LAB REPORT

Submitted by

Sidharth Bakshi
[RA2011003010577]

Under the Guidance of

Ms. G.K. Sandhia

Assistant Professor (Sr. G), Department of Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR - 603203
JUNE 2022

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this lab report titled **"GYMverse "** is the bonafide work done by Sidharth Bakshi (RA2011003010577) who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Ms. G.K. Sandhia

SEPM – Course Faculty

Assistant Professor (Sr. G)

Department of Computing Technologies

TABLE OF CONTENTS

ABSTRACT

In today's world scenario where fitness is considered to be the topmost priority of every person and combining technology to fitness is the new way of working out. And when we see today's world condition where people have to stay at their homes and operate the things, gym and fitness is becoming more and more home centric.

So here we are with our GYMverse ,it is a new level of technology in which our application will use A.I. to monitor you while you workout. our app will let you correct your postures while you work out

This app will be a boom for people who loves to workout but did not find time to go to gym

and moreover it will help people to monitor themselves and see what they are doing wrong and what they are doing correctly.

LIST OF FIGURES

LIST OF ABBREVIATIONS

API	APPLICATION PROGRAMMING INTERFACE
HTML	HYPER TEXT MARKUP LANGUAGE
CSS	CASCADING STYLES SHEET
JS	JAVASCRIPT
RAM	RANDOM ACCESS MEMORY
UI	USER INTERFACE
UX	USER EXPERIENCE
IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
WBS	WORK BREAKDOWN STRUCTURE
SWOT	STRENGTH, WEAKNESS, OPPURTUNITIES AND THREATS
RMM	RISK MONITERING AND MANAGEMENT
UML	UNIFIED MODELING LANGUAGE
JSON	JAVASCRIPT OBJECT NOTATION
ER	ENTITY RELATION
DFD	DATA FLOW DIAGRAM
HTTP	HYPER TEXT TRANSFER PROTOCOL
XML	EXTENSIBLE MARKUP LANGUAGE
CRUD	CREATE, READ, UPDATE, DELETE
AJAX	ASYNCHRONOUS JAVASCRIPT AND XML
DB	DATABASE
CLI	COMMAND LINE INTERFACE
SQL	STRUCTURED QUERY LANGUAGE

1. PROBLEM STATEMENT

Aim

To Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the 'GYMversei' project.

Team Members:

Sl No	Register No	Name	Role
1	RA2011003010568	Mukund Umashankar	Developer
2	RA2011003010577	Sidharth Bakshi	Developer
3	RA2011003010561	Saiteja Nagishetty	Designer

Project Title: Gymverse

Project Description:

A platform for learners to discover learning materials available on the internet

Business Case

DATE	09/03/2022
SUBMITTED BY	Mukund Umashankar, Saiteja Nagishetty ,Sidharth Bakshi
TITLE / ROLE	Developer

THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

To provide a community space for students and working professionals:
1) to develop a platform in which to help people do gym exercises properly
2) to educate users about the correct exercising postures
3) to share other resources regarding better lifestyle

THE HISTORY

In bullet points, describe the current situation.

1) Our aim is to develop a platform in which to help people do gym exercises correctly and know about the correct postures
2) This platform will create a metaverse in which people can share the realtime video of doing workouts and our AI will tell them if they are doing it correctly or the posture is correct or not.
3) Our platform also aims at telling people about different muscles and specific exercises and how to do those exercises.

LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

1) In our project people don't need expensive equipment, people will only need a laptop or simply their mobile phones with a web camera.
2) the only thing that can lack is bad weather when the internet is not working properly, but soon will be able to provide videos to download after our project gets a good response

APPROACH

List what is needed to complete the project.

1) tensorflow
2) matplotlib
3) sklearn
4) opencv
5) python
6) numpy
7) pandas

BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

1) The project will help people to understand the benefits of gym and the need to keep themselves fit and
2) the importance of a workout that can shape their life in a good and healthy way. also to know the correct way to doing their workout and not hurting themselves while doing so.

Result

Thus, the project team formed, the project was described, the business case was prepared and the problem statement was arrived at.

2. STAKEHOLDERS & PROCESS MODELS

Aim

       To identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

Project Title: Gymverse

1. Executive Summary

1.1 Project Methodology:

Agile Methodology is a people-focused, results-focused approach to software development that respects our rapidly changing world.
It's centred around adaptive planning, self-organization, and short delivery times.
It's flexible, fast, and aims for continuous improvements in quality, to put in simple terms, Agile helps teams in delivering value to customers quickly and effortlessly.
Thus, developing such a software management project, Agile methodology brings out the most effective growth out of it.

1.2 Stakeholder Identification:

Internal stakeholders:

       The internal stakeholders include the team members, managers, executives who are all internally related.

Project Role	Responsibilities	Team members assigned to
Project Manager	Managing the project	Saiteja Nagishetty
Technical Lead	Leading the development	Mukund Umashankar
Business Analyst	Analysing business opportunities	Sidharth Bakshi
Developer	Developing the platform	Mukund Umashankar
Tester	Testing the platform	Sidharth Bakshi

External stakeholders:

Project Role	Responsibilities	Team members assigned to
Customer	Use and grow the platform	Sayantan Ganguly
Customer	Use and grow the platform	Abhinay Ponaganti
Creditor	Providing funds for the project	Parents
Supplier	Providing resources	SRMIST
Shareholder	Monitor direction of the project	Mukund Umashankar
Shareholder	Monitor direction of the project	Saiteja Nagishetty

2. Stakeholder Management

Interest and Influence matrix

Interest	Influence
High	High
Low	Low
Low	High
High	Low

STAKEHOLDER INTEREST, INFLUENCE, PRIORITY IDENTIFICATION

Stakeholder	Responsibility	Interest	Influence	Estimated Priority
Owner	Making important decisions	High	High	1
Sponsor	Providing funds	Low	High	2
Team members	Design, develop and deploy the project	High	High	1
Project Manager	To supervise and manage the project	High	High	1
Investors	To invest in the project	Low	High	2
Resource Manager	To manage resources for the project	Low	Low	3
Suppliers	To supply resources for the project	High	High	2
End Users	Use and grow the platform	High	Low	4

Result

Thus the Project Methodology was identified and stakeholders were described.

3. IDENTIFYING REQUIREMENTS

Aim

To identify the system, functional and non-functional requirements for the project.

Project Title: Gymverse

User Requirements

Our cross-device platform will allow users to discover and share learning resources on any topics, get recommendations/ best-practices, interact with other learners, etc.

