**Interactive web application for mental well being**

|  |
| --- |
| **CONTENT** |
| **Abstract** |
| **1.INTRODUCTION** |
| 1.1 Motivation |
| 1.2 Problem Statement |
| 1.3 Objective of the Project |
| 1.4 Scope |
| 1.5 Project Introduction |
| **2.LTERATURE SURVEY** |
| 2.1 Related Work |
| **3. SYSTEM ANALYSIS** |
| 3.1 Existing System |
| 3.2 Proposed System |
| 3.3 work Flow of Proposed system |
| **4. REQUIREMENT ANALYSIS** |
| 4.1 Hardware Requirements |
| 4.2 Software Requirements |
| 4.3 Architecture |
| **5. METHODOLOGY** |
| **6. SYSTEM DESIGN** |
| 6.1 Introduction of Input design |
| 6.2 UML Diagram(class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram) |
| 6.3 Data Flow Diagram |
| **7. IMPLEMENTATION AND RESULTS** |
| 7.1 Modules |
| 7.2 Output Screens |
| **8. SYSTEM STUDY AND TESTING** |
| 8.1 Feasibility study |
| 8.2 Types of test & Test Cases |
| **9. CONCLUSION** |
| **10. FUTURE ENHANCEMENT** |
| **11. REFERENCES** |

**ABSTRACT**

In the contemporary landscape where mental well-being is paramount, the integration of technology into wellness practices offers innovative solutions to enhance personal health regimes. This project, "Interactive Web Application for Mental Well-Being," aims to develop a comprehensive platform that synergizes personalized yoga practice with an emotion-aware chatbot to promote holistic health. The system comprises two primary modules: Yoga and Chatbot.

The Yoga Module begins by assessing the user’s current emotional state through a selection of 19 predefined moods. Based on this input, the system recommends three tailored yoga poses from a dataset of 77 poses designed to address specific emotional needs. To ensure correct pose execution, the module employs advanced computer vision techniques, utilizing the YOLO (You Only Look Once) model for accurate pose detection and PoseNet for extracting key joint features. An angle heuristic algorithm analyzes the user’s posture, providing real-time corrective feedback to enhance the effectiveness and safety of the practice.

Complementing the Yoga Module, the Chatbot Module engages users by recognizing and responding to 59 distinct emotions. Through an intuitive conversational flow, the chatbot inquires about the reasons behind the user’s emotions, offers empathetic support, and recommends relevant YouTube videos to aid in emotional regulation. Post-interaction, the chatbot gathers feedback to refine its responses, ensuring a personalized and supportive user experience.

By combining personalized physical activity with emotional support, this web application provides a unique and effective tool for individuals seeking to improve their mental and physical well-being. The integration of cutting-edge machine learning and natural language processing technologies ensures that users receive tailored guidance and support, fostering a balanced and healthy lifestyle.

Keywords: Personalized yoga, Computer vision, Machine learning, Mood assessment, Emotion recognition, Pose recommendation, YOLO model, PoseNet model, Chatbot, Pose correction, Mental well-being.

**1. INTRODUCTION**

**1.1 Motivation:**

The motivation behind this project stems from a desire to leverage technology to enhance yoga practice and promote emotional well-being. Traditional yoga methods often lack personalized feedback and guidance, leading to challenges in achieving optimal results and avoiding injuries. Additionally, emotional support is crucial but often not integrated into well-being practices. By combining computer vision, machine learning, and natural language processing, we aim to provide personalized recommendations, real-time feedback, and supportive interactions tailored to individual needs, making yoga practice and emotional support more accessible, effective, and safe for people of all levels.

**1.2 Problem Statement:**

Traditional yoga practice often lacks personalized guidance and correction, leading to potential inefficiencies and injury risks, while emotional support systems are limited. By integrating computer vision, machine learning, and natural language processing, our project aims to provide a solution that analyzes users' mood inputs, recommends suitable yoga poses, offers pose correction, and interacts through a supportive chatbot. This addresses the need for personalized guidance, correction, and emotional support in well-being practices, catering to individual needs and enhancing overall well-being.

**1.3 Objective of the Project:**

The objective of our project is to enhance physical and mental well-being by providing personalized yoga practice and emotional support. Using computer vision, machine learning, and a chatbot, it tailors yoga pose recommendations, and delivers supportive interactions based on user emotions.

**1.4 Scope:**

The scope of our project encompasses the development of a comprehensive system for personalized yoga practice and emotional support, integrating various technologies and methodologies. This includes:

1. Mood Assessment: Implementing a mechanism to assess users' emotional states to tailor yoga recommendations and provide appropriate chatbot interactions.

2. Pose Recommendation: Providing personalized recommendations of yoga poses based on the user's mood input.

3. Pose Detection: Utilizing the YOLO (You Only Look Once) model for accurate detection of yoga poses from user-uploaded images or videos.

