TABLE OF CONTENTS

CHAPTER - 1

INTRODUCTION

1.1 PROJECT OVERVIEW

In this project we use gestures to browse images obtained during radiology. Gestures refer to non-verbal form of communication made using hands. 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. Humans can 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 0 - then images is converted into rectangle, 1 - image is Resized into (200,200), 2 - image is rotated by -45°, 3 - image is blurred, 4 - image is Resized into (400,400), 5 - image is converted into grayscale etc.

1.2 PURPOSE

It is used to browse through the images obtained using radiology using hand gestures rather than using mouse, keyboard,etc thereby maintaining sterility.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.

CHAPTER - 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

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 . We found that many hospitals rely on mouse and keyboard to browse the images that are obtained during different surgeries, scans, etc. This can contaminate the environment with various infections thus compromising the sterility. 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.

2.2 SURVEY WORK

2.2.1 A gesture-controlled projection display for CT-guided interventions

[A.Mewes et al.,2016]

The interaction with interventional imaging systems among a sterile surroundings could be a challenging task for physicians. Direct physician–machine interaction throughout associate intervention is quite restricted owing to sterility and space restrictions. Methods We gift a gesture-controlled projection show that allows an immediate and natural physician–machine interaction throughout computerized tomography (CT)-based interventions.we have a tendency to propose

a gesture set to regulate basic functions of intervention software system like gestures for 2nd image exploration, 3D object manipulation and choice. Our ways were evaluated in an exceedingly clinically homeward user study with twelve participants.

2.2.2 Agreement Study Using Gesture Description Analysis [Naveen Madapana et al.,2020]

Choosing adequate gestures for touchless interfaces could be a difficult task that incorporates a direct impact on human–computer interaction. Such gestures are unremarkably determined by the designer, ad-hoc, rule-based, or agreement-based strategies.Our method is evaluated and tested through a guessability study conducted with a gaggle of the neurosurgeons. yet, our formulation may be applied to the other user-elicitation study. Results show that the extent of agreement obtained by SAR is a pair of.64 times beyond the previous metrics.

2.2.3 A Comprehensive Leap Motion Database for Hand Gesture Recognition [Safa AMEUR et al.,2016]

The touchless interaction has received goodly attention in recent years with benefit of removing the burden of physical contact. The recent introduction of novel acquisition devices, just like the leap motion controller, permits getting a really informative description of the hand create and motion which will be exploited for correct gesture recognition. during this work, we have a tendency to gift Associate in Nursing interactive application with gestural hand management using

leap motion for medical visualization, that specialize in the satisfaction of the user as an important part within the composition of a brand new specific information.

2.2.4 Application of Natural User Interface Devices for Touch-free Control of Radiological Images During Surgery

[Nikola Nestorov et al.,2016]

Natural programme (NUI) systems will alter the clean practician to assume direct management of medical image interaction whereas maintaining sterility within the in operation Room. Surgeons and radiologists trialed a touch-free image system supported the Leap Motion and Microsoft Kinect v2 controllers. Feedback was according on the perceived utility and usefulness of each devices. The speed and accuracy of the 2 controllers was measured. Results showed marginal to average acceptableness of each controllers. Surgeons and Interventional Radiologists found Microsoft Kinect to own better utility and to be probably helpful for the bulk (54%) of them.

2.2.5 You Can’t Touch This: Touch-free Navigation Through Radiological Images

[Michael Thali et al.,2012]

Keyboards, mice, and bit screens area unit a possible supply of infection or contamination in operational rooms, medical aid units, and autopsy suites. The authors gift a low-cost epitome of a system, that permits for touch-free management of a medical image viewer. This touch-free navigation system consists of a computing system (IMac, OS X 10.6 Apple, USA) with a medical image viewer (OsiriX,

OsiriX foundation, Switzerland) and a depth camera (Kinect, Microsoft, USA).

2.2.6 Leap Motion Gesture Control With Carestream Software in the Operating Room to Control Imaging: Installation Guide and Discussion

[Julien Pauchot et al.,2015]

Nowadays, routine cross-sectional imaging viewing throughout a surgical treatment needs physical contact with AN interface (mouse or touch-sensitive screen). Such contact risks exposure to antiseptic conditions and causes loss of your time. Devices like the recently introduced Leap Motion (Leap Motion Society, city, CA), that permits interaction with the pc with none physical contact, ar of wide interest within the field of surgery, however configuration and bioengineering ar key challenges for the practitioner, imaging computer code, and surgical setting.

2.2.7 Out of touch – A plugin for controlling OsiriX with gestures using the leap controller

[L.C.Ebert et al.,2014]

In recent years, totally different systems for gesture management of medical devices are presented. Today, cheap gesture management systems ar commercially obtainable. In this article, we tend to gift a plugin for the OsiriX medical image viewer, that operates the viewer victimization finger gestures. we tend to use a tool referred to as the Leap controller for gesture control input. The device contains a low price, and it uses structured lightweight to form

a depth image of the screen.

2.2.8 Informatics in Radiology: Developing a Touchless User Interface for Intraoperative Image Control during Interventional Radiology Procedures.

[Justin H.Tan et al.,2013]

Review of previous Associate in Nursing time period patient pictures is crucial throughout an interventional radiology procedure; but, it usually poses the challenge of with efficiency reviewing pictures whereas maintaining a sterile field. though interventional radiologists will “scrub out” of the procedure, use sterile console covers, or verbally relay directions to Associate in Nursing assistant, the ability of the interventionalist to directly management the pictures while not having to the touch the console might supply potential gains in terms of sterility, procedure potency, and radiation reduction. The authors investigated a possible resolution with a inexpensive, touch-free motion-tracking device that was originally designed as a game controller.. A custom code program known as the Touchless Radiology Imaging management System interprets motion info obtained with the motion-tracking device into commands to review pictures on a digital computer.The majority (69%) of these surveyed believed that the device might be helpful in Associate in Nursing interventional radiology apply and did not foresee issues with maintaining a sterile field.