1. A content-creator can post his/her content on our platform for better reach
2. A student can search for the best and most up-to-date learning resources available on the internet
3. A more experienced person can share his/her experience and recommendation to help budding learners avoid mistakes and pitfalls.

System Requirements

SR1: A stable internet connection

SR2: A modern browser supporting latest HTML, CSS and JS features

SR3: A relatively modern Intel/AMD processor

SR4: Minimum 4GB of RAM

SR5: Webserver to deploy and run the platform

Functional Requirements

FR1. The platform shall have a search bar to search about a particular topic. **Must Have**

FR2. The platform shall have a user login/signup page to save posts made by others or post their content. **Must Have**

FR3. The platform shall have an intuitive UI to make it easy for anyone to use. **Should Have**

FR4. The platform shall have a rating/popularity-based sorting of posts to give better suited content. **Should Have**

FR5. The platform shall have an AI based categorization of posts for better discovery. **Could Have**

FR6. The platform shall have an interactive space for fellow GYMbros to interact. **Could Have**

Non-Functional Requirements

NFR1. The platform should be scalable to support more concurrent users in future. **Must Have**

NFR2. The platform should be available 24x7 throughout the year. **Must Have**

NFR3. The platform should not lose any posts and its interaction data. **Must Have**

NFR4. The platform should login users within 5 seconds. **Should Have**

NFR5. The platform should display posted content within 10 seconds. **Should Have**

NFR6. The platform should reflect post interaction within 30 seconds. **Should Have**

Result
      Thus, the requirements were identified and accordingly described.

4. PROJECT PLAN & EFFORT

Aim

To Prepare Project Plan based on scope, Calculate Project effort based on resources, Find Job roles and responsibilities

Requirements

1. Project Management Plan

Describe the key issues driving the project. **[Min 3 Focus Areas]**

Focus Area	Details
Schedule Management	Milestones: 1) Understanding project: Identifying goals and approach to the project. 2) Identification of Requirement: Function, Non-functional and User Requirements 3) Project Management: Cost estimation, Risk Analysis, Project Timeline 4) Designing: Designing UI, review and revising 5) Developing Front-end: Coding the front-end part 6) Developing Back-end: Coding the functional elements, creating database, debugging and testing 7) Deployment: Deploying the website and giving it a domain. Schedule Control:

Task	Start	End
Understanding project	07/03/2022	17/03/2022
Identifying requirements	17/03/2022	30/03/2022
Project Management	31/03/2022	10/04/2022
Designing	11/04/2022	30/04/2022
Developing front-end	01/05/2022	15/06/2022
Developing Back-end	16/06/2022	31/07/2022

	Deployment	1/08/2022	10/08/2022

Stakeholder	Internal Stakeholders:
	   1) Project Manager
	   2) Technical Lead
	   3) Business Analyst
	   4) Developer
	   5) Tester
	External Stakeholder:
	   1) Customer
	   2) Supplier
	   3) Shareholder
	   4) Creditor

Risk Management	Risks Identified:
	   1) System crash
	   2) Corrupt Code
	   3) Server breakdown
	   4) Other commitments
	   5) Lack of initial interest in the project
	   6) Lack of commitment from developer

2. Estimation

Effort and Cost Estimation

Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
Design the user interface	User Requirement Confirmation	Confirm the user requirements (acceptance criteria)	3	1500
	Designing	Designing the UI	50	25000
	Review	Reviewing the UI for final adjustments	10	5000
Developing front end	Develop basic skeleton	Go through designed UI and code the basic skeleton of the platform	70	35000
	Add UI elements	Add more complex UI elements and make	100	50000

		website responsive and dynamic.		
Developing back end	Make UI interactive	Make the UI elements interactive	30	15000
	Develop core functionalities	Implement core functionalities of the platform	110	55000
	Integrate with DBMS	Integrate the platform with a DBMS system for data storage	10	5000
	Debug	Extensively test and debug the platform	20	10000
Deployment	Test run	Deploy it in open-beta for user and developers to test out the platform in real world scenarios	10	5000
	Deploy publicly	Make final changes as per the test run results and deploy it stably for public	10	5000

Effort (hr)	Cost (INR)
1	500

Infrastructure/Resource Cost [CapEx]

< One-time Infra requirements >

Infrastructure Requirement	Qty	Cost per qty	Cost per item
Laptop	3	50000	150000
Smartphone	3	20000	60000
Network Connection	3	2000	6000

2.3 Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin Developer, Support Consultant	3	2,000,000	6,000,000
License	Operating System Database Middleware IDE	10	10000	100,000
Infrastructures	Server, Storage and Network	20	20000	400,000

3. Project Team Formation

Identification Team members

Name	Role	Responsibilities
Mukund Umashankar	Key Business User (Product Owner)	Provide clear business and user requirements
Saiteja Nagishetty	Project Manager	Manage the project
Sidharth Bakshi	Business Analyst	Discuss and Document Requirements
Mukund Umashankar	Technical Lead	Design the end-to-end architecture
Saiteja Nagishetty	UX Designer	Design the user experience
Saiteja Nagishetty	Frontend Developer	Develop user interface
Mukund Umashankar	Backend Developer	Design, Develop and Unit Test Services/API/DB
Sidharth Bakshi	Cloud Architect	Design the cost effective, highly available and scalable architecture
Mukund Umashankar	Cloud Operations	Provision required Services
Sidharth Bakshi	Tester	Define Test Cases and Perform Testing

Responsibility Assignment Matrix

RACI Matrix	Team Members			
Activity	Saiteja Nagishetty (BA)	Mukund Umashankar (Developer)	Sidharth Bakshi(Project Manager)	Key Business User
User Requirement Documentation	A	C/I	R	R
Project Management	R	C/I	A	I
Designing	I	A	R	C/I
Development	I	A	R	I
Deployment	C/I	A	R	I

A	Accountable
R	Responsible
C	Consult
I	Inform

Reference

1. https://www.pmi.org/
2. https://www.projectmanagement.com/

Result:
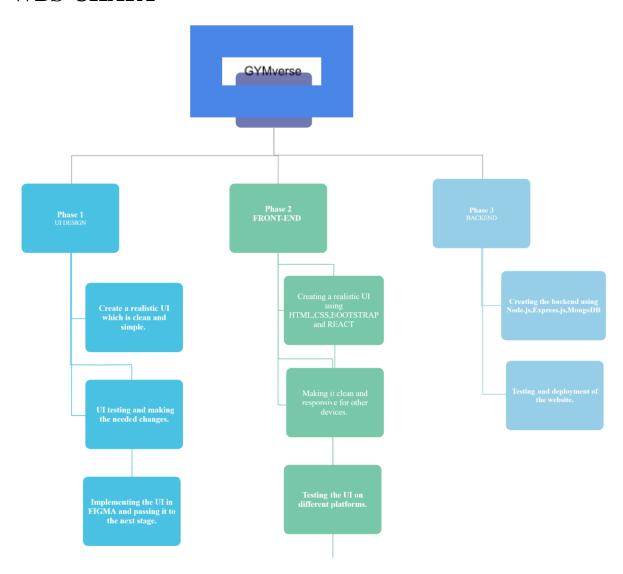
Thus, the Project Plan was documented successfully.