4. Key Features Extraction: Employing the PoseNet model to extract key features of the detected yoga poses for further analysis.

5. Pose Correction: Implementing an angle heuristic algorithm to provide real-time feedback and corrective measures for improving the execution of yoga poses.

6. Chatbot Interaction: Developing a chatbot module to interact with users, assess their emotional states, provide supportive responses, and recommend resources such as YouTube videos.

Overall, the project aims to provide a holistic solution for enhancing both physical and mental well-being through advanced technology.

**Project Introduction:**

In today's fast-paced and digitally-driven world, mental well-being has emerged as a critical component of overall health. The increasing prevalence of stress, anxiety, and other mental health challenges underscores the need for accessible and effective solutions that can seamlessly integrate into individuals' daily lives. Concurrently, the rise of digital health technologies presents unprecedented opportunities to enhance traditional wellness practices through personalization and real-time feedback. It is within this context that the "Interactive Web Application for Mental Well-Being" is conceived—a cutting-edge platform designed to harmonize physical and emotional health through the synergistic integration of personalized yoga practices and an emotion-aware chatbot.

Yoga, a practice revered for its benefits in promoting physical flexibility, strength, and mental tranquility, often relies on consistent guidance to ensure correct posture and technique. Traditional methods, whether conducted in-person or through pre-recorded tutorials, frequently lack the ability to provide individualized feedback, which can lead to inefficiencies and, in some cases, increase the risk of injury. Moreover, while yoga inherently supports mental well-being, the absence of a dedicated emotional support system within these practices limits their potential to address the multifaceted nature of mental health challenges.

Addressing these gaps, the proposed web application incorporates advanced technologies such as computer vision, machine learning, and natural language processing to deliver a holistic wellness experience. The platform is divided into two primary modules: the Yoga Module and the Chatbot Module. The Yoga Module leverages the YOLO (You Only Look Once) model for accurate real-time pose detection and PoseNet for extracting key joint features, enabling precise pose analysis and correction through an angle heuristic algorithm. This ensures that users receive immediate, personalized feedback on their yoga practice, enhancing both safety and effectiveness.

Complementing the physical aspect, the Chatbot Module serves as an empathetic companion, engaging users by recognizing and responding to a wide spectrum of emotions. By facilitating conversations that delve into the reasons behind users' emotional states and providing supportive resources such as curated YouTube videos, the chatbot fosters emotional regulation and mental resilience. This dual approach not only personalizes the yoga experience based on users' emotional needs but also offers continuous emotional support, creating a comprehensive tool for holistic well-being.

Furthermore, the application features a user-friendly interface that includes functionalities such as mood assessment, personalized pose recommendations, image upload for pose analysis, and a dashboard for tracking personal progress. Secure user authentication and data management ensure that sensitive information is protected, adhering to stringent privacy standards.

The integration of these sophisticated technologies within a unified platform represents a significant advancement in the realm of digital wellness solutions. By providing tailored physical activity recommendations and robust emotional support, the "Interactive Web Application for Mental Well-Being" empowers individuals to take proactive steps towards maintaining and enhancing their mental and physical health. This innovative approach not only makes yoga practice more accessible and effective for users of all levels but also bridges the gap between physical exercise and emotional well-being, fostering a balanced and healthy lifestyle in an increasingly complex world.

Key Features:

1. Personalized Yoga Recommendations: Tailors yoga poses based on users' selected moods from a diverse set of 19 predefined emotions.
2. Real-Time Pose Detection and Correction: Utilizes YOLO and PoseNet models to analyze and provide corrective feedback on users' yoga poses, ensuring proper technique and safety.
3. Emotion-Aware Chatbot: Engages users in empathetic conversations, recognizes a wide range of emotions, and offers supportive resources to aid emotional regulation.
4. User Dashboard: Allows users to track their yoga practice history, monitor progress, and access past interactions for a comprehensive view of their well-being journey.

By seamlessly blending physical exercise with emotional support, this web application stands as a pioneering solution in the digital wellness landscape, offering users a personalized, interactive, and supportive environment to cultivate both their mental and physical well-being.

.

1. **LITERATURE SURVEY**

### **1. Digital Interventions for Mental Well-Being**

A growing body of literature highlights the potential of digital platforms to promote mental health and emotional resilience. Early work focused on web-based cognitive behavioral therapy (CBT) modules, chatbots for emotional support, and guided meditation apps (Andersson & Titov, 2014; Fitzpatrick et al., 2017). With the smartphone revolution, mindfulness and stress reduction apps proliferated, enabling users to access mental health resources conveniently (Bakker et al., 2016). More recently, research has explored integrative systems that combine lifestyle adjustments—such as yoga, exercise, and sleep hygiene—with emotional well-being support tools. This body of work suggests that personalizing recommendations based on user mood or stress levels can enhance user adherence and improve mental health outcomes (Mohr et al., 2017).