2.3 PROBLEM STATEMENT DEFENITION

Problem Statement (PS)	I am a	I'm trying to	But	Because	Which makes me feel
PS-1	News Editor	Blur and resize the images	It takes long time for large volume of images	Large volume of images to beedited	Tired
PS-2	Doctor	Analyze the scanned reports	I will also be performing surgery	I can't analyze the reports while performing surgery	Incapable
PS-3	Designer	Modify the image	It feels likea repeated task	Everything has to be modified manually	Bored

CHAPTER - 3

IDEATION & PROPOSED SOLUTION

3.1EMPATHY MAP CANVAS

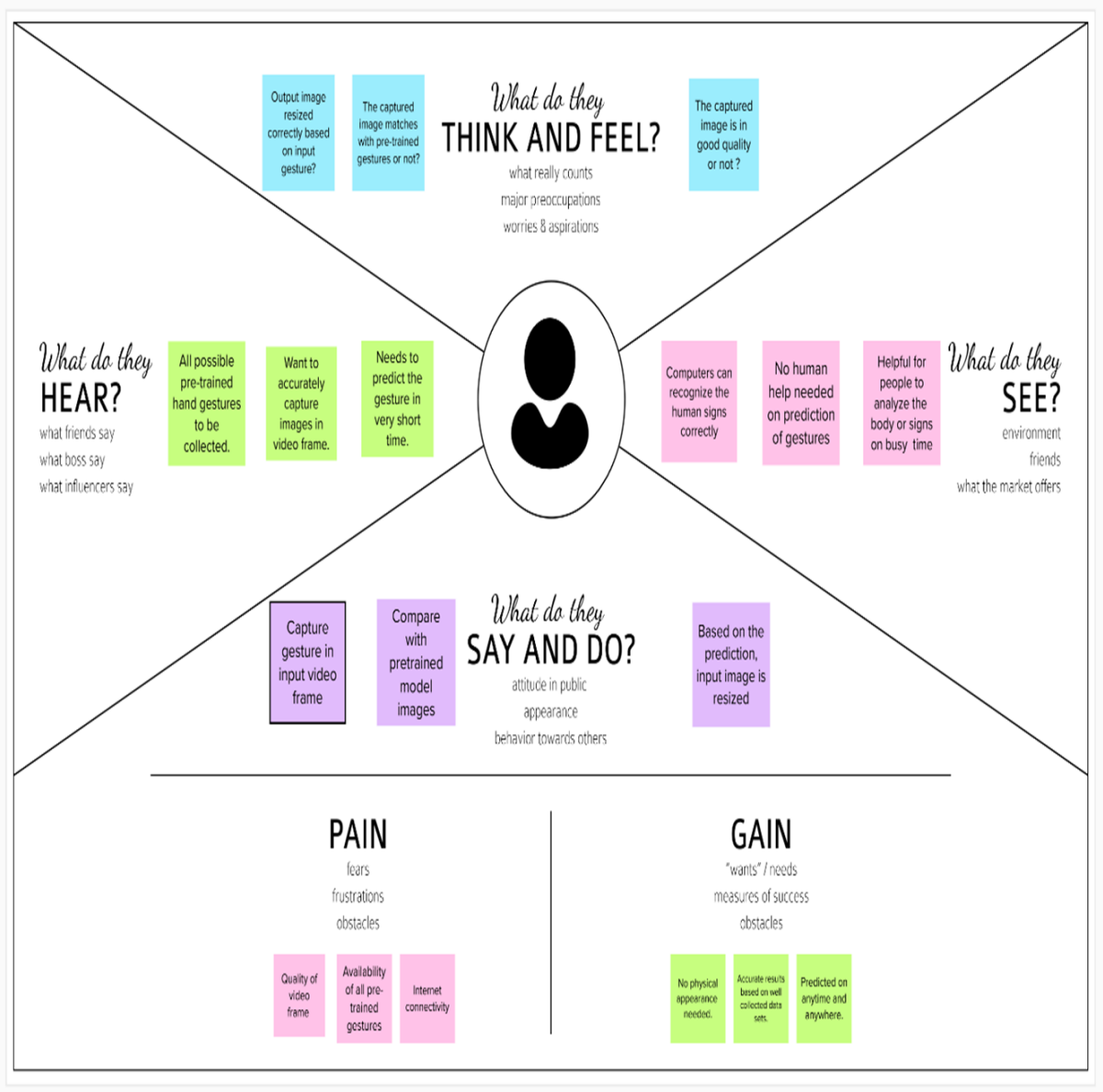


Fig 3.1 Empathy Map canvas

3.2 IDEATION & BRAINSTROMING

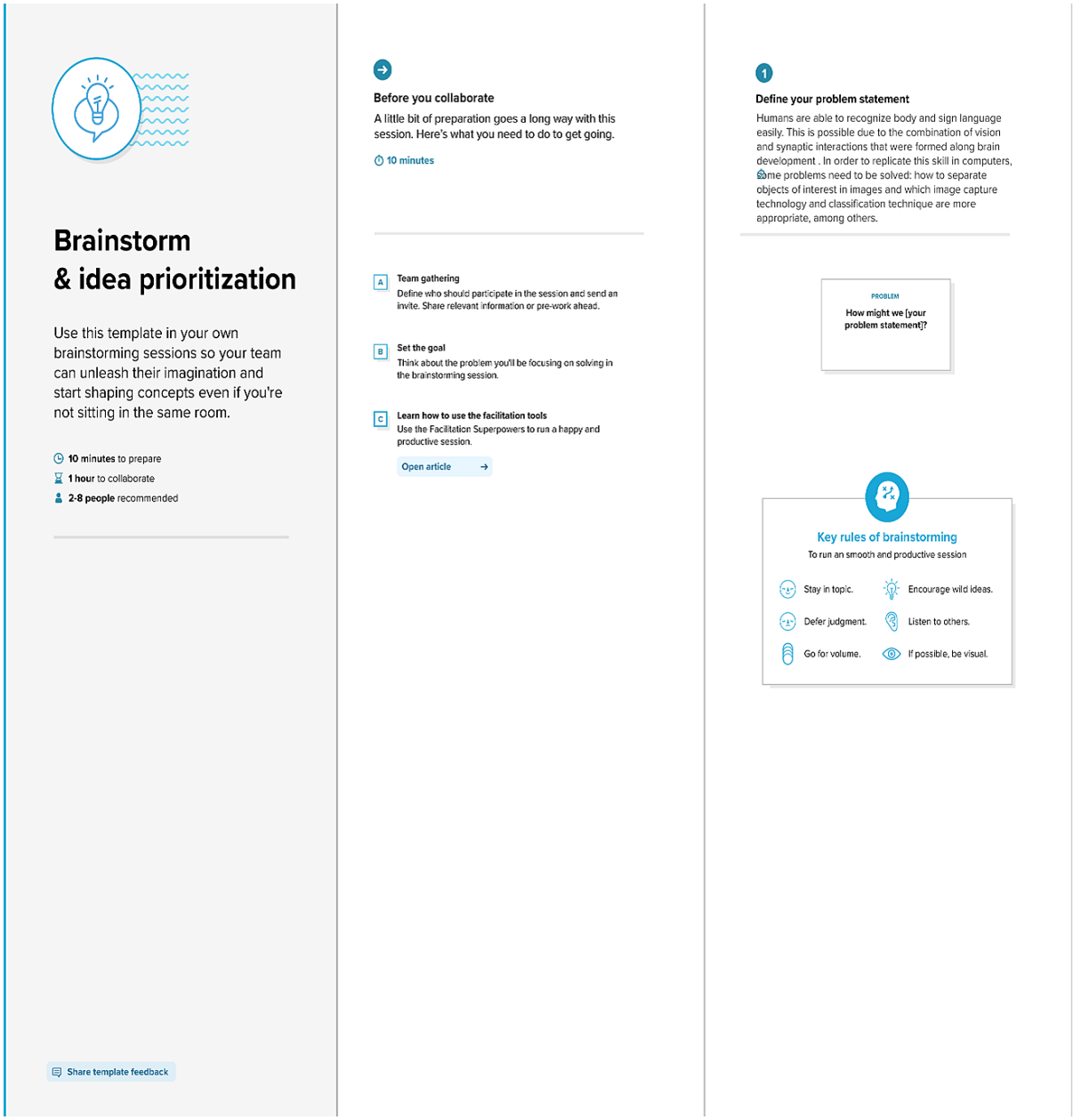


Fig 3.2.1 Brainstroming and Idea Prioritization

Fig 3.2.2 Brainstroming and Idea Prioritization

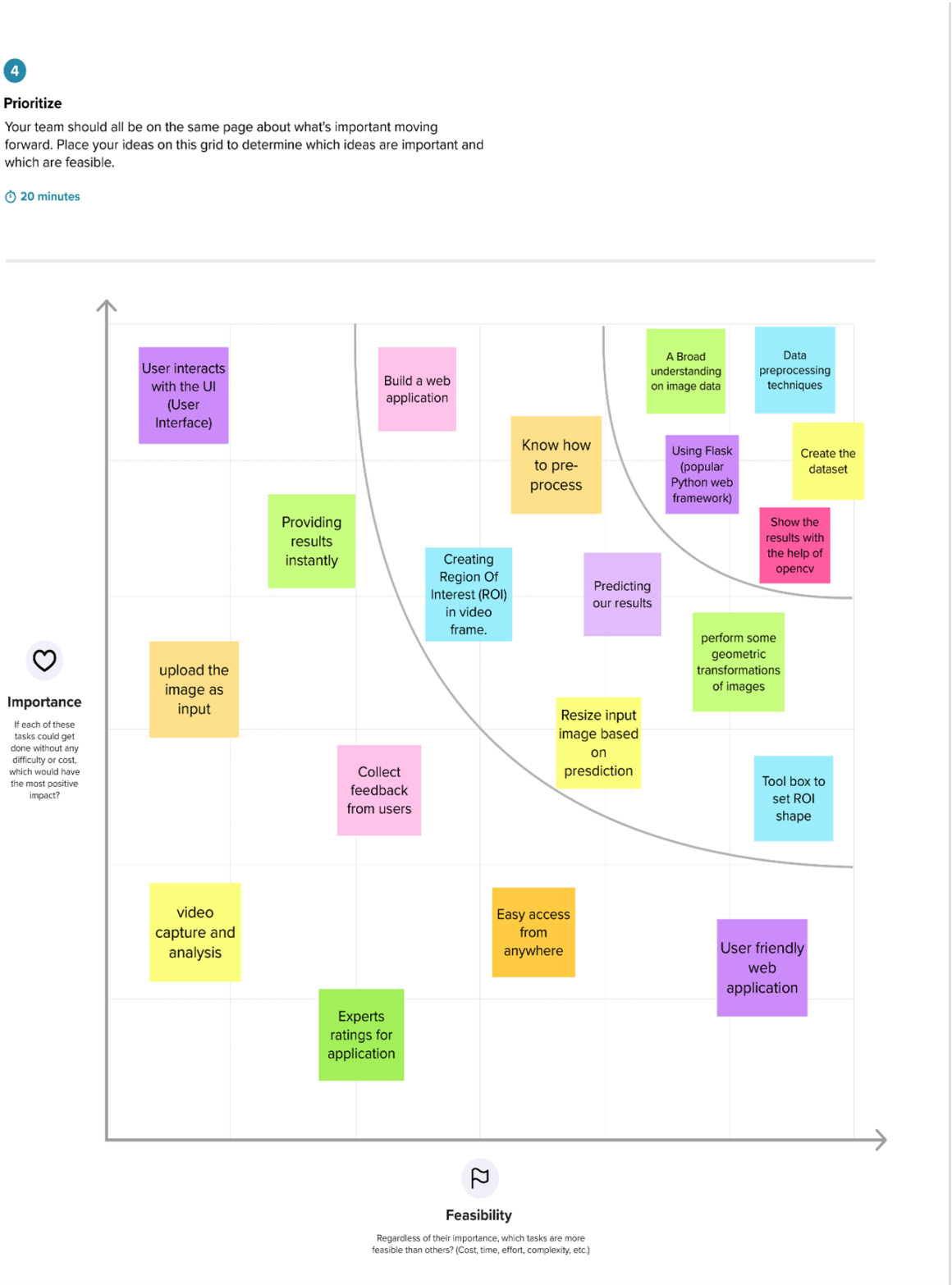


Fig 3.2.3 Brainstroming and Idea Prioritization

3.3 PROPOSED SOLUTION

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Computers to recognize the human signs language easily.Performing some tasks based on a sign by the humans by Browsing through the images obtained using radiology using hand gestures rather than using mouse,keyboard.
2.	Idea / Solution description	We are going to solve this problem using Convolutional Neural Network (CNN) algorithm using Open Source Computer Vision Library (Open CV) which are mainly used for image processing, video capture and analysis with Python Flask framework.
3.	Novelty / Uniqueness	Easy to use and quicker than the existing methods to browse images. The tool does not need the person using it to have an apparatus or any devices. It can be used regardless of the users location.
4.	Social Impact / Customer Satisfaction	Medical industry to browse images for Scans and Surgeries, Industries presenting certain ideas during meetings and used by teachers while teaching.

5.	Business Model (Revenue Model)	This can be extended to other industries like it can be used by presenters, by teachers for show images in the classroom, etc. 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.
6.	Scalability of the Solution	More number of gestures can be added thereby increasing this tool's functionality and useability. Tracking of both hands can be added to increase the set of commands.

