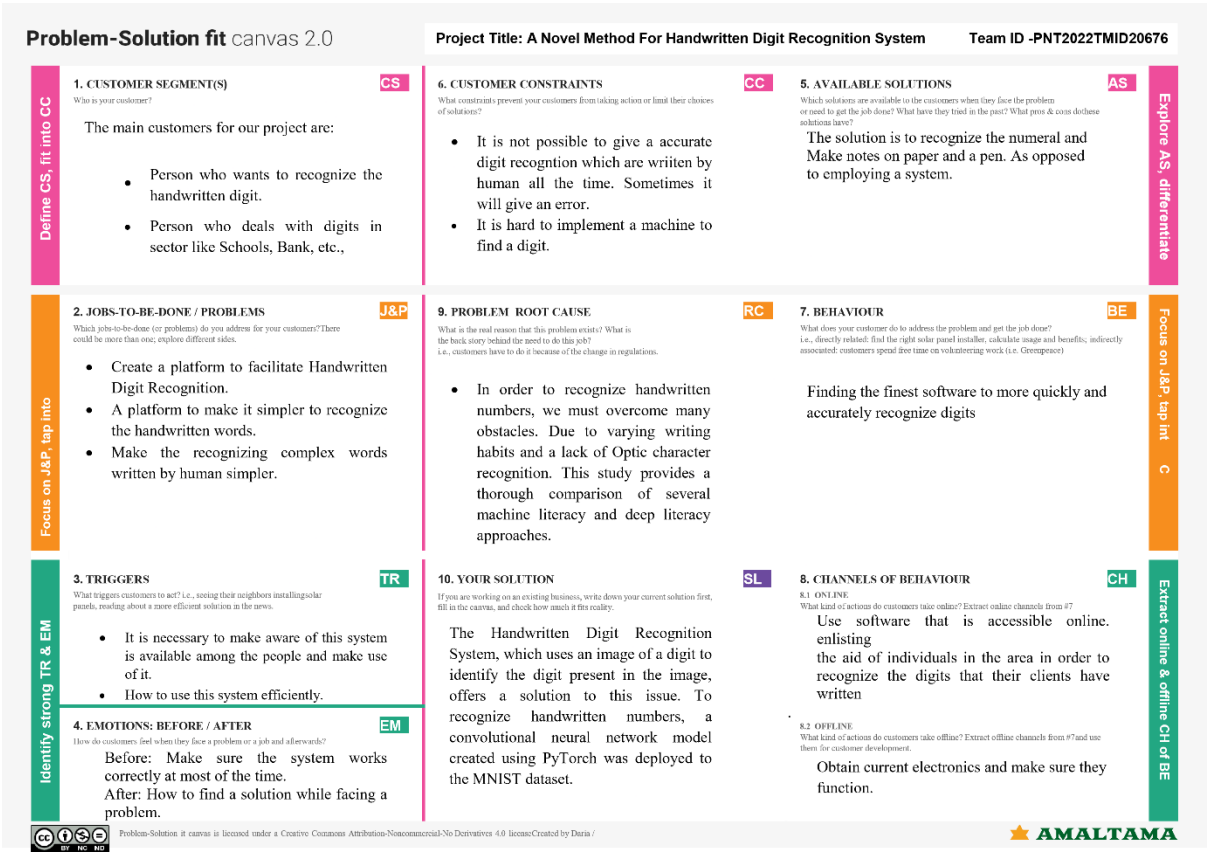
		However, it is hard to control when the gesture recognition system is working in the real environment or is become a product. Hence, the image processing method is a better solution to solve these image problems to construct an adaptive and robust gesture recognition system. The second stage is object recognition. The detected hand objects are recognized to identify the gestures. At this stage, differentiated features and effective classifiers selection are a major issue in most researches. The third stage is to analyze sequential gestures to identify users' instructs or behaviours.

3.	Novelty / Uniqueness	In this project, we mainly focus on using pointing behavior for a natural interface, Hand gesture recognition based human machine interface is being developed vigorously in recent years. Due to the effect of lightning and complex background, most visual hand gesture recognition systems work only under restricted environment. To classify the dynamic hand gestures, we developed a simple and fast motion history image based method. In recent years, the gesture control technique has become a new developmental trend for many humanbased electronics products. This technique let people can control these products more naturally, intuitively and conveniently. In this paper, a fast gesture recognition scheme is proposed to be an interaction (HMI) of systems. Our project mainly presents some low complexity algorithms and gestures to reduce the gesture recognition
		complexity and be more suitable for controlling real-time computer systems.

4.	Social Impact / Customer Satisfaction	Gesture technology comes as a boon to society, providing contact-less, safe, and inclusive experiences and they are easier representation, makes the presentation attractive, Quick expressing of message. A study shows that when the speaker uses gestures, the probability of the audience remembering the point is double than a normal speech.
5.	Business Model (Revenue Model)	Hand gesture recognition is a process of understanding and classifying meaningful movements by the human hands. Nowadays vehicles launched from the industry offers an increasing number of infotainment systems as well as comfort functions that can be controlled by the driver. Though they are feature rich which demands more attention of the driver and degrade the driving performance and thereby reducing the safety. The gestural interaction is a promising means to cover the full range of driver's operational needs while minimizing the visual workload thereby enhancing the drivers safety.

6.	Scalability of the Solution	Human hand is very smaller with very complex articulations comparing with the entire human body and therefore errors can be easily affected. Hand gesture recognition is of great importance for human computer interaction (HCI) because of its extensive applications in virtual reality and sign language recognition**.**

3.4 Problem Solution fit



Problem-Solution fit canvas 2.0

Project Title: A Novel Method For Handwritten Digit Recognition System

Team ID -PNT2022TMID20676

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?

The main customers for our project are:

- Person who wants to recognize the handwritten digit.
- Person who deals with digits in sector like Schools, Bank, etc.,

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions?

- It is not possible to give a accurate digit recogntion which are wriiten by human all the time. Sometimes it will give an error.
- It is hard to implement a machine to find a digit.

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons dothese solutions have?

The solution is to recognize the numeral and Make notes on paper and a pen. As opposed to employing a system.

Explore AS, differentiate

Focus on J&P, tap into

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers?There could be more than one; explore different sides.

- Create a platform to facilitate Handwritten Digit Recognition.
- A platform to make it simpler to recognize the handwritten words.
- Make the recognizing complex words written by human simpler.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?
i.e., customers have to do it because of the change in regulations.

- In order to recognize handwritten numbers, we must overcome many obstacles. Due to varying writing habits and a lack of Optic character recognition. This study provides a thorough comparison of several machine literacy and deep literacy approaches.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e., directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Finding the finest software to more quickly and accurately recognize digits

Focus on J&P, tap int C

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e., seeing their neighbors installingsolar panels, reading about a more efficient solution in the news.

- It is necessary to make aware of this system is available among the people and make use of it.
- How to use this system efficiently.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?

Before: Make sure the system works correctly at most of the time.
After: How to find a solution while facing a problem.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.

The Handwritten Digit Recognition System, which uses an image of a digit to identify the digit present in the image, offers a solution to this issue. To recognize handwritten numbers, a convolutional neural network model created using PyTorch was deployed to the MNIST dataset.