5. WORK BREAKDOWN STRUCTURE & RISK ANALYSIS

Aim

To Prepare Work breakdown structure, Timeline chart, and Risk identification table

WBS-CHART

WORK-BREAKDOWN STRUCTURE:

1. Gather Requirements

    Technical specifications

        Expected bandwidth

        User registration

        Restricted areas

    User requirements

        Menu navigation

        Interactive modules

        Static pages

        Flash elements

    Reporting requirements

        Bandwidth & usage

        Page views

        Session length

2. Establish Design

    Design elements

        Banner

        Footer

        Logo

        Color scheme

        Font usage

    Overall layout

Column setup

Optional modules

Navigation layout

Content elements

About page

Contact page

Services page

FAQ page

Photo Gallery

3. Select Technical Framework

Evaluate options against requirements

Evaluate cost and time to develop

Make decision

4. Implement Technical Framework

Build or acquire back end

Build or acquire front end (user interface)

Integrate back end and front end

5. Create Content

Create content summary

Establish content details

Assign content creation

Create detailed content

6. Load Content 7. Test Site

Navigation

Interactive elements

Contact form

User registration

Browser compatibility

8. Roll Out Site

Establish target date
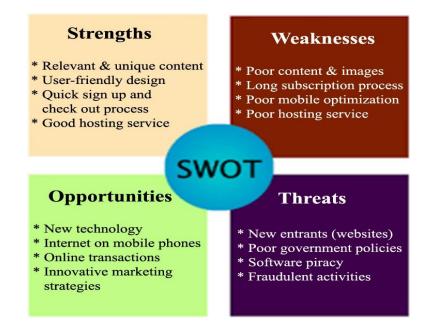
Create communication plan

Make site live

Time - Gantt chart

Result:

      Thus, the work breakdown structure with timeline chart and risk table were formulated successfully.

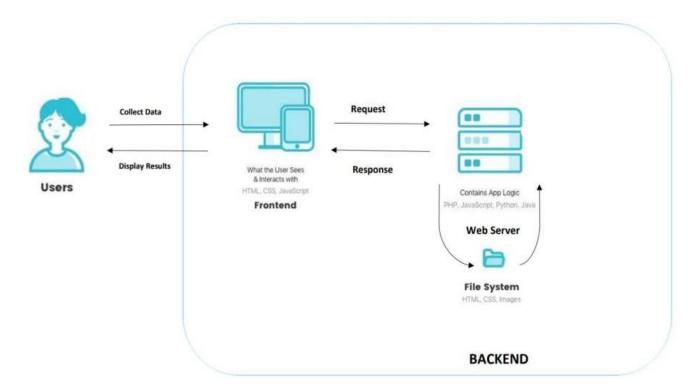# 6. SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM

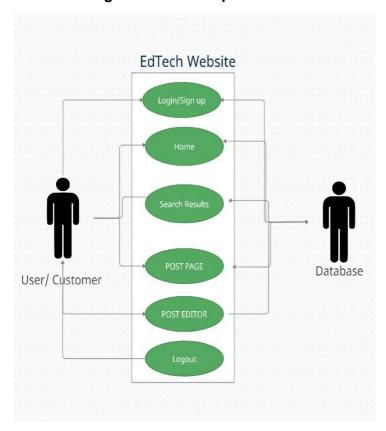**Aim**

To prepare architecture and design of the system

Software Used

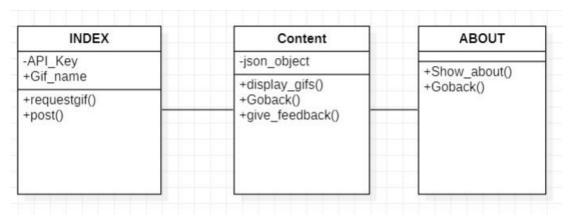**Star UML,** Rational Rose, Etc…

Architecture Diagram with description

Use Case Diagram with Description

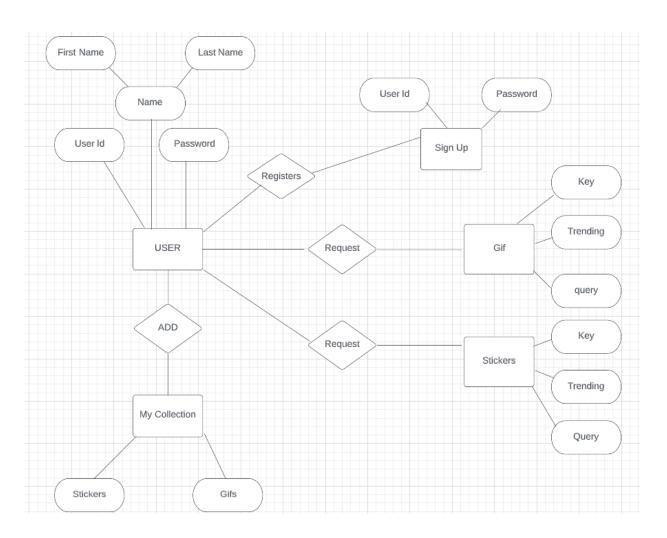**Class Diagram (Applied For OOPS based Project)**

Result:

Thus, architecture and design of the system was documented successfully.