3.4 PROBLEM SOLUTION FIT

CUSTOMER SEGMENTS(S)	CUSTOMER LIMITATIONS	AVAILABLE SOLUTIONS (PROS AND CONS)
News image editor, Doctors, Analyst,Designer etc..	The tool can be quite expensive as it requires cameras and other expensive devices to capture images and process it.	The "Gibson" imagebrowser is a 3D visualization medical tool that enables examination of images, such as: MRIs, CT scans and X-rays.

PROBLEMS/ PAINS (ITS FREQUENCY)	PROBLEM ROOT/ CAUSE	BEHAVIORITS INTENSITY
Humans can recognize body and sign language easily. This is possible due to the combination of vision and synaptic interactions thatwere 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 techniqueare more appropriate, among others.	Computers to recognize the human signs language easily.Performing some tasks based on a sign by the humans by Browsing through the images obtained using radiology using hand gestures rather than using mouse,keyboard.	Research for variations in the Hand gestures, Search for the solutions and seek the suggestions on hand variatioms from others.
TRIGGERS TO ACT Takes quite a long time to handle certain process manually.	YOUR SOLUTION We are going to solve this problem using Convolutional Neural Network (CNN) algorithm using Open Source Computer Vision Library (Open CV) which are mainly used for image	CHANNELS OF BEHAVIOR (ONLINE) Social Media , Blogs, Forums.

	processing, video capture and analysis with Python Flaskframework.	
EMOTIONS (BEFORE/AFTER) Before - Large volume of images to be edited so it makesTired andIncapable. After - Relaxed		OFFLINE Friends, Colleagues, and Image Analysts.

CHAPTER - 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Mobile Number Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login using credentials
FR-4	User Dashboard	Home – About the page Introduction – About the technology used and functionality of the application
FR-5	User Launch	Launch for performing actions

4.2 NON-FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Performing certain operations based on actions
NFR-2	Security	Login credentials verification and users can have own private dashboard
NFR-3	Reliability	Handles different types of images and runs continuously on the web server
NFR-4	Performance	Accuracy is high due to high trained and tested data
NFR-5	Availability	Available to users over internet and requires basic browser and camera
NFR-6	Scalability	It can be further extended to any industries for performing action on images using camera