8. CHANNELS OF BEHAVIOUR CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

Use software that is accessible online. enlisting
the aid of individuals in the area in order to recognize the digits that their clients have written

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7and use them for customer development.

Obtain current electronics and make sure they function.

Extract online & offline CH of BE

AMALTAMA

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Link (HTML page)
FR-2	User Confirmation	Confirmation via Email
FR-3	Upload images as input	Add image through Device or through Drive
FR-4	Camera on	When the camera is on it recognize the hand gestures and return the required output.

4.2 Non-functional Requirements

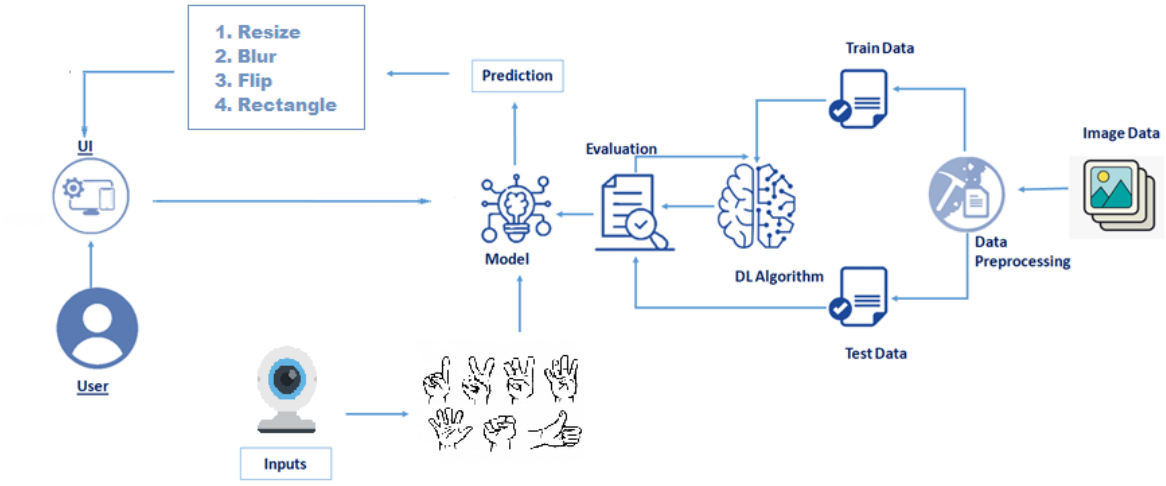
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	This software will be easy to use for all users with minimal instructions. 100% of the languages on the graphical user interface (GUI) shall be intuitive and understandable by non-technical users.
NFR-2	Security	The user of the system should be provided the surety that their account details are secure. The system will provide security against cross site request forgery.
NFR-3	Reliability	This software will be operable in all lighting conditions. Regardless of the brightness level in user's operating environment, the program shall always detect user's hands.
NFR-4	Performance	This software will minimize the number of calculations needed to perform image processing and hand gesture detection. Each captured video frame shall be processed fast.
NFR-5	Availability	This software will be available to all operating system. While it is currently has a relatively limited role in direct patient care, its evolving role in complex clinical decision making is foreseeable.

NFR-6	Scalability	This software will be enterprise scalability of AI development and deployment.