### **2. Pose Estimation and Analysis Techniques**

The advent of deep learning and computer vision has revolutionized human pose estimation. Traditional methods relied on handcrafted features and structural models (e.g., pictorial structures), but modern deep convolutional neural networks and models like OpenPose, MoveNet, and PoseNet have significantly improved accuracy (Cao et al., 2017; Papandreou et al., 2018; Chen et al., 2022). These pose estimation models can robustly identify joint keypoints under varying conditions, making them suitable for fitness and sports applications. In the context of yoga, several studies have employed pose estimation to assess practitioner alignment, detect common errors, and provide corrective feedback (Sharma et al., 2019; Velloso et al., 2017). The literature emphasizes the importance of well-annotated datasets and domain-specific fine-tuning to achieve reliable performance in specialized tasks such as yoga pose correction.

### **3. Recommender Systems in Wellness and Physical Activity**

Recommendation engines have evolved from e-commerce and media applications to health, fitness, and wellness domains (Karimi et al., 2018). Early health recommender systems primarily focused on nutrition and exercise plans, using collaborative filtering and content-based methods to tailor advice. In yoga and wellness, recommendation models have expanded to include mood-based suggestions, adapting routines to users’ emotional states (Jadhav et al., 2020). Hybrid recommender systems combine user similarity, item features, and contextual information—such as user mood or stress levels—to improve relevance and personalization (Abbas et al., 2022). Literature in this area supports the integration of user psychological states into recommendation algorithms, showing that personalized suggestions can improve engagement, satisfaction, and adherence to recommended exercises or therapies.

### **4. Integrating Pose Correction with Emotional Support**

While fitness applications and virtual personal trainers have existed for some time, more recent studies look at integrating emotional cues, empathic dialogue systems, and affective computing into these platforms (McDuff et al., 2020). By correlating user mood data with pose recommendations and feedback, research suggests a more holistic approach: users not only receive instructions on correct body alignment but also supportive messages that acknowledge their emotional state. For instance, yoga apps that detect stress or anxiety may adjust the recommended sequence to more calming poses, while simultaneously providing gentle, empathetic guidance. This line of work bridges the gap between purely physical guidance (e.g., posture correction) and psychological support, with literature indicating that multi-modal feedback (verbal, textual, and visual corrections) combined with empathic responses can enhance the user’s overall experience and outcomes (Laamarti et al., 2020).

### **5. Gaps, Challenges, and Future Directions**

Despite advances, several gaps remain. First, the availability of large-scale, high-quality yoga pose datasets annotated for both keypoint accuracy and pose difficulty is limited. Most pose estimation models work well on general human activities but struggle with the complexity and variety of yoga postures. Second, integrating emotional states into recommendation systems poses challenges around reliable mood detection and modeling user context without becoming invasive (Calvo & Peters, 2014). Ethical considerations related to data privacy, accuracy of mood inference, and bias in recommendations are highlighted in recent ethics-focused literature (Mittelstadt et al., 2016).

Addressing these gaps can guide the next wave of research and application development, enabling more nuanced, context-aware, and ethically grounded yoga recommendation and correction systems that support users’ ultimate well-being and personal growth.

.

**3. SYSTEM ANALYSIS**

**3.1 Existing System:**

Existing methods for personalized yoga practice typically involve manual instruction from yoga teachers or the use of pre-recorded video tutorials, which often lack real-time feedback and personalized guidance. While some online platforms offer algorithm-driven recommendations based on user preferences or skill level, they do not account for the practitioner's emotional state or provide detailed feedback on pose execution. These methods face challenges in accurately assessing users' emotional states, providing precise feedback on pose alignment, and offering tailored recommendations for individual needs. Moreover, the lack of integration across different technologies results in fragmented user experiences.

**Disadvantages:**

1. Lack of Personalized Feedback: Traditional methods often rely on generic instructions, which may not address the specific needs or limitations of individual practitioners.
2. Risk of Injury: Without real-time feedback, practitioners may perform poses incorrectly, leading to potential strain or injury.
3. Inability to Address Emotional States: Many methods do not consider the practitioner's emotional state when recommending poses or sequences.
4. Lack of Integration with Advanced Technology: Existing solutions may not leverage computer vision and machine learning for real-time feedback and personalized guidance.
5. Fragmented User Experience: Users may need to utilize multiple platforms for mood assessment, pose recommendations, and instructional content, resulting in a disjointed experience.