CHAPTER - 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

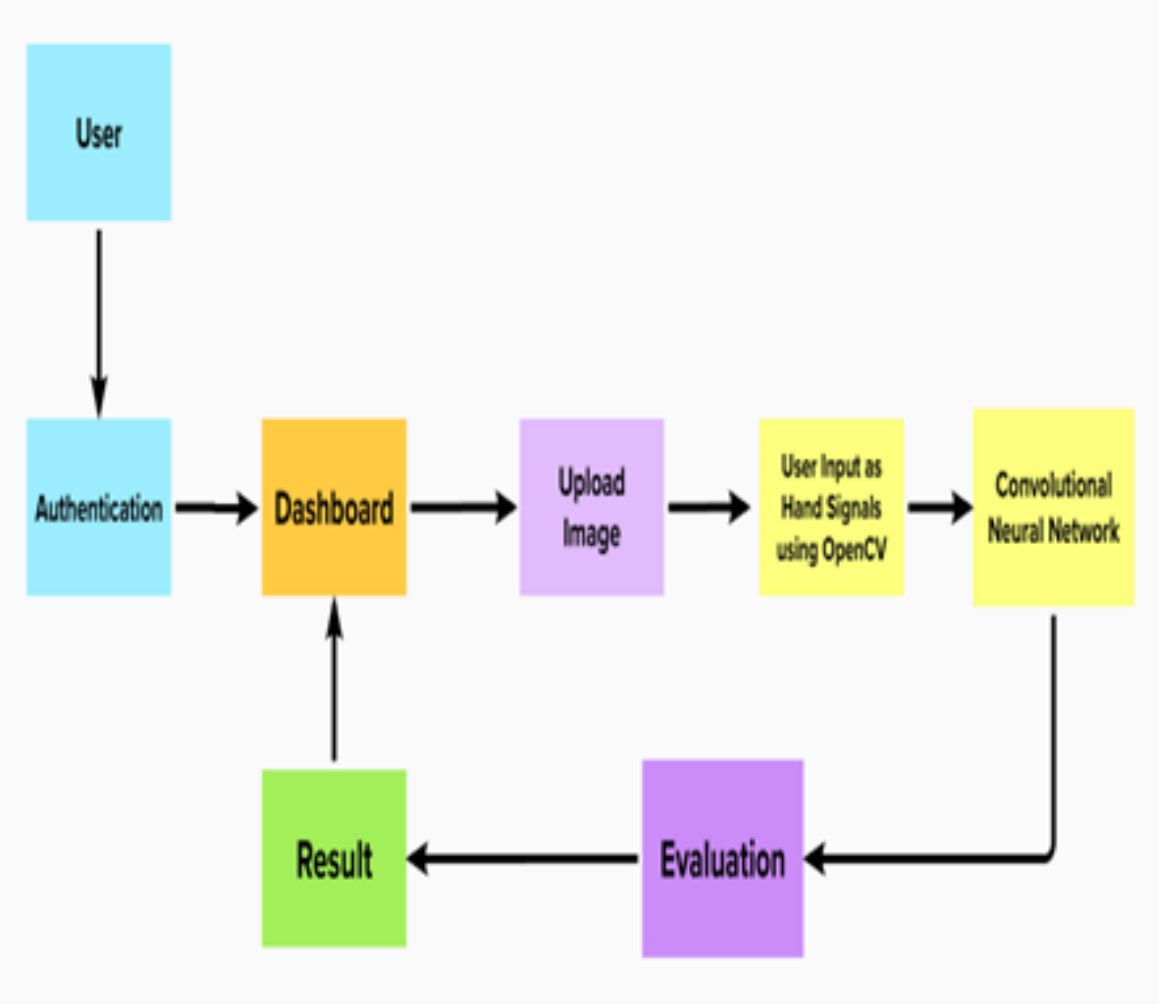


Fig 5.1.1 Data Flow Diagram

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

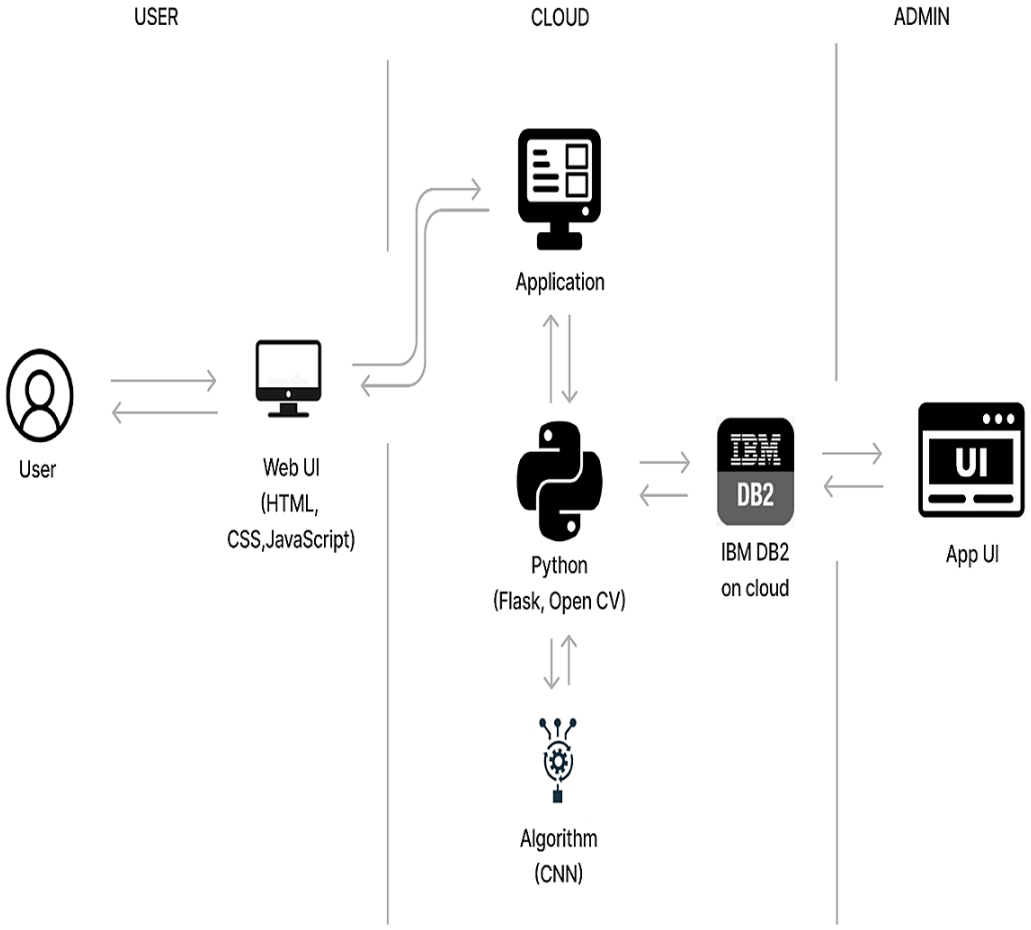


Fig 5.2.1 Technical Architecture

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration/ Login	USN-1	As a user,I can register for the application byentering my email, password, and confirming mypassword	I can access my account / dashboard	High	Sprint-1
	Dashboard	USN-2	Once I enterthe dashboard I can choose Home / Introduction page	I canreceive confirmation email &click confirm	High	Sprint-2
Customer (Organization)	Apply	USN-3	Once I enter the Home page I can input my Hand Signs as inputfor prediction	I canperform multiple sample predictions	Medium	Sprint-1

Administrator		USN-4	As a user,I can get a imagerepresentation of the prediction	I havedifferent forms of output	High	Sprint-1
		USN-5	As a user, I can view the report of my predict ion	I can access details of my prediction	Medium	Sprint-2
		USN-6	As a userI can referthe documentation for support and guidance	I canuse Introduction page for guidance	Medium	Sprint-2
	Forward	USN-7	As a Admin I mustforward the application to the respective companies	The application is received by the companies	High	Sprint-1
	Confirmation	USN-8	Confirmation mail is sent fromthe respective companie	Confirmation is received by the user	High	Sprint-2
	Manage Review	USN-9	As an admin I mustmake the reviews	Reviews appearon the	Low	Sprint-2

			appear on company's profile	company's page		

CHAPTER-6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-1	As a user, I can log into the application by website link	4	High	Goghul JP
		USN-2	As a user, I can log into the application by website link	3	Medium	Kaviraja AS
		USN-3	As a user, I can log into the application by website link	3	Medium	Chandiramohan R
Sprint-2	Dashboard	USN-4	As a user, I can get the details of the application from	4	Medium	Arun GR

			Introduction page			
		USN-5	As a user, I can Perform my Hand signal operations on launch page	4	Medium	Manoj R
Sprint-3	Apply	USN-6	I can upload a input image to get prediction result of my actions	6	High	Kaviraja AS
		USN-7	As a user I want to show my hand actions on the region of interest through camera	8	High	Chandiramohan R
Sprint-4	Result	USN-8	As a user I needed to get the actual output for my actions performed	8	High	Arun GR

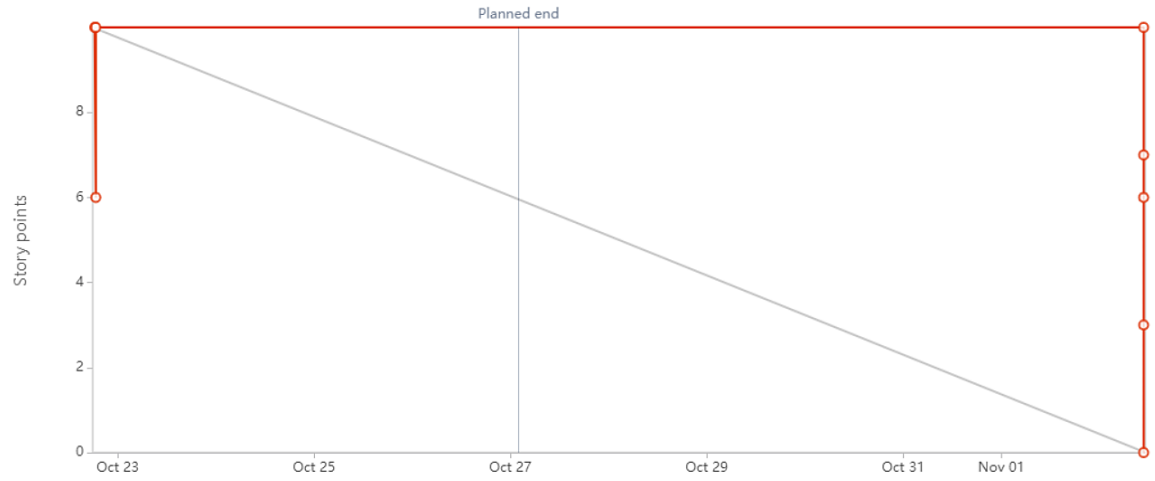
		USN-9	As a user I get the result as resized input image (blur,resize,flip) based on my hand signals	9	High	Manoj R