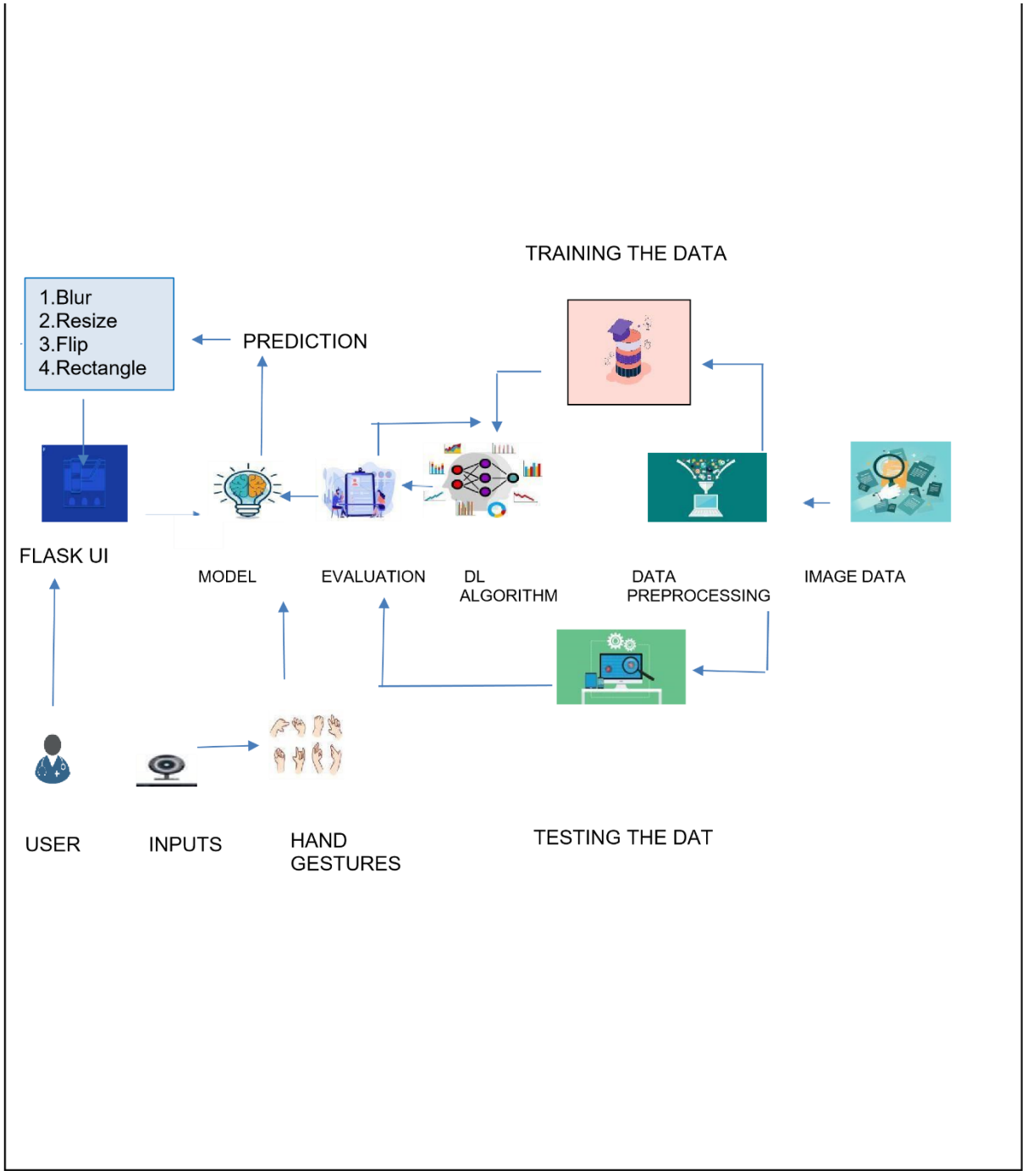
CHAPTER 5

PROJECT DESIGN

5.1 Data flow diagams



5.2 Solution Architecture



Technical Architecture

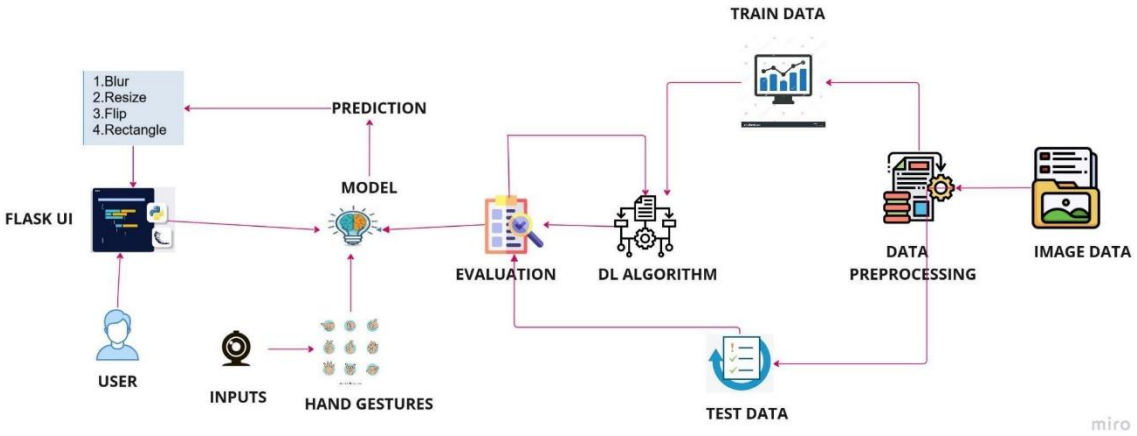


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI	HTML, CSS, JavaScript
2.	Dataset	Collect or create the hand gesture dataset	From Online
3.	Application Logic-1 – Data preprocessing	Import all the library files required for data pre-processing	Python
4.	Application Logic-2-Model Building	Build the CNN model	Python
5.	Application Logic-3- Application Building	Create HTML File	HTML, CSS, Javascript.
6.	File Storage	Store the code files and datasets	System Storage
7.	Deep Learning	Used to analyse visual imagery, image processing, video capture and analysis including features like face detection and object detection.	CNN, Opencv
8.	Cloud database	Train the model on IBM cloud	IBM Cloud

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Application development, data pre-processing.	Visual studio code, anaconda navigator.
2.	Security Implementations	It identify the gesture action only when the hand is in front of the camera.	Opencv

3.	Scalable Architecture	It can be used in any environment and is able to identify the gesture actions in both bright and dim backgrounds. It can recognize the gesture action upto 5 meters distance between the camera and person.	Opencv
4.	Availability	It is used to reduce the possibility of spreading infections, avoid the delay and the focus of doctors on surgery is improved.	Artificial Intelligence
5.	Performance	Rapid response to the gesture actions.	CNN model

5.3User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Launch Web App deployed in cloud	USN-1	As a user, I can launch the webapp where I can upload the images for recognition	I can upload the images for classification	High	Sprint-4
	Prediction	USN-2	As a user, I can get the predicted results from the model deployed in the cloud	I can resize the radiology image, blur the image, flip based on the hand gesture	High	Sprint-4
	Deployment of webapp in the cloud	USN-3	As a user, I need the webapp to be accessible all over the world	I can access the webapp deployed in the IBM cloud	Medium	Sprint-3
	Deployment of AI model in the cloud	USN-4	As a user, I need the AI model to be accessible all over the world	I can access the model deployed in the IBM cloud	Medium	Sprint-3

	Model Building	USN-5	As a user, I need an AI model which could classify or recognize the hand gestures	I can get the prediction from the AI model	Medium	Sprint-1
	User Interface Building	USN-6	As a user, I need a web app for human computer interaction	I get User Interface for interaction with the model	Medium	Sprint-2

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 Product Backlog, Sprint Schedule, and Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	High	Manikandan, Balaguru
		USN-2	As a user, I will receive confirmation email once I have registered for the application	High	Aravind, Dharani
		USN-3	As a user, I can register for the application through Facebook	Low	Manikandan, Dharani
		USN-4	As a user, I can register for the application through Gmail	Medium	Aravind, Dharani
Sprint 2	Login	USN-5	As a user, I can log into the application by entering email & password	High	Balaguru, Manikandan
	Dashboard	USN-6	As a user, I can view my profile and update my details	Medium	Balaguru, Manikandan
		USN-7	As a user, I can view all images uploaded	High	Aravind ,Dharani
		USN-8	As a user, I can change my password	High	Dharani, Balaguru
Sprint-3	Image Capturing	USN-9	As a user, I can capture images of hand gestures made by me	High	Manikandan, Aravind
	Image Processing	USN-10	In the application, the captured images are processed to identify the hand gesture	High	Dharani, Aravind
	Data Storage	USN-11	In the application radiology images uploaded by the user are stored using a database	High	Balaguru, Manikandan
Sprint-4	Sterile Browsing	USN-12	Depending on the different gesture inputs different operations are performed on the input image	High	Dharani, Aravind
	Displaying the operations performed	USN-13	Once the model analyses the gesture, the prediction with operation applied on the image is showcased on the user interface	High	Balaguru, Manikandan,Aravind

6.2 Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	06 Nov 2022
Sprint-2	20	6 Days	31Oct 2022	05 Nov 2022	20	08 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	14 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	18 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