7. ENTITY RELATIONSHIP DIAGRAM

Aim

To create the Entity Relationship Diagram

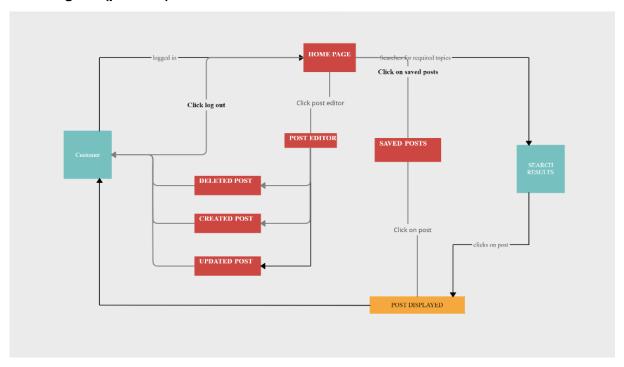**Entity Relationship Diagram**

Result:

Thus, the entity relationship diagram was created successfully.

8. DATAFLOW DAGRAM

Aim

To develop the data flow diagram for Gymverse
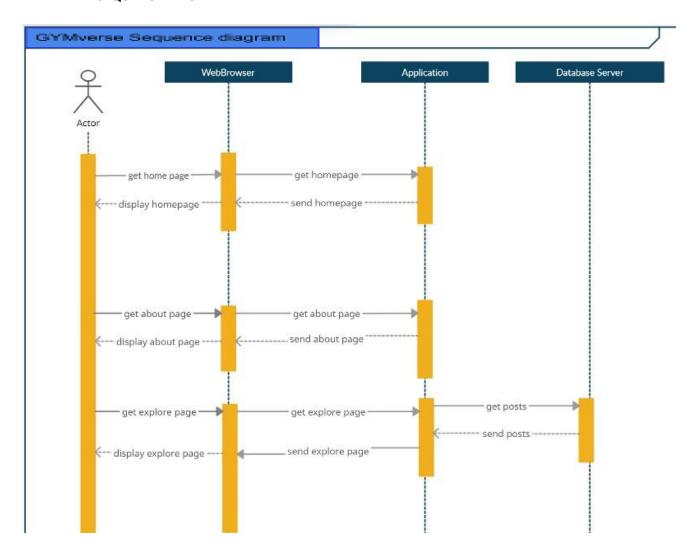
DFD Diagram (process)

Result:

Thus, the data flow diagrams have been created for Gymverse
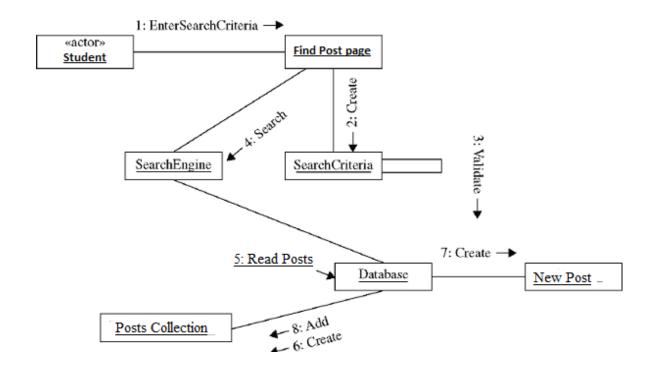
9. SEQUENCE AND COLLABORATION DIAGRAM

Aim

To create the sequence and collaboration diagram for Gymverse

SEQUENCE DIAGRAM:

Result:

Thus, the sequence and collaboration diagrams were created for Gymverse.

10. DEVELOPMEMNT OF TESTING FRAMEWOK/ USER INTERFACE

Aim

      To develop the testing framework and/or user interface framework for Gymverse

Executive Summary

The scope of a test defines what areas of a customer's product are supposed to get tested. The scope of testing of our software application is to check functional requirements like login, creating new posts and displaying posts and non-functional requirements like speed and performance.

The objective of testing our software is to identify bugs and solve them, improve performance and speed and fulfil various functional and non- functional requirements.

The approach to test the software application is manual using word template and checking for each sample test case and maintain a test manual and check for expected and actual outcome

Test Plan

Scope of Testing

The testing will cover testing the home page, signup/login page and post page on things like valid passwords, creation and display of posts, etc. and testing various non-functional requirements like speed, performance, delay.

Functional: All functional requirements are being taken care of. For ex: - functioning of home page, login/signup, post display page, etc.

Non-Functional: Almost all non- functional requirements are being taken care of. For ex: - Performance, Speed, Time delay, huge traffic of users.

Types of Testing, Methodology, Tools

Category	Methodology	Tools Required
Functional Requirements	Manual	Word Template
Non-Functional Requirements	Manual	Word Template

Result:

      Thus, the testing framework/user interface framework has been created for Gymverse.

11. TEST CASES & REPORTING

Aim

To develop the test cases manual for the GYMverse.

Test Case

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
T_001	Check signup functionality	Accepts valid email and password for new account	1. User clicks on signup 2. Enter the valid details 3. Click submit button	User should be taken to the next page where they can create new post	User goes to next page where they can create new post	Pass	success
T_002	Check signup functionality	Check if given password is valid or not	1. User clicks on signup 2. Enter email and password 3. Click submit button	User will be shown wrong password error upon entering invalid password	User gets shown wrong password after entering invalid password	Pass	success
T_003	Create post	Accept post details and store in database	1. User click on create post 2. Enter post details 3. Click submit button	User will be shown the page where all posts are displayed	User goes to the page where all posts are displayed	Pass	success

Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
T_001	Multiple users creating posts	Accepts a valid post	1. Users will click on	All users will go to other	All users go to other page where	Pass	Success

	at the same time	and store it	create post tab. 2. Fill necessary details. 3. Press Submit button.	page where all the posts are displayed	all the posts are displayed		
T_002	Users visit the home page	Enter the website link to visit the page	1. Users will type the website link. 2. User press enter.	Users will go to the root directory with a neat and beautiful user interface	Users goes to root directory with a neat and beautiful user interface.	Pass	Success

Testing manual was prepared and manual test cases were tested covering the given scope of all functional and non-functional requirements.

present obstacles to proceed further

In non-functional requirements the time delay and performance can be improved to achieve excellence in the development of software.

help from stakeholders to remove obstacles/constraints

Using paid and renowned web hosting service through the help of stakeholders can improve the performance, response time and reduce latency in the software.

Using high speed network can also contribute to the enhancement of software.