6.2 SPRINT DELIVERY SCHEDULE

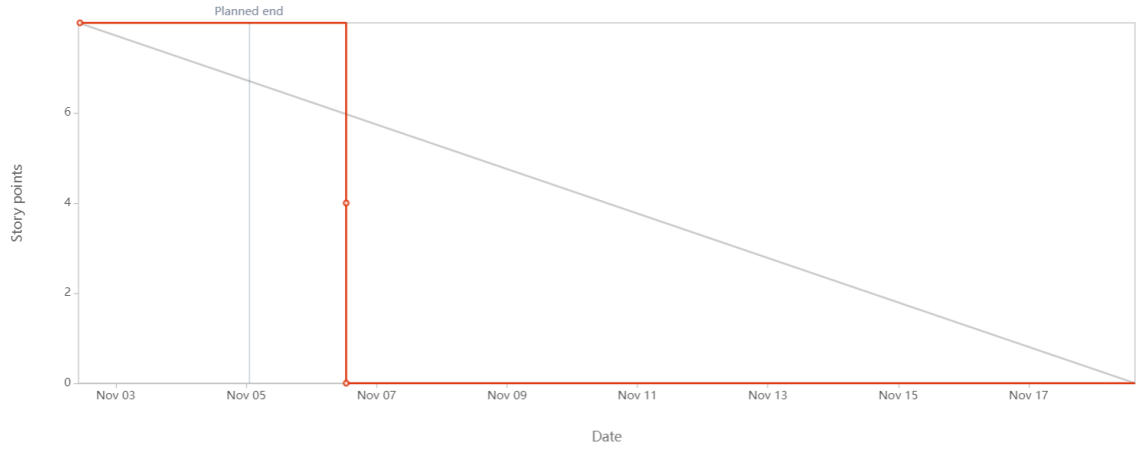
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	10	6 Days	22 Oct 2022	27 Oct 2022	10	28 Oct 2022
Sprint-2	8	6 Days	31 Oct 2022	05 Nov 2022	8	06 Nov 2022
Sprint-3	14	6 Days	07 Nov 2022	12 Nov 2022	14	13 Nov 2022
Sprint-4	17	6 Days	14 Nov 2022	19 Nov 2022	17	18 Nov2022

6.3 REPORT FROM JIRA

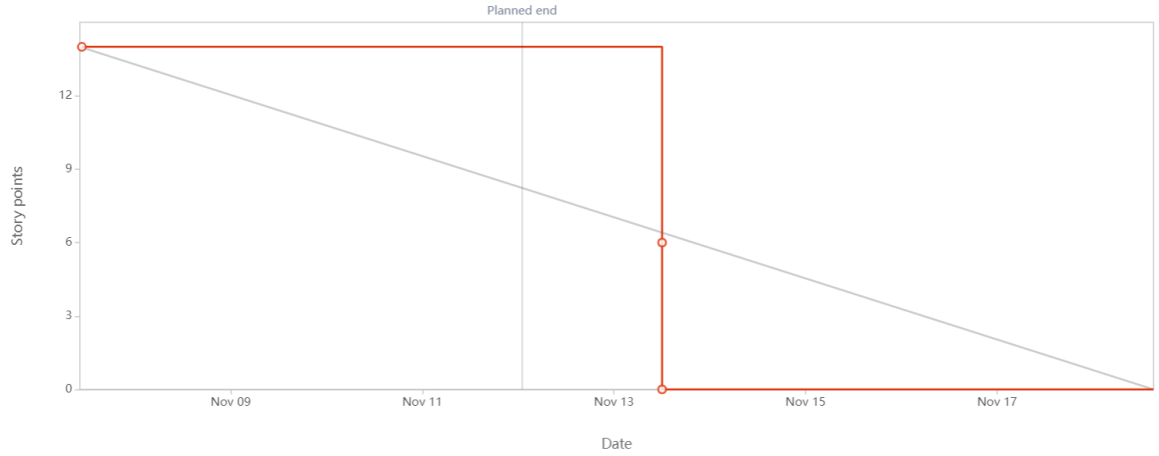
SPRINT 1



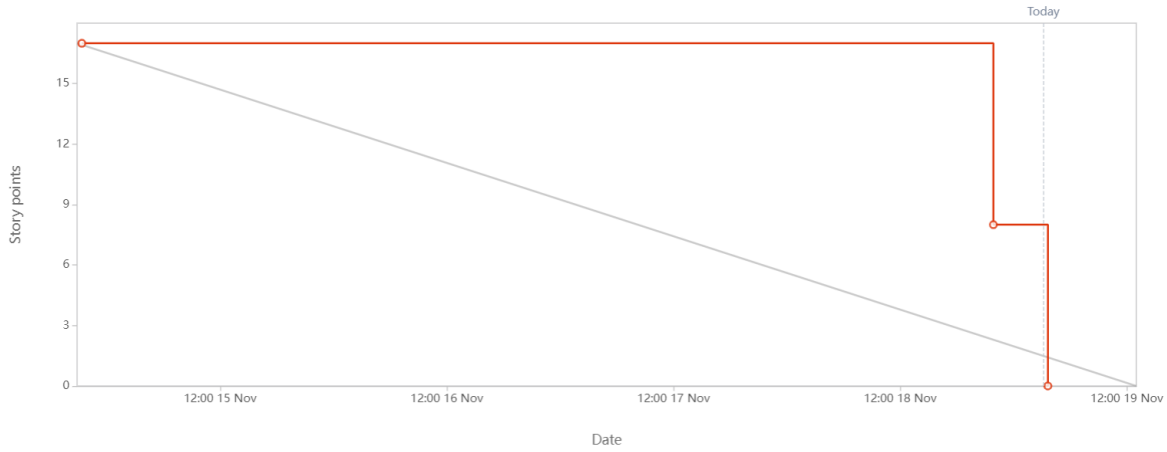
SPRINT 2



SPRINT 3



SPRINT 4



CHAPTER - 7

CODING & SOLUTIONING

7.1 FEATURE 1

We found that many hospitals rely on mouse and keyboard to browse the images that are obtained during different surgeries, scans, etc. This can contaminate the environment with various infections thus compromising the sterility.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.

7.2 FEATURE 2

In this research, we propose a novel approach for that is based on concepts and techniques of Convolutional Neural Network (CNN). In this we will be improving the data that supresses unwilling distortions or enhances some image features important for further processing, although perform some geometric transformations of images like rotation, scaling, translation etc.