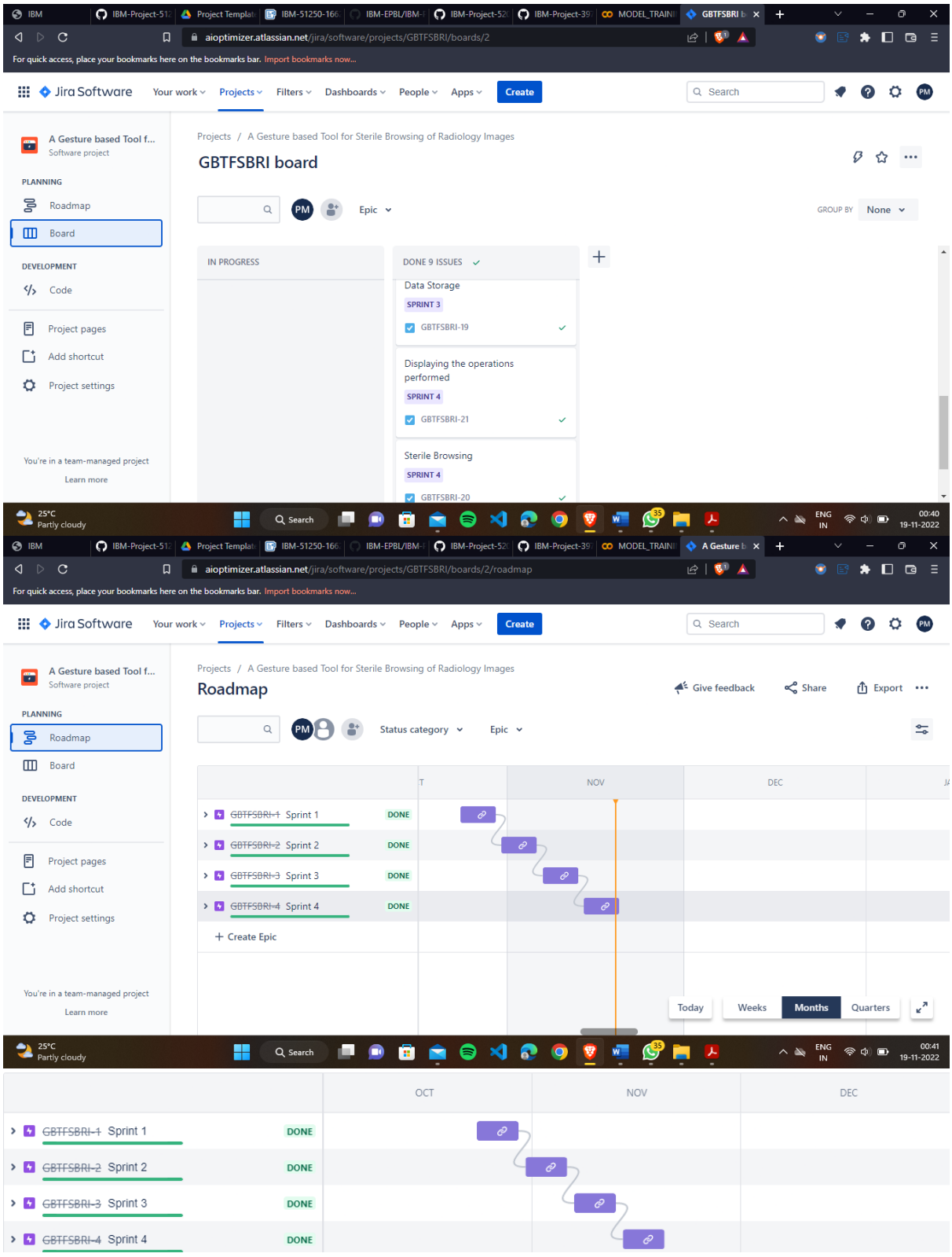
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

AV = sprint duration / velocity = 20/6 = 3.33

Burn down Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

6.3 Reports from JIRA



CHAPTER 7

CODING & SOLUTIONING

7.1 Feature 1

Main.css

```
.img-preview {
    width: 256px;
    height: 256px;
    position: relative;
    border: 5px solid #F8F8F8;
    box-shadow: 0px 2px 4px 0px rgba(0, 0, 0, 0.1);
    margin-top: 1em;
    margin-bottom: 1em;
}

.img-preview>div {
    width: 100%;
    height: 100%;
    background-size: cover;
    background-repeat: no-repeat;
    background-position: center;
}

input[type="file"] {
    display: none;
}

.upload-label{
    display: inline-block;
    padding: 12px 30px;
    background: #39D2B4;
    color: #fff;
    font-size: 1em;
    transition: all .4s;
    cursor: pointer;
}

.upload-label:hover{
    background: #34495E;
    color: #39D2B4;
}

.loader {
    border: 8px solid #f3f3f3; /* Light grey */
    border-top: 8px solid #3498db; /* Blue */
    border-radius: 50%;
```

```
    width: 50px;
    height: 50px;
    animation: spin 1s linear infinite;
}

@keyframes spin {
    0% { transform: rotate(0deg); }
    100% { transform: rotate(360deg); }
}
```

Main.js

```
$(document).ready(function () {
    // Init
    $('.image-section').hide();
    $('.loader').hide();
    $('#result').hide();

    // Upload Preview
    function readURL(input) {
        if (input.files && input.files[0]) {
            var reader = new FileReader();
            reader.onload = function (e) {
                $('#imagePreview').css('background-image', 'url(' + e.target.result + ')');
                $('#imagePreview').hide();
                $('#imagePreview').fadeIn(650);
            }
            reader.readAsDataURL(input.files[0]);
        }
    }
    $("#imageUpload").change(function () {
        $('.image-section').show();
        $('#btn-predict').show();
        $('#result').text('');
        $('#result').hide();
        readURL(this);
    });

    // Predict
    $('#btn-predict').click(function () {
        var form_data = new FormData($('#upload-file')[0]);

        // Show loading animation
        $(this).hide();
        $('.loader').show();

        // Make prediction by calling api /predict
        $.ajax({
            type: 'POST',
            url: '/predict',
            data: form_data,
            contentType: false,
            cache: false,
            processData: false,
```

```
            async: true,
            success: function (data) {
                // Get and display the result
                $('.loader').hide();
                $('#result').fadeIn(600);
                $('#result').html(data);
                console.log('Success!');
            },
        });
    });

});
```

7.2 Feature 2

home.html

```
<html>
<script>


</script>


<style>
.header { position: relative;
   top:0;
   margin:0px;
   z-index: 1;
   left: 0px;
   right: 0px;
   position: fixed;
   background-color:rgb(38, 38, 41) ;
   color: rgb(138, 58, 127);
   box-shadow: 0px 8px 4px rgb(215, 218, 8);
   overflow: hidden;
   padding-left:20px;
   font-family: 'Times New Roman';
   font-size: 2vw;
   width: 100%;
   height:8%;
   text-align: center;
  }
  .topnav {
  overflow: hidden;
  background-color: #84d39e;
}

.topnav-right a.active {
  background-color: rgb(38, 38, 41);
  color: rgb(241, 247, 241);
  font-size: 22px;

}
.topnav-right a:hover {
```

```
  color: rgb(13, 10, 236);
}

.topnav-right {
  float: right;
  padding-right:100px;
}

body {
  background-image: url("https://img3.goodfon.com/wallpaper/nbig/a/af/ruki-znaki-
      steny.jpg");
      background-size: cover;



  }
  .button {
  background-color: #1150ad;
  border: none;
  color: rgb(229, 232, 233);
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(7, 212, 18, 0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}



button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: rgb(181, 228, 236);
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
```

```
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
 margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}

.active {
  color: rgb(145, 216, 221);
}

/* Fading animation */
.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
```

```
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}




@import url("https://fonts.googleapis.com/css?family=Luckiest+Guy");
/* BODY */
body {
  position: absolute;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  width: 100%;
  height: 100%;
  overflow: hidden;
  font-family: "Arial", cursive;
  -webkit-font-smoothing: antialiased;
}

::selection {
  background: transparent;
}
/* CLOUDS */
body:before {
  content: "";
  position: absolute;
  top: 0;
  left: 0;
  right: 0;
  width: 0;
  height: 0;
  margin: auto;
  border-radius: 100%;
  background: transparent;
  display: block;
  box-shadow: 0 0 150px 100px rgba(255, 255, 255, 0.6),
    200px 0 200px 150px rgba(255, 255, 255, 0.6),
    -250px 0 300px 150px rgba(255, 255, 255, 0.6),
    550px 0 300px 200px rgba(255, 255, 255, 0.6),
    -550px 0 300px 200px rgba(255, 255, 255, 0.6);
}
/* JUMP */
h1 {
  cursor: default;
  position: absolute;
  top: 0;
  left: 0;
```

```
  right: 0;
  bottom: 0;
  width: 100%;
  height: 100px;
  margin: 70px;
  display: block;
  text-align: center;
}

h1 span {
  position: relative;
  top: 5px;
  display: inline-block;
  font-size: 35px;
  color: #190233;font-family: 'Times New Roman', Times, serif;

  text-shadow: 0 1px 0 rgb(221, 218, 55), 0 2px 0 rgb(146, 14, 235), 0 3px 0 rgb(151, 201,
      197), 0 4px 0 rgb(151, 201, 197),
    0 5px 0 rgb(151, 201, 197), 0 6px 0 transparent, 0 7px 0 transparent, 0 8px 0
      transparent,
    0 9px 0 transparent ;
}


/* ANIMATION */
@-webkit-keyframes bounce {
  100% {
    top: -20px;
    text-shadow: 0 1px 0 #ccc, 0 2px 0 #ccc, 0 3px 0 #ccc, 0 4px 0 #ccc,
      0 5px 0 #ccc, 0 6px 0 #ccc, 0 7px 0 #ccc, 0 8px 0 #ccc, 0 9px 0 #ccc,
      0 50px 25px rgba(0, 0, 0, 0.2);
  }
}

</style>

<body>
<h1 class="main" style="color: rgb(193, 207, 207);">
  <br>
    <span>HAND   GESTURE   RECOGNITION OF RADIOLOGY   IMAGES
      THROUGH </span>
  </br>
    <span>STERILE   BROWSING</span>

</h1>
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:#f4f5f4; padding-
      top:1%;padding-left:5%;"><b>HAND GESTURE RECOGNITION</b></div>
  <div class="topnav-right"style="padding-top:0.5%;color:white">
    <a  class="active" href="{{ url_for('home')}}"><u>Home</u></a>
    <a  class="active" href="{{ url_for('intro')}}">Introduction</a>
    <a  class="active" href="{{ url_for('image1')}}">Launch</a>
  </div>
</div>
```

```
</body>

</html>
```