Category	Progress Against Plan	Status
Functional Testing	Green	Completed
Non-Functional Testing	Amber	In-Progress

Functional	Test Case Coverage (%)	Status
Home page	100%	Completed
Login/Signup	45%	Completed
Post page	45%	Completed

Result:

Thus, the software test conducted and documented the report successfully

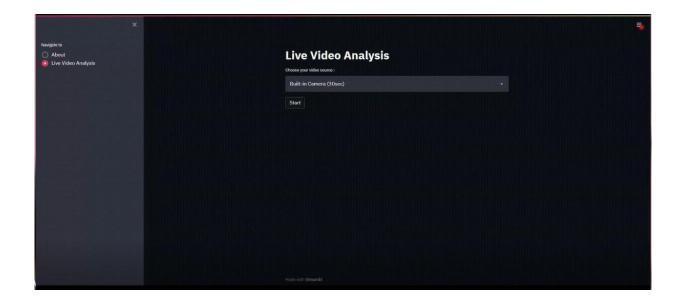# 12. ARCHITECTURE/ DESIGN/ FRAMEWORK/ IMPLEMENTATION

Aim

To provide the details of architectural design/framework/implementation

Website architecture is the hierarchical structure of website pages. This structure is reflected through internal linking. Website's structure should help users easily find information and help search engine crawlers understand the relationship between different pages.

Root Directory

When Get request called over root directory index.html is sent and hence and showed on user screen.
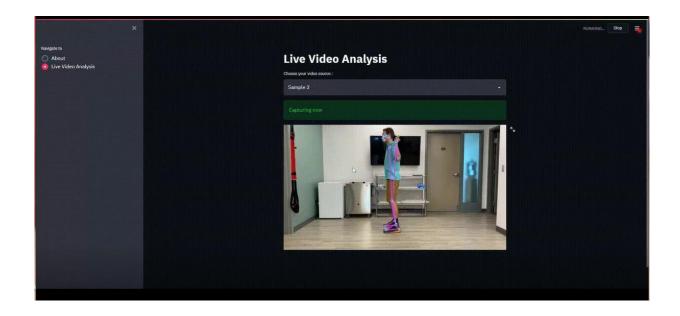

About Tab

When about is pressed in navbar server will make a get request to about tab and it will fetch about.ejs . Below the implementation of it.

CODE :

```python
from time import perf_counter
import streamlit as st
import cv2
import io
import mediapipe as mp
import numpy as np
import imageio
```

```python
if 'video_captured' not in st.session_state:
    st.session_state.video_captured = False
if 'analysis_done' not in st.session_state:
    st.session_state.analysis_done = False


def videoCapture(src):
    mp_drawing = mp.solutions.drawing_utils
    mp_pose = mp.solutions.pose

    message = st.empty();
    message.warning('Started Capturing')


    def calculate_angle(a,b,c):
        a = np.array(a) # First
        b = np.array(b) # Mid
        c = np.array(c) # End

        radians = np.arctan2(c[1]-b[1], c[0]-b[0]) - np.arctan2(a[1]-b[1],
a[0]-b[0])
        angle = np.abs(radians*180.0/np.pi)

        if angle >180.0:
            angle = 360-angle

        return angle

    k={"IMAGE":[],"KAI":[],'HKA':[], 'SHK':[],}
    cap = cv2.VideoCapture(src)
    screen = st.empty()

    webcam_timer = 30
    if src == 0:
        start_time = perf_counter()

    with mp_pose.Pose(min_detection_confidence=0.5,
min_tracking_confidence=0.5) as pose:
        message.success('Capturing now')
```
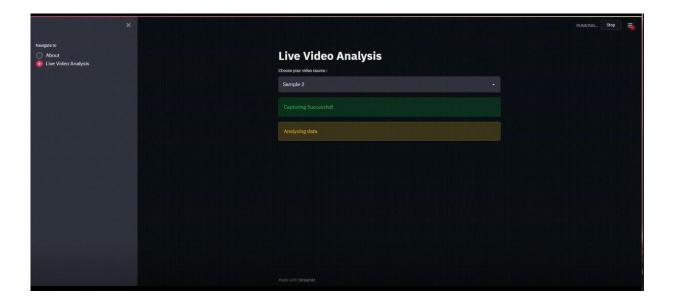
Live video analysis

This part lets you select a previously recorded or a live video for analysis.

Code:

```
while cap.isOpened():
    ret, frame = cap.read()
    if ret==False:
        break;
    image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    image.flags.writeable = False
    results = pose.process(image)
    image.flags.writeable = True
    image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
    try:
        landmarks = results.pose_landmarks.landmark
        hip =
[landmarks[mp_pose.PoseLandmark.LEFT_HIP.value].x,landmarks[mp_pose.PoseLandmar
k.LEFT_HIP.value].y]
        knee =
[landmarks[mp_pose.PoseLandmark.LEFT_KNEE.value].x,landmarks[mp_pose.PoseLandma
rk.LEFT_KNEE.value].y]
        ankle =
[landmarks[mp_pose.PoseLandmark.LEFT_ANKLE.value].x,landmarks[mp_pose.PoseLandm
ark.LEFT_ANKLE.value].y]
```

```python
            index =
[landmarks[mp_pose.PoseLandmark.LEFT_FOOT_INDEX.value].x,landmarks[mp_pose.Pose
Landmark.LEFT_FOOT_INDEX.value].y]
            shoulder =
[landmarks[mp_pose.PoseLandmark.LEFT_SHOULDER.value].x,landmarks[mp_pose.PoseLa
ndmark.LEFT_SHOULDER.value].y]
            angle_foot=calculate_angle(knee,ankle,index)
            ankle_angle = calculate_angle(hip,knee,ankle)
            hip_angle = calculate_angle(shoulder,hip,knee)


            k['IMAGE'].append(frame)
            k['KAI'].append(angle_foot)
            k['HKA'].append(ankle_angle)
            k['SHK'].append(hip_angle)

            # Visualize angle
            cv2.putText(image, str(angle),
                        tuple(np.multiply(elbow, [640, 480]).astype(int)),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2,
cv2.LINE_AA
                                )
        except:
            pass


        # Render detections
        mp_drawing.draw_landmarks(image, results.pose_landmarks,
mp_pose.POSE_CONNECTIONS,
                                  mp_drawing.DrawingSpec(color=(245,117,66),
thickness=2, circle_radius=2),
                                  mp_drawing.DrawingSpec(color=(245,66,230),
thickness=2, circle_radius=2)
                                )

        screen.image(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
        if cv2.waitKey(10) & 0xFF == ord('q'):
            break

        if (src == 0 and (perf_counter()-start_time) > webcam_timer):
#Timer if webcam is source
            cap.release()
    cap.release()
```

```
    screen.empty()
    message.success('Capturing Successful!')
    st.session_state.video_captured = True
    st.session_state.imgData = k
```


Analysis data

This is a part where the video is analysed by the machine learning algorithm designed by us in order to count and predict the proper form of the excierce and also give valuable feedback to improve the form to prevent injuries.