CHAPTER - 8

TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result
HomePage_TC_OO2	Functional	Home Page	Verify the UI header element in home page		1.Enter URL 2.Check whether the home page is rendered 3.Click on the header to see if it lands to other page	https://127.0.0.1:5000/	Details of smartinternz project page appears
DemoPage_TC_OO1	UI	Demo page	Verify user is able to see and interact with demo page		1.Enter URL and click on demo 2.Check whether the demo page is rendered	https://127.0.0.1:5000/demo https://127.0.0.1:5000/	Demo page should display
DemoPage_TC_OO2	Functional	Demo page	Verify user is able to watch the demo video and manipulate as a normal video player		1.Enter URL(https://127.0.0.1:5000/) and click demo 2.Check whether the demo page is rendered 3.Click on the demo video to play it 4.Check whether the video plays 5.Check whether the webpage behaves like normal video	https://127.0.0.1:5000/demo https://127.0.0.1:5000/	Demo video should run and the video should be manipulated as a normal video player
LaunchPage_TC_OO1	UI	Launch page	Verify user is able to see and interact with launch page		1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch	Launch page should display
LaunchPage_TC_OO2	Functional	Launch page	Verify user is able to upload the image to be manipulated		1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered 3.Click on the file upload icon and upload the image 4.Click on the manipulate button to start manipulating the	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch Image to be manipulated	Application should show upload the image to be manipulated and manipulation operation should begin
aunchPage_TC_OO	Functional	Launch page	Verify user is able to resize the image to 400x400 size	Image needs to uploaded which is going to be manipulated	1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered 3.Click on the file upload icon and upload the image 4.Click on the manipulate button to start manipulating the image 5.Show 4 (four fingers) infront of camera as input 6.Check whether the image is resized into 400x400	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch Image to be manipulated	Application should show resized image of 400x400
aunchPage_TC_OO	Functional	Launch page	Verify user is able to blur the image	Image needs to uploaded which is going to be manipulated	1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered 3.Click on the file upload icon and upload the image 4.Click on the manipulate button to start manipulating the image 5.Show 3 (three fingers) infront of camera as input 6.Check whether the image is blurred	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch Image to be manipulated	Application should show blurred image
aunchPage_TC_OO	Functional	Launch page	Verify user is able to resize the image to 200x200 size	Image needs to uploaded which is going to be manipulated	1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered 3.Click on the file upload icon and upload the image 4.Click on the manipulate button to start manipulating the image 5.Show 1 (one finger) infront of camera as input 6.Check whether the image is resized into 200x200	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch Image to be manipulated	Application should show resized image of 200x200
LaunchPage_TC_O	Functional	Launch page	Verify user is able to rotate the image 45 degree right	Image needs to uploaded which is going to be manipulated	1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered 3.Click on the file upload icon and upload the image 4.Click on the manipulate button to start manipulating the image 5.Show 2 (two fingers) infront of camera as input 6.Check whether the image is turned 45 degree to right	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch Image to be manipulated	Application should show image turned 45 degree right
LaunchPage_TC_OO7	Functional	Launch page	Verify user is able to convert the image to grayscale	Image needs to uploaded which is going to be manipulated	1.Enter URL(https://127.0.0.1:5000/) and click launch 2.Check whether the launch page is rendered 3.Click on the file upload icon and upload the image 4.Click on the manipulate button to start manipulating the image 5.Show 5 (five fingers) infront of camera as input 6.Check whether the image is converted to grayscale	https://127.0.0.1:5000/ https://127.0.0.1:5000/launch Image to be manipulated	Application should show grayscaled image

8.2 USER ACCEPTANCE TESTING

8.2.1 Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the A Gesture-based Tool for Sterile Browsing of Radiology Image project at the time of the release to User Acceptance Testing (UAT).

8.2.2 Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved. Defect analysis is usually performed with help of knowledge learned and gained from defects that were discovered previously.

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
By Design	6	4	3	5	18
Duplicate	1	2	1	0	4
External	2	3	0	2	7
Fixed	11	2	3	13	29
Not Reproduced	0	1	1	1	3
Skipped	0	0	2	2	4
Won't Fix	0	4	2	2	8
Totals	20	16	12	25	73

8.2.3 Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested. The Test Analysis Report records results of the tests., presents the capabilities and deficiencies for review, and provides a means of assessing software progression to the next stage of development or testing.

Section	TotalCases	Not Tested	Fail	Pass
Print Engine	9	0	0	9
Client Application	45	0	5	40
Security	5	0	1	4
Outsource Shipping	3	0	0	3
Exception Reporting	8	2	1	5
Final Report Output	10	1	2	7
Version Control	3	0	0	3

CHAPTER - 9

RESULT

9.1 PERFORMANCE METRICS

S.No.	Parameter	Values	Screenshot

1.	Model Summary	-	`model.summary()#summary of our model`
			`Model: "sequential"`
			`_____`
			`Layer (type) Output Shape Param #`
			`=====`
			`conv2d (Conv2D) (None, 62, 62, 32) 320`
			`_____`
			`max_pooling2d (MaxPooling2D) (None, 31, 31, 32) 0`
			`_____`
			`conv2d_1 (Conv2D) (None, 29, 29, 32) 9248`
			`_____`
			`max_pooling2d_1 (MaxPooling2 (None, 14, 14, 32) 0`
			`_____`
			`flatten (Flatten) (None, 6272) 0`
			`_____`
			`dense (Dense) (None, 512) 3211776`
			`_____`
			`dense_1 (Dense) (None, 6) 3078`
			`=====`
			`Total params: 3,224,422`
			`Trainable params: 3,224,422`
			`Non-trainable params: 0`
			`_____`