intro.html

```
<html>
<script>


</script>


<style>
.header { position: relative;
   top:0;
   margin:0px;
   z-index: 1;
   left: 0px;
   right: 0px;
   position: fixed;
   background-color: rgb(38, 38, 41) ;
   color: white;
   box-shadow: 0px 8px 4px grey;
   overflow: hidden;
   padding-left:20px;
   font-family: 'Josefin Sans';
   font-size: 2vw;
   width: 100%;
   height:8%;
   text-align: center;
  }
  .topnav {
  overflow: hidden;
  background-color: #FCAD98;
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 22px;
}

.topnav-right a.active {
  background-color: rgb(38, 38, 41);
  color: rgb(238, 226, 234);
}

.topnav-right a:hover {
  background-color: rgb(38, 38, 41);
  color: rgb(13, 10, 236);
}
```

```
.topnav-right {
  float: right;
  padding-right:100px;
}

body {
  background-repeat: no-repeat;
  background-size:cover;

    background-size: cover;
  background-position: 0px 0px;
  }
  .button {
  background-color: #091425;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 12px;
  border-radius: 16px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: white;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
```

```
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}
.active {
  background-color: #FCAD98;
}


@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}



@import url("https://fonts.googleapis.com/css?family=Montserrat&display=swap");

* {
  padding: 0;
  margin: 0;
}

body {
  height: 100vh;
  display: flex;
  flex-direction: column;
```

```
  justify-content: center;
  align-items: center;
}

h1 {
  font-family: "Montserrat Medium";
  max-width: 90ch;
  text-align: center;
  transform: scale(0.94);
  animation: scale 3s forwards cubic-bezier(0.5, 1, 0.89, 1);
}
@keyframes scale {
  100% {
    transform: scale(1);
  }
}

span {
  display: inline-block;
  opacity: 0;
  filter: blur(4px);
}

span:nth-child(1) {
  animation: fade-in 1s 0.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(2) {
  animation: fade-in 0.8s 0.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(3) {
  animation: fade-in 0.8s 0.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(4) {
  animation: fade-in 0.8s 0.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(5) {
  animation: fade-in 0.8s 0.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(6) {
  animation: fade-in 0.8s 0.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(7) {
  animation: fade-in 0.8s 0.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(8) {
  animation: fade-in 0.8s 0.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(9) {
```

```
  animation: fade-in 0.8s 0.9s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(10) {
  animation: fade-in 0.8s 1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(11) {
  animation: fade-in 0.8s 1.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(12) {
  animation: fade-in 0.8s 1.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(13) {
  animation: fade-in 0.8s 1.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(14) {
  animation: fade-in 0.8s 1.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(15) {
  animation: fade-in 0.8s 1.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(16) {
  animation: fade-in 0.8s 1.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(17) {
  animation: fade-in 0.8s 1.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}

span:nth-child(18) {
  animation: fade-in 0.8s 1.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(19) {
  animation: fade-in 0.8s 1.9s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(20) {
  animation: fade-in 0.8s 2.0s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(21) {
  animation: fade-in 0.8s 2.1s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(22) {
  animation: fade-in 0.8s 2.2s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
span:nth-child(23) {
  animation: fade-in 0.8s 2.3s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(24) {
  animation: fade-in 0.8s 2.4s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(25) {
```

```
  animation: fade-in 0.8s 2.5s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(26) {
  animation: fade-in 0.8s 2.6s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(27) {
  animation: fade-in 0.8s 2.7s forwards cubic-bezier(0.11, 0, 0.5, 0);
}span:nth-child(28) {
  animation: fade-in 0.8s 2.8s forwards cubic-bezier(0.11, 0, 0.5, 0);
}
@keyframes fade-in {
  100% {
    opacity: 1;
    filter: blur(0);
  }
}
body h1{
  color: #030635;
  top: 100px;
  height: 50px;

}
body h2 span{
  color:teal;
  height: 50px;
  font-family: serif;

}

</style>

<body>
  <h1 class="intro">INTRODUCTION</h1>
<h2>
  <span>&#9757;&#127995;  Hand Gesture </span> <span> recognition system </span>
      <span> provides us </span > <span> an innovative,</span> <span>natural,</span>
      <span>  user friendly </span> <span> way of interaction </span > <span> with the
      computer</span>
  <span> which is more </span> <span> familiar to the </span> <span>human beings. </span >
      <br><span>&#9996;&#127995;  In our project, </span> <span> the hand region </span>
      <span> is extracted from </span> <span> the background  </span > <span> by
      using</span>
<span> Region of intrest. </span> <br><span>&#128400;&#127995;   Then, </span> <span>
      we will be </span > <span>predicting the labels </span> <span> based on the </span>
      <span> CNN trained model weights  </span> <span> of hand gestures </span > <span>
      using that predicted labels</span>
<span> we apply if conditions </span> <span> to control some of the actions </span>
      <span>like </span > <span>reshaping , blur, flip of the given image.</span>

</h2>
<!--Brian Tracy-->

<div class="header">
<div style="width:50%;float:left;font-size:30px;text-align:left;color:#f4f5f4; padding-
      top:1%;padding-left:5%;">Hand Gesture System</div>
  <div class="topnav-right"style="padding-top:0.5%;">
```

```
    <a class="active" href="{{ url_for('home')}}">Home</a>
    <a class="active" href="{{ url_for('intro')}}"><u>Introduction</u></a>
    <a class="active" href="{{ url_for('image1')}}">Launch</a>
  </div>
</div>
</body>

</html>
```