CODE:

```
def analyseVideo():
    message = st.empty()

    if not st.session_state.video_captured:
        pass

    k = st.session_state.imgData
    message.warning("Analysing data")
    def rep_analyse(start_index,end_index, tot_KAI):
        min_HKA = k['HKA'][start_index]
        min_HKA_i = start_index
        tot_frames = end_index - start_index + 1
        avg_KAI = tot_KAI/tot_frames
        var_KAI = 0
```
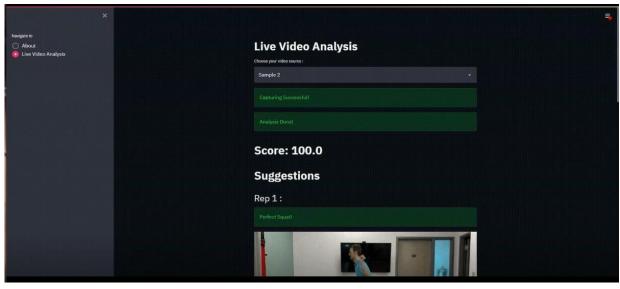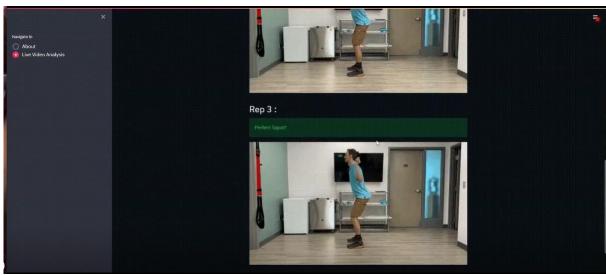
```python
        for i in range(start_index,end_index+1):
            if min_HKA >= k['HKA'][i]:
                min_HKA = k['HKA'][i]
                min_HKA_i = i
            var_KAI += (k['KAI'][i] - avg_KAI)**2


        squat_angle = int(min_HKA - 90)
        HKA_SHK_Diff = k['HKA'][min_HKA_i] - k['SHK'][min_HKA_i]
        var_KAI = var_KAI/(tot_frames-1)


        return squat_angle, HKA_SHK_Diff, var_KAI, tot_frames


    rep_start = False
    rep_mid = False
    tot_rep = 0
    tot_KAI = 0
    rep_report =
{'start_i':[],'end_i':[],'squat_angle':[],'HKA_SHK_Diff':[],'var_KAI':[],'tot_f
rames':[]}
    for i in range(len(k['IMAGE'])):
        print("Reps start:", rep_start)
        print("Reps mid:", rep_mid)
        print(tot_rep)
        print(k['KAI'][i], k['HKA'][i], k['SHK'][i])


        if not rep_start:
            if(int(abs(k['HKA'][i]) - 160) < 5):
                rep_start = True
                rep_report['start_i'].append(i)
                tot_KAI = k['KAI'][i]
        else:
            tot_KAI += k['KAI'][i]
            if(int(abs(k['HKA'][i] - 150)) < 5 and rep_mid):
                rep_report['end_i'].append(i)
                squat_angle, HKA_SHK_Diff, var_KAI, tot_frames =
rep_analyse(rep_report['start_i'][tot_rep],rep_report['end_i'][tot_rep],tot_KAI
)
                rep_report['squat_angle'].append(squat_angle)
                rep_report['HKA_SHK_Diff'].append(HKA_SHK_Diff)
                rep_report['var_KAI'].append(var_KAI)
                rep_report['tot_frames'].append(tot_frames)
                tot_rep += 1
                rep_start = rep_mid = False
```

```python
                tot_KAI = 0
            elif(int(abs(k['HKA'][i] - 95)) < 10 and rep_start):
                rep_mid = True


    print(rep_report)
    if not rep_report['start_i']:
        st.error("No valid data points found")
        return


    #Generating Sugegstions based on rep_report:
    suggestions = []
    perfect_rep = 0
    for rep in range(0,tot_rep):
        rep_suggestion = []

        if(rep_report['squat_angle'][rep] > 15):
            rep_suggestion.append("Incomplete squat! Squat little lower")
        elif(rep_report['squat_angle'][rep] < -10):
            rep_suggestion.append("Too much sqautting! Squat little less")

        if(rep_report['tot_frames'][rep] < 50):
            rep_suggestion.append("Too fast! Squat slower")

        if(rep_report['HKA_SHK_Diff'][rep] < 0):
            rep_suggestion.append("Knee too much forward! Try to keep knee from
leaning forward")
        elif(rep_report['HKA_SHK_Diff'][rep] > 15):
            rep_suggestion.append("Try to keep you back straight")

        if(rep_report['var_KAI'][rep] > 25):
            rep_suggestion.append("Your feet are shifting! Keep your legs
planted")

        if not rep_suggestion:
            rep_suggestion.append("Perfect Squat!")
            perfect_rep += 1

        suggestions.append(rep_suggestion)

        with imageio.get_writer("rep_"+str(rep+1)+".gif", mode="I") as writer:
            for gif_frame,img_frame in
enumerate(range(rep_report['start_i'][rep],rep_report['end_i'][rep]+1)):
```

```python
                rgb_frame = cv2.cvtColor(k['IMAGE'][img_frame],
cv2.COLOR_BGR2RGB)
                writer.append_data(rgb_frame)


    print("Suggestions and gif stored")
    print(suggestions)
    st.session_state.analysis_done = True
    message.success("Analysis Done!")
    st.session_state.analysisData = {'suggestions':suggestions,
'perfect_rep':perfect_rep, 'tot_rep':tot_rep}

nav = st.sidebar.radio("Navigate to",('About','Live Video Analysis'))
if nav == "About":
    st.title("About")
    #TO BE DONE

elif nav == "Live Video Analysis":
    st.title("Live Video Analysis")
    video_src = st.selectbox("Choose your video source :",('Built-in Camera
(30sec)','Sample 1','Sample 2', 'Other video'))
    if video_src == 'Built-in Camera (30sec)':
        src = 0
    elif video_src == 'Sample 1':
        src = "squat_2.mp4"
    elif video_src == 'Sample 2':
        src = "squat_3.mp4"
    else:
        video = st.file_uploader("Upload .mp4 file",type=["mp4"])
        if video:
            byte_video = io.BytesIO(video.read())
            with open("uploaded_vid.mp4", 'wb') as out:
                out.write(byte_video.read())
            out.close()
            src = "uploaded_vid.mp4"
```

```
    start_button = st.empty()
    if start_button.button("Start"):
        start_button.empty()
        videoCapture(src)


    if st.session_state.video_captured:
        if not st.session_state.analysis_done:
            analyseVideo()
            if st.session_state.analysis_done:
                analysisData = st.session_state.analysisData


        #Attempt to export video from frames
        # frames = st.session_state.imgData['IMAGE']
        # out =
cv2.VideoWriter('output_video.avi',cv2.VideoWriter_fourcc(*'DIVX'), 60,
(640,480))
        # for frame in frames:
```

```python
#       out.write(frame)
# out.release()
# with open('exercise.mp4') as f:
#       if st.download_button("Export Video",f):
#            st.success("Video Exported Successfuly")


if st.session_state.analysis_done:
    perfect_rep = st.session_state.analysisData['perfect_rep']
    tot_rep = st.session_state.analysisData['tot_rep']
    suggestions = st.session_state.analysisData['suggestions']
    st.title("Score: " + str((perfect_rep/tot_rep)*100))
    st.title("Suggestions")
    for rep in range(0,tot_rep):
        st.header("Rep " + str(rep+1) + " :")
        for suggestion in suggestions[rep]:
            print(suggestion)
            if suggestion == "Perfect Squat!":
                st.success(suggestion)
            else:
                st.error(suggestion)
        st.image("rep_"+str(rep+1)+".gif")
```