2.	Accuracy	Training Accuracy - 0.9966 Validation Accuracy - 0.9667	Epoch 13/25 198/198 [==========] - 14s 70ms/step - loss: 0.0864 - accuracy: 0.9731 - val_loss: 0.2522 - val_accuracy: 0.9667 Epoch 14/25 198/198 [==========] - 15s 74ms/step - loss: 0.0518 - accuracy: 0.9781 - val_loss: 0.3543 - val_accuracy: 0.9333 Epoch 15/25 198/198 [==========] - 13s 66ms/step - loss: 0.0781 - accuracy: 0.9764 - val_loss: 0.3312 - val_accuracy: 0.9667 Epoch 16/25 198/198 [==========] - 13s 66ms/step - loss: 0.0487 - accuracy: 0.9832 - val_loss: 0.2124 - val_accuracy: 0.9667 Epoch 17/25 198/198 [==========] - 13s 66ms/step - loss: 0.0385 - accuracy: 0.9865 - val_loss: 0.2988 - val_accuracy: 0.9667 Epoch 18/25 198/198 [==========] - 13s 66ms/step - loss: 0.0573 - accuracy: 0.9832 - val_loss: 0.2565 - val_accuracy: 0.9667 Epoch 19/25 198/198 [==========] - 13s 66ms/step - loss: 0.0281 - accuracy: 0.9865 - val_loss: 0.2607 - val_accuracy: 0.9667 Epoch 20/25 198/198 [==========] - 13s 66ms/step - loss: 0.0501 - accuracy: 0.9848 - val_loss: 0.5327 - val_accuracy: 0.8667 Epoch 21/25 198/198 [==========] - 15s 74ms/step - loss: 0.0513 - accuracy: 0.9832 - val_loss: 0.3753 - val_accuracy: 0.9667 Epoch 22/25 198/198 [==========] - 13s 67ms/step - loss: 0.0369 - accuracy: 0.9882 - val_loss: 0.3396 - val_accuracy: 0.9667 Epoch 23/25 198/198 [==========] - 13s 68ms/step - loss: 0.0276 - accuracy: 0.9916 - val_loss: 0.4354 - val_accuracy: 0.9667 Epoch 24/25 198/198 [==========] - 15s 74ms/step - loss: 0.0752 - accuracy: 0.9697 - val_loss: 0.7543 - val_accuracy: 0.8667 Epoch 25/25 198/198 [==========] - 13s 67ms/step - loss: 0.0489 - accuracy: 0.9798 - val_loss: 0.2213 - val_accuracy: 0.9667 <keras.callbacks.History at 0x7f5cb5795dd0>

CHAPTER - 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES

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

DISADVANTAGES

The tool can be quite expensive as it requires cameras and other expensive devices to capture images and process it. Such systems are difficult to develop because of their complexity and their cost of implementation. As each gesture is assigned a specific control command, this system is not platform independent since certain control commands vary as the operating system varies.

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

The tool can be made quicker by increasing the recognition speed. More number of gestures can be added thereby increasing this tool's functionality and useability for different purposes. Tracking of both hands can be added to increase the set of commands. Voice commands can also be added to further increase the functionality. The tool can be made to add multiple images to be uploaded and predict the output for the particular image. For that, user can choose any image from the multiple images and apply the prediction for the respective image for manipulation.

CHAPTER - 13

APPENDIX

13.1 SOURCE CODE

```
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our
application.
#request-for accessing file which was uploaded by the user on our application.
import operator
import cv2 # opencv library
import numpy as np

from tensorflow.keras.models import load_model#to load our trained model
import os
from werkzeug.utils import secure_filename

app = Flask(__name__,template_folder="templates") # initializing a flask app
# Loading the model
model=load_model('gesture.h5')
print("Loaded model from disk")



@app.route('/')# route to display the home page
def home():
return render_template('home.html')#rendering the home page
```

```
@app.route('/intro') # routes to the intro page
def intro():
return render_template('intro.html')#rendering the intro page

@app.route('/image1',methods=['GET','POST'])# routes to the index html
def image1():
return render_template("launch.html")


@app.route('/predict',methods=['GET', 'POST'])# route to show the predictions
in a web UI
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() #capturing the video frame values
# Simulating mirror image
frame = cv2.flip(frame, 1)
```

```
# Got this from collect-data.py
# Coordinates of the ROI
x1 = int(0.5*frame.shape[1])
y1 = 10
x2 = frame.shape[1]-10
y2 = int(0.5*frame.shape[1])
# Drawing the ROI
# The increment/decrement by 1 is to compensate for the bounding box
cv2.rectangle(frame, (x1-1, y1-1), (x2+1, y2+1), (255,0,0) ,1)
# Extracting the ROI
roi = frame[y1:y2, x1:x2]

# Resizing the ROI so it can be fed to the model for prediction
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]}
# Sorting based on top prediction
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, (11, 11), 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(1)
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)
```

13.2 GITHUB & PROJECT DEMO LINK

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-38487-1660381510

VIDEO LINK:

https://www.youtube.com/embed/3zzl7ZzdAkY

CHAPTER - 14

REFERENCES

[1] Mewes, A., Saalfeld, P., Riabikin, O. et al. A gesture-controlled projection display for CT-guided interventions. Int J CARS 11, 157–164 (2016).

[2] N. Madapana, G. Gonzalez, L. Zhang, R. Rodgers and J. Wachs, "Agreement Study Using Gesture Description Analysis," in IEEE Transactions on Human-Machine Systems, vol. 50, no. 5, pp. 434-443, Oct. 2020, doi: 10.1109/THMS.2020.2992216.

[3] S. Ameur, A. B. Khalifa and M. S. Bouhlel, "A comprehensive leap motion database for hand gesture recognition," 2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2016, pp. 514-519, doi: 10.1109/SETIT.2016.7939924

[4] N. Nestorov, P. Hughes, N. Healy, N. Sheehy and N. O'Hare, "Application of Natural User Interface Devices for Touch-Free Control of Radiological Images During Surgery," 2016 IEEE 29th International Symposium on Computer-Based Medical Systems (CBMS), 2016, pp. 229-234, doi: 10.1109/CBMS.2016.20.

[5] Ebert LC, Hatch G, Ampanozi G, Thali MJ, Ross S. You Can’t Touch This: Touch-free Navigation Through Radiological Images. Surgical Innovation. 2012;19(3):301-307. doi:10.1177/1553350611425508

[6] Pauchot J, Di Tommaso L, Lounis A, et al. Leap Motion Gesture Control With Carestream Software in the Operating Room to Control Imaging: Installation Guide and Discussion. Surgical Innovation. 2015;22(6):615-620. doi:10.1177/1553350615587992

[7] L.C. Ebert, P.M. Flach, M.J. Thali, S. Ross, Out of touch – A plugin for controlling OsiriX with gestures using the leap controller,Journal of Forensic

Radiology and Imaging,Volume 2, Issue 3,2014,Pages 126-128,ISSN 2212-4780,https://doi.org/10.1016/j.jofri.2014.05.006.

[8] Justin H. Tan , Cherng Chao, Mazen Zawaideh, Anne C. Roberts, Thomas B. Kinney, Informatics in Radiology: Developing a Touchless User Interface for Intraoperative Image Control during Interventional Radiology Procedures, Volume 33, No 2, https://doi.org/10.1148/rg.332125101