launch.html

```
<html lang="en">

  <head>
  <meta charset="utf-8">
     <meta http-equiv="X-UA-Compatible" content="IE=edge">
     <meta name="viewport" content="width=device-width, initial-scale=0.6">
     <script
     src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

     <link href="https://fonts.googleapis.com/icon?family=Material+Icons"
     rel="stylesheet">
     <meta charset="UTF-8">
     <title>Predict</title>
     <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
     rel="stylesheet">

     <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
     <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
     <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
     <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">
  <style>

  a
  {
  color:black;
  float:right;
  text-decoration:none;
  font-style:normal;
  padding-right:20px;
  }
  a:hover{
  background-color:black;
  color:black;
  font-size:30px;
  padding-left:10px;
  }

  div1{
    text-align: center;
    width: 650spx;
    height: 800px;
    padding: 190px;
    margin: 10px;
```

```
  position: absolute;
}



body
{
    background-image: url("https://img3.goodfon.com/wallpaper/nbig/a/af/ruki-znaki-
     steny.jpg");
    background-size: cover;
}

.header { position: relative;
      top:0;
      margin:0px;
      z-index: 1;
      left: 0px;
      right: 0px;
      position: fixed;
      background-color: rgb(87, 91, 92) ;
      color: black;
      box-shadow: 0px 8px 4px grey;
      overflow: hidden;
      padding-left:20px;
      font-family: 'Josefin Sans';
      font-size: 2vw;
      width: 100%;
      height:8%;
      text-align: center;
    }
    .topnav {
  overflow: hidden;
  background-color: #056959;
}

.topnav-right a {
  float: left;
  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 22px;
}

.topnav-right a.active {
  background-color:  rgb(87, 91, 92);
  color: rgb(238, 226, 234);
}

.topnav-right a:hover {
  background-color: rgb(87, 91, 92);
  color: rgb(13, 10, 236);
}
```

```
.topnav-right {
  float: right;
  padding-right:100px;
}


.button {
background-color: #091425;
border: none;
color: black;
padding: 15px 32px;
text-align: center;
text-decoration: none;
display: inline-block;
font-size: 12px;
border-radius: 16px;
}
.button:hover {
  box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
form {border: 2px solid black; margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  display: inline-block;
  margin-bottom:18px;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #091425;
  color: black;
  padding: 14px 20px;
  margin-bottom:10px;
  border: none;
  cursor: pointer;
  width: 17%;
  border-radius:4px;
  font-family:Montserrat;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
```

```
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 30%;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}

.home{
  margin:80px;

  width: 84%;
  height: 500px;
  padding-top:10px;
  padding-left: 30px;

}
.login{
  margin:80px;
  box-sizing: content-box;
  width: 84%;
  height: 420px;
  padding: 30px;
  border: 10px solid rgb(12, 91, 94);
}
.left,.right{
 box-sizing: content-box;
 height: 400px;
 margin:20px;
 border: 10px solid rgb(12, 91, 94);
}

.mySlides {display: none;}
img {vertical-align: middle;}
```

```
/* Slideshow container */
.slideshow-container {
  max-width: 1000px;
  position: relative;
  margin: auto;
}


.active {
  background-color: #267481;
}

@-webkit-keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

@keyframes fade {
  from {opacity: .4}
  to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
  .text {font-size: 11px}
}

</style>
</head>

<body>
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:#c1e2d9; padding-
    top:1%;padding-left:5%;">Hand Gesture System</div>
  <div class="topnav-right"style="padding-top:0.5%;">

    <a class="active" href="{{ url_for('home')}}">Home</a>
    <a class="active" href="{{ url_for('intro')}}">Introduction</a>
    <a class="active" href="{{ url_for('image1')}}"><u>Launch</u></a>
  </div>
</div>
<br>

  <div1 style=""><h1><font color="Black" size="6" font-family="Roboto">Hand
    Gesture Recognition</h1><br>
  <p><i><font color="Black" size="4" fonr-family="sans-serif"></i>Provide an image
    for which you want to perform various operations</p>
  <br>
        <div>
          <h4>Upload Image Here</h4>
      <form action = "http://localhost:5000/" id="upload-file" method="post"
    enctype="multipart/form-data">
        <label for="imageUpload" class="upload-label">
          Choose...
        </label>
```

```
        <input type="file" name="image" id="imageUpload" accept=".png, .jpg,
    .jpeg,.pdf">
      </form>
          <center>
      <div class="image-section" style="display:none;">
        <div class="img-preview">
          <div id="imagePreview">
          </div>
        </div>
        <div>
          <button type="button" class="btn btn-info btn-lg " id="btn-
    predict">Predict!</button>
        </div>
      </div>
      <div class="loader" style="display:none;"></div>
          </center>
    </div>
    </div1>

        <footer>
    <script src="{{ url_for('static', filename='js/main.js') }}"
    type="text/javascript"></script>
</footer>
</html>
```

CHAPTER 8

TESTING

8.1 Acceptance Testing

Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [A Gesture based hand recognition system] project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	4	2	1	12
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	5	2	4	1	12
Not Reproduced	0	0	1	0	1
Skipped	2	0	1	1	4
Won't Fix	1	2	2	1	6
Totals	16	11	13	5	45

Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Web UI	7	0	0	7
Launching Camera	10	1	0	9

Training Set	14	0	2	12
Flask Integration	6	0	0	3
Cloud Deployment	10	0	1	9
Final Report Output	5	0	0	5
Version Control	2	0	0	2