Result:

Thus, the details of architectural design/framework/implementation along with the screenshots were provided.

CONCLUSION

The GYMverse website finally gets deployed after the testing and verification of functional and non-functional requirements. The website in working condition can be found at the link

Link to the website:
https://protected-sierra-97021.herokuapp.com/

The following QR-code can also be scanned to access the website link:

REFRENCES

- https://github.com/josephmisiti/awesome-machine-learning

- https://www.geeksforgeeks.org/machine-learning/

- https://www.ibm.com/in-en/topics/computer-vision#:~:text=Computer%20vision%20is%20a%20field,recommendations%20based%20on%20that%20information.

- https://streamlit.io/

- https://opencv.org/

- https://stackoverflow.com/

APPENDIX (CODE)

```python
from time import perf_counter

import streamlit as st

import cv2

import io

import mediapipe as mp

import numpy as np

import imageio


if 'video_captured' not in st.session_state:

    st.session_state.video_captured = False

if 'analysis_done' not in st.session_state:

    st.session_state.analysis_done = False


def videoCapture(src):

    mp_drawing = mp.solutions.drawing_utils

    mp_pose = mp.solutions.pose


    message = st.empty();

    message.warning('Started Capturing')


    def calculate_angle(a,b,c):

        a = np.array(a) # First

        b = np.array(b) # Mid

        c = np.array(c) # End
```

```python
            radians  =  np.arctan2(c[1]-b[1],  c[0]-b[0])  -
np.arctan2(a[1]-b[1], a[0]-b[0])

        angle = np.abs(radians*180.0/np.pi)



        if angle >180.0:

            angle = 360-angle



        return angle



    k={"IMAGE":[],"KAI":[],'HKA':[], 'SHK':[],}

    cap = cv2.VideoCapture(src)

    screen = st.empty()



    webcam_timer = 30

    if src == 0:

        start_time = perf_counter()



                with      mp_pose.Pose(min_detection_confidence=0.5,
min_tracking_confidence=0.5) as pose:

        message.success('Capturing now')



        while cap.isOpened():

            ret, frame = cap.read()

            if ret==False:

                break;

            image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

            image.flags.writeable = False

            results = pose.process(image)
```

```python
        image.flags.writeable = True

        image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)

        try:

            landmarks = results.pose_landmarks.landmark

                                                    hip      =
[landmarks[mp_pose.PoseLandmark.LEFT_HIP.value].x,landmarks[mp_pose.PoseL
andmark.LEFT_HIP.value].y]

                                                   knee      =
[landmarks[mp_pose.PoseLandmark.LEFT_KNEE.value].x,landmarks[mp_pose.Pose
Landmark.LEFT_KNEE.value].y]

                                                   ankle     =
[landmarks[mp_pose.PoseLandmark.LEFT_ANKLE.value].x,landmarks[mp_pose.Pos
eLandmark.LEFT_ANKLE.value].y]

                                                   index     =
[landmarks[mp_pose.PoseLandmark.LEFT_FOOT_INDEX.value].x,landmarks[mp_pos
e.PoseLandmark.LEFT_FOOT_INDEX.value].y]

                                                 shoulder    =
[landmarks[mp_pose.PoseLandmark.LEFT_SHOULDER.value].x,landmarks[mp_pose.
PoseLandmark.LEFT_SHOULDER.value].y]

            angle_foot=calculate_angle(knee,ankle,index)

            ankle_angle = calculate_angle(hip,knee,ankle)

            hip_angle = calculate_angle(shoulder,hip,knee)



            k['IMAGE'].append(frame)

            k['KAI'].append(angle_foot)

            k['HKA'].append(ankle_angle)

            k['SHK'].append(hip_angle)


            # Visualize angle

            cv2.putText(image, str(angle),
```

```python
                                                tuple(np.multiply(elbow, [640,
480]).astype(int)),
                                        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255,
255), 2, cv2.LINE_AA
                                    )

            except:
                pass



            # Render detections
                mp_drawing.draw_landmarks(image, results.pose_landmarks,
mp_pose.POSE_CONNECTIONS,

mp_drawing.DrawingSpec(color=(245,117,66), thickness=2, circle_radius=2),

mp_drawing.DrawingSpec(color=(245,66,230), thickness=2, circle_radius=2)
                                    )

            screen.image(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
            if cv2.waitKey(10) & 0xFF == ord('q'):
                break

            if (src == 0 and (perf_counter()-start_time) > webcam_timer):
#Timer if webcam is source
                cap.release()
        cap.release()


    screen.empty()

    message.success('Capturing Successful!')
```

```python
        st.session_state.video_captured = True

        st.session_state.imgData = k


def analyseVideo():

    message = st.empty()


    if not st.session_state.video_captured:

        pass


    k = st.session_state.imgData

    message.warning("Analysing data")

    def rep_analyse(start_index,end_index, tot_KAI):

        min_HKA = k['HKA'][start_index]

        min_HKA_i = start_index

        tot_frames = end_index - start_index + 1

        avg_KAI = tot_KAI/tot_frames

        var_KAI = 0


        for i in range(start_index,end_index+1):

            if min_HKA >= k['HKA'][i]:

                min_HKA = k['HKA'][i]

                min_HKA_i = i

            var_KAI += (k['KAI'][i] - avg_KAI)**2


        squat_angle = int(min_HKA - 90)

        HKA_SHK_Diff = k['HKA'][min_HKA_i] - k['SHK'][min_HKA_i]

        var_KAI = var_KAI/(tot_frames-1)
```

```python
        return squat_angle, HKA_SHK_Diff, var_KAI, tot_frames