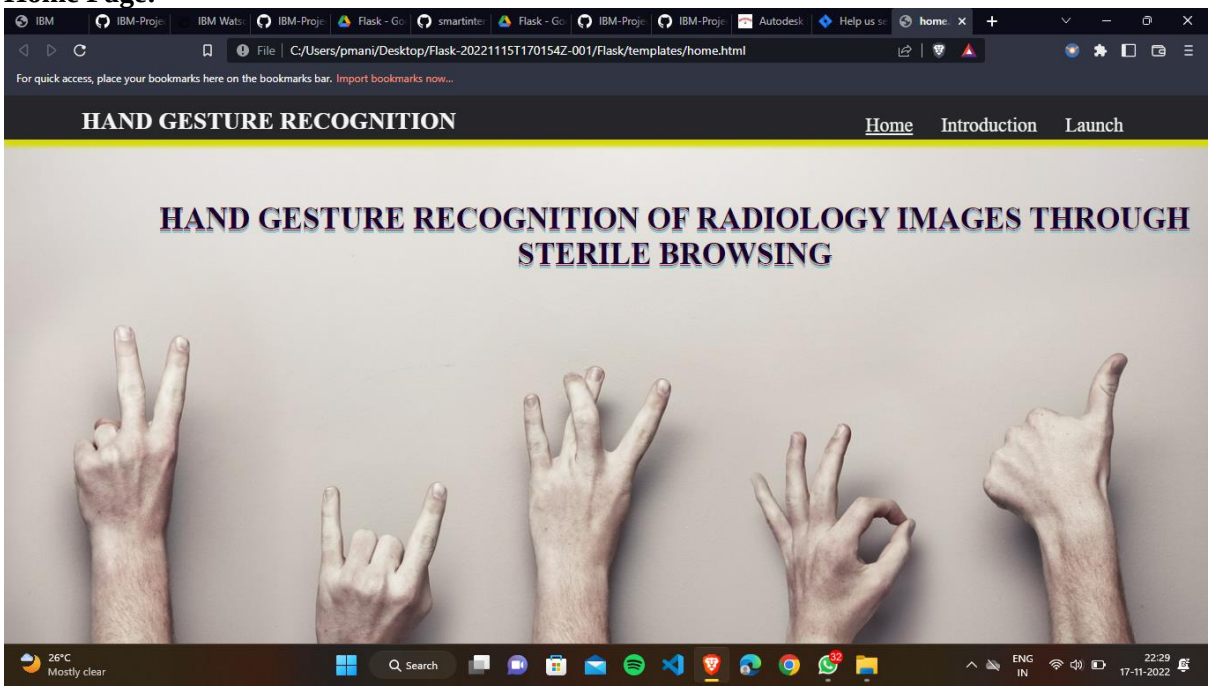
8.2 Perfoemance Testing

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 3,224,422 Trainable params: 3,224,422 Non-trainable params: 0	Attached below
2.	Accuracy	Training Accuracy - 98% Validation Accuracy - 60%	Attached below

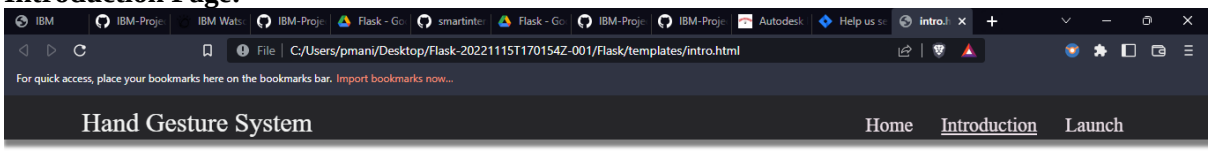
CHAPTER 9
RESULTS

9.1 Performance Metrics:

Home Page:



Introduction Page:



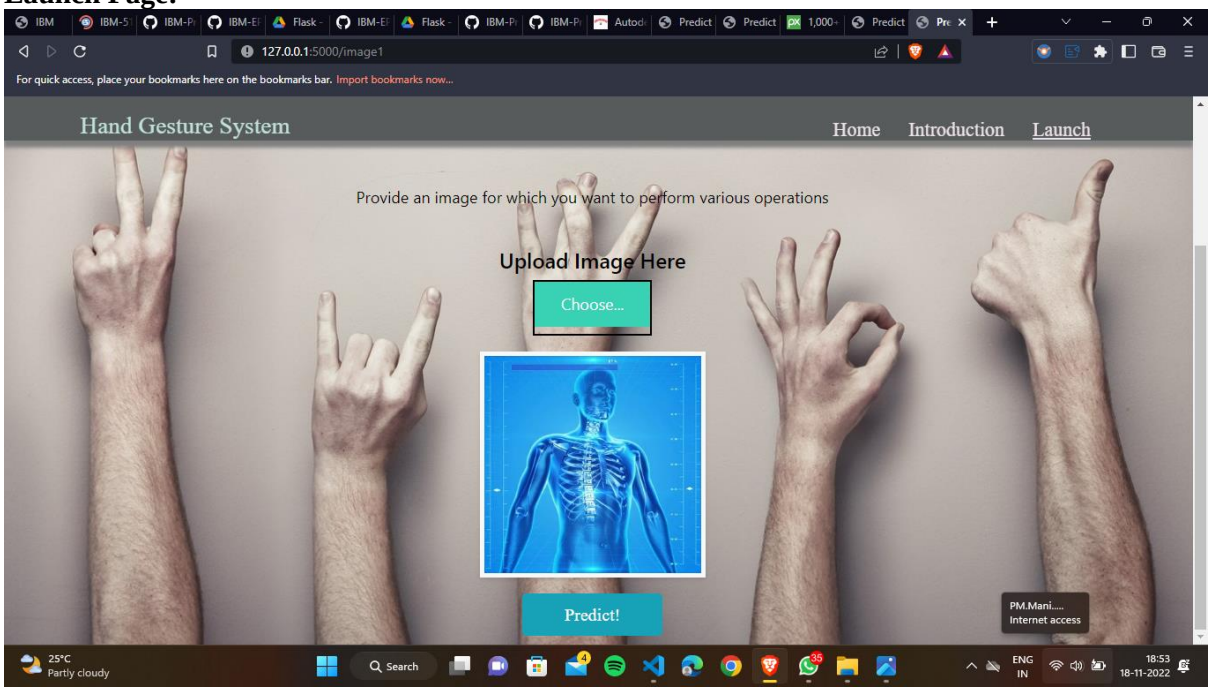
INTRODUCTION

☝ Hand Gesture recognition system provides us an innovative, natural, user friendly way of interaction with the computer which is more familiar to the human beings.

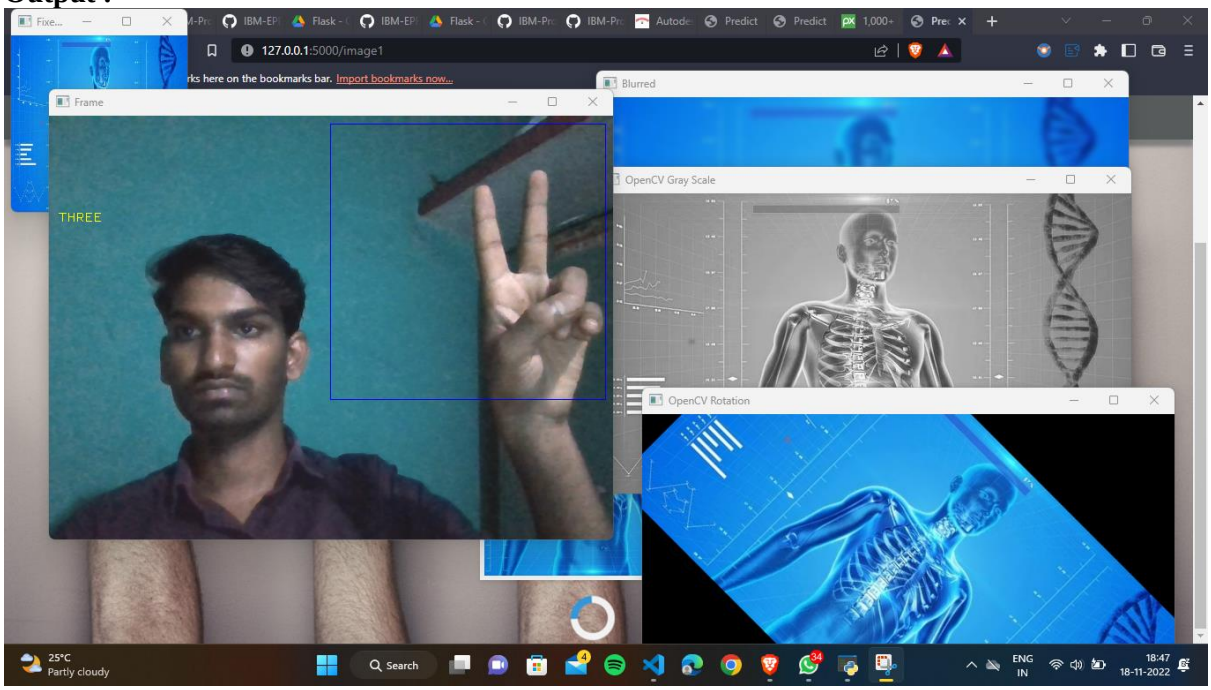
✌ In our project, the hand region is extracted from the background by using Region of intrest.

✋ Then, we will be predicting the labels based on the CNN trained model weights of hand gestures using that predicted labels we apply if conditions to control some of the actions

Launch Page:



Output :



CHAPTER 10
ADVANTAGES & DISADVANTAGES

Advantages:

1. Major advantage of this tool is that it helps to maintain the sterility of the environment.
2. It is also easy to use and is quicker than the existing methods to browse images.
3. It can also be performed even if the surgeon is a bit far away from the system, this helps t o save time.
4. The tool does not need the person using it to have an apparatus or any devices on them to use it.
5. They can simply move their hands to browse through the images.

Disadvantages:

1.The tool can be quite expensive as it requires cameras and other expensive devices to cap
ture images and process it.

CHAPTER 11

CONCLUSION:

- In this project we developed a tool which recognises hand gestures and enables doctors to browse througH radiology images using these gestures.
- This enables doctors and surgeons to maintain the sterility as they would not have to touch any mouse or keyboard to go through the images.
- This tool is also easy to use and is quicker than the regular method of using mouse/keyboard.
- It can be used regardless of the users location since they don’t have to be in contact with any device.
- It also does not require the user to have any device on them to use it.
- Further this technology can be extended to other industries like it can be used by presenters, by teachers for show images in the classroom, etc.

CHAPTER 12

FUTURE SCOPE

1. The tool can be made quicker by increasing the recognition speed.

2. More number of gestures can be added thereby increasing this tool's functionality and use ability for different purposes.

3. Tracking of both hands can be added to increase the set of commands.
   Voice commands can also be added to further increase the functionality.

CHAPTER 13

APPENDIX

Source Code:

app.py

```
from flask import Flask,render_template,request
import operator
import cv2
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import numpy as np
import os
from tensorflow.keras.models import load_model
from werkzeug.utils import secure_filename
app = Flask(__name__,template_folder="templates")
print(os.getcwd())
model=load_model('Flask/gesture.h5')
print("Loaded model from disk")
@app.route('/')
def home():
    return render_template('home.html')

@app.route('/intro')
def intro():
    return render_template('intro.html')
@app.route('/image1',methods=['GET','POST'])
def image1():
    return render_template("launch.html")


@app.route('/predict',methods=['GET', 'POST'])
def launch():
    if request.method == 'POST':
        print("inside image")
        f = request.files['image']

        basepath = os.path.dirname(__file__)
        file_path = os.path.join(basepath, 'uploads', secure_filename(f.filename))
        f.save(file_path)
        print(file_path)
        cap = cv2.VideoCapture(0)
        while True:
            _, frame = cap.read()
            frame = cv2.flip(frame, 1)

            x1 = int(0.5*frame.shape[1])
            y1 = 10
            x2 = frame.shape[1]-10
            y2 = int(0.5*frame.shape[1])
```

```
        cv2.rectangle(frame, (x1-1, y1-1), (x2+1, y2+1), (255,0,0) ,1)
        roi = frame[y1:y2, x1:x2]
        roi = cv2.resize(roi, (64, 64))
        roi = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
        _, test_image = cv2.threshold(roi, 120, 255, cv2.THRESH_BINARY)
        cv2.imshow("test", test_image)
        # Batch of 1
        result = model.predict(test_image.reshape(1, 64, 64, 1))
        prediction = {'ZERO': result[0][0],
                      'ONE': result[0][1],
                      'TWO': result[0][2],
                      'THREE': result[0][3],
                      'FOUR': result[0][4],
                      'FIVE': result[0][5]}
        prediction = sorted(prediction.items(), key=operator.itemgetter(1),
    reverse=True)

        # Displaying the predictions
        cv2.putText(frame, prediction[0][0], (10, 120),
    cv2.FONT_HERSHEY_PLAIN, 1, (0,255,255), 1)
        cv2.imshow("Frame", frame)

        #loading an image
        image1=cv2.imread(file_path)
        if prediction[0][0]=='ONE':

            resized = cv2.resize(image1, (200, 200))
            cv2.imshow("Fixed Resizing", resized)
            key=cv2.waitKey(3000)

            if (key & 0xFF) == ord("1"):
                cv2.destroyWindow("Fixed Resizing")

        elif prediction[0][0]=='ZERO':

            cv2.rectangle(image1, (480, 170), (650, 420), (0, 0, 255), 2)
            cv2.imshow("Rectangle", image1)
            cv2.waitKey(0)
            key=cv2.waitKey(3000)
            if (key & 0xFF) == ord("0"):
                cv2.destroyWindow("Rectangle")

        elif prediction[0][0]=='TWO':
            (h, w, d) = image1.shape
            center = (w // 2, h // 2)
            M = cv2.getRotationMatrix2D(center, -45, 1.0)
            rotated = cv2.warpAffine(image1, M, (w, h))
            cv2.imshow("OpenCV Rotation", rotated)
            key=cv2.waitKey(3000)
            if (key & 0xFF) == ord("2"):
                cv2.destroyWindow("OpenCV Rotation")
```

```
            elif prediction[0][0]=='THREE':
                blurred = cv2.GaussianBlur(image1, (21, 21), 0)
                cv2.imshow("Blurred", blurred)
                key=cv2.waitKey(3000)
                if (key & 0xFF) == ord("3"):
                    cv2.destroyWindow("Blurred")

            elif prediction[0][0]=='FOUR':

                resized = cv2.resize(image1, (400, 400))
                cv2.imshow("Fixed Resizing", resized)
                key=cv2.waitKey(3000)
                if (key & 0xFF) == ord("4"):
                    cv2.destroyWindow("Fixed Resizing")

            elif prediction[0][0]=='FIVE':
                '''(h, w, d) = image1.shape
                center = (w // 2, h // 2)
                M = cv2.getRotationMatrix2D(center, 45, 1.0)
                rotated = cv2.warpAffine(image1, M, (w, h))'''
                gray = cv2.cvtColor(image1, cv2.COLOR_RGB2GRAY)
                cv2.imshow("OpenCV Gray Scale", gray)
                key=cv2.waitKey(3000)
                if (key & 0xFF) == ord("5"):
                    cv2.destroyWindow("OpenCV Gray Scale")

            else:
                continue


            interrupt = cv2.waitKey(10)
            if interrupt & 0xFF == 27: # esc key
                break


        cap.release()
        cv2.destroyAllWindows()
    return render_template("home.html")

if __name__ == "__main__":
    # running the app
    app.run(debug=False)
```

GitHub Link : https://github.com/IBM-EPBL/IBM-Project-51250-1660976530

Project Demo Link:

https://drive.google.com/file/d/1JZ2vyxpiMQsmF5EC0sHJmSPMrSMHbiSv/view?usp=share_link