    rep_start = False

    rep_mid = False

    tot_rep = 0

    tot_KAI = 0

    rep_report                              =
{'start_i':[],'end_i':[],'squat_angle':[],'HKA_SHK_Diff':[],'var_KAI':[],
'tot_frames':[]}

    for i in range(len(k['IMAGE'])):

        print("Reps start:", rep_start)

        print("Reps mid:", rep_mid)

        print(tot_rep)

        print(k['KAI'][i], k['HKA'][i], k['SHK'][i])


        if not rep_start:

            if(int(abs(k['HKA'][i]) - 160) < 5):

                rep_start = True

                rep_report['start_i'].append(i)

                tot_KAI = k['KAI'][i]

        else:

            tot_KAI += k['KAI'][i]

            if(int(abs(k['HKA'][i] - 150)) < 5 and rep_mid):

                rep_report['end_i'].append(i)

                squat_angle, HKA_SHK_Diff, var_KAI, tot_frames =
rep_analyse(rep_report['start_i'][tot_rep],rep_report['end_i'][tot_rep],t
ot_KAI)
```

```python
                    rep_report['squat_angle'].append(squat_angle)

                    rep_report['HKA_SHK_Diff'].append(HKA_SHK_Diff)

                    rep_report['var_KAI'].append(var_KAI)

                    rep_report['tot_frames'].append(tot_frames)

                    tot_rep += 1

                    rep_start = rep_mid = False

                    tot_KAI = 0

                elif(int(abs(k['HKA'][i] - 95)) < 10 and rep_start):

                    rep_mid = True


    print(rep_report)

    if not rep_report['start_i']:

        st.error("No valid data points found")

        return


    #Generating Sugegstions based on rep_report:

    suggestions = []

    perfect_rep = 0

    for rep in range(0,tot_rep):

        rep_suggestion = []


        if(rep_report['squat_angle'][rep] > 15):

            rep_suggestion.append("Incomplete squat! Squat little lower")

        elif(rep_report['squat_angle'][rep] < -10):

                rep_suggestion.append("Too much sqautting! Squat little
less")
```

```python
        if(rep_report['tot_frames'][rep] < 50):

            rep_suggestion.append("Too fast! Squat slower")



        if(rep_report['HKA_SHK_Diff'][rep] < 0):

            rep_suggestion.append("Knee too much forward! Try to keep
knee from leaning forward")

        elif(rep_report['HKA_SHK_Diff'][rep] > 15):

            rep_suggestion.append("Try to keep you back straight")



        if(rep_report['var_KAI'][rep] > 25):

            rep_suggestion.append("Your feet are shifting! Keep your legs
planted")



        if not rep_suggestion:

            rep_suggestion.append("Perfect Squat!")

            perfect_rep += 1



        suggestions.append(rep_suggestion)



        with imageio.get_writer("rep_"+str(rep+1)+".gif", mode="I") as
writer:

                        for   gif_frame,img_frame   in
enumerate(range(rep_report['start_i'][rep],rep_report['end_i'][rep]+1)):

                    rgb_frame = cv2.cvtColor(k['IMAGE'][img_frame],
cv2.COLOR_BGR2RGB)

                writer.append_data(rgb_frame)



    print("Suggestions and gif stored")

    print(suggestions)
```

```python
        st.session_state.analysis_done = True

    message.success("Analysis Done!")

        st.session_state.analysisData  =  {'suggestions':suggestions,
'perfect_rep':perfect_rep, 'tot_rep':tot_rep}


nav = st.sidebar.radio("Navigate to",('About','Live Video Analysis'))

if nav == "About":

    st.title("About")

    #TO BE DONE


elif nav == "Live Video Analysis":

    st.title("Live Video Analysis")

      video_src = st.selectbox("Choose your video source :",('Built-in
Camera (30sec)','Sample 1','Sample 2', 'Other video'))

    if video_src == 'Built-in Camera (30sec)':

        src = 0

    elif video_src == 'Sample 1':

        src = "squat_2.mp4"

    elif video_src == 'Sample 2':

        src = "squat_3.mp4"

    else:

        video = st.file_uploader("Upload .mp4 file",type=["mp4"])

        if video:

            byte_video = io.BytesIO(video.read())

            with open("uploaded_vid.mp4", 'wb') as out:

                out.write(byte_video.read())

            out.close()

            src = "uploaded_vid.mp4"
```

```python
    start_button = st.empty()

    if start_button.button("Start"):

        start_button.empty()

        videoCapture(src)


    if st.session_state.video_captured:

        if not st.session_state.analysis_done:

            analyseVideo()

            if st.session_state.analysis_done:

                analysisData = st.session_state.analysisData


        #Attempt to export video from frames

        # frames = st.session_state.imgData['IMAGE']

                                                    #    out    =
cv2.VideoWriter('output_video.avi',cv2.VideoWriter_fourcc(*'DIVX'),   60,
(640,480))

        # for frame in frames:

        #    out.write(frame)

        # out.release()

        # with open('exercise.mp4') as f:

        #    if st.download_button("Export Video",f):

        #        st.success("Video Exported Successfuly")


    if st.session_state.analysis_done:

        perfect_rep = st.session_state.analysisData['perfect_rep']

        tot_rep = st.session_state.analysisData['tot_rep']

        suggestions = st.session_state.analysisData['suggestions']
```

```python
        st.title("Score: " + str((perfect_rep/tot_rep)*100))

        st.title("Suggestions")

        for rep in range(0,tot_rep):

            st.header("Rep " + str(rep+1) + " :")

            for suggestion in suggestions[rep]:

                print(suggestion)

                if suggestion == "Perfect Squat!":

                    st.success(suggestion)

                else:

                    st.error(suggestion)

            st.image("rep_"+str(rep+1)+".gif")
```