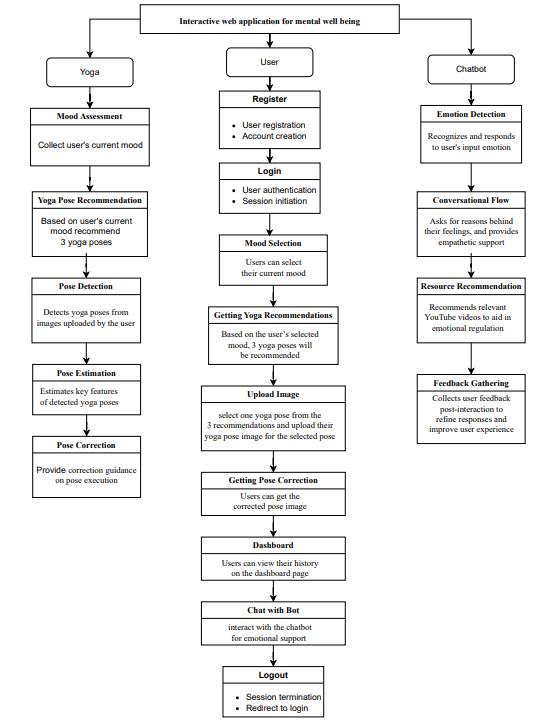
**3.2 PROPOSED SYSTEM:**

The proposed system for personalized yoga practice and emotional support addresses these limitations by integrating advanced computer vision, machine learning, and natural language processing techniques. The system includes a mood assessment module to tailor yoga recommendations and chatbot interactions based on users' emotional states. It provides personalized yoga pose recommendations using the YOLO model for pose detection and PoseNet for key features extraction, with an angle heuristic algorithm offering real-time feedback and correction. The chatbot module engages users by recognizing and responding to various emotions, offering empathetic support and recommending resources. This unified platform, accessible online and via mobile applications, ensures a streamlined and comprehensive experience, promoting enhanced physical and mental well-being through personalized guidance, real-time feedback, and emotional support.

**Advantages:**

1. Personalized Guidance: The system recommends yoga poses tailored to the user's emotional state, providing a more personalized and effective practice experience.
2. Real-Time Feedback: Leveraging computer vision and machine learning, the system provides immediate feedback on pose alignment and execution.
3. Reduced Risk of Injury: The pose correction algorithm offers detailed guidance to improve pose execution, minimizing the risk of injury.
4. Enhanced User Experience: The seamless integration of various technologies ensures a streamlined and comprehensive solution for personalized yoga practice, enhancing user engagement and satisfaction.

**3.3 PROJECT FLOW**



**4. REQUIREMENT ANALYSIS**

**4.1 Hardware Requirements:**

Processor - I3/Intel Processor

Hard Disk - 160GB

Key Board - Standard Windows Keyboard

Mouse - Two or Three Button Mouse

Monitor - SVGA

RAM - 8GB

**4.2 S/W CONFIGURATION:**

Operating System : Windows 7/8/10

Server side Script : HTML, CSS, Bootstrap & JS

Programming Language : Python

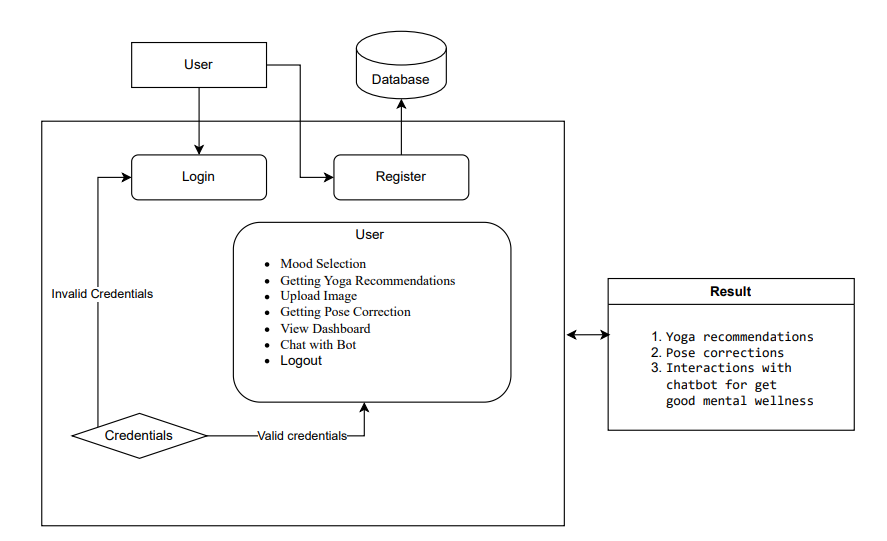
Libraries : Flask, Pandas, Mysql.connector, Os, Scikit-learn, Numpy

IDE/Workbench : PyCharm

Technology : Python 3.6+

Server Deployment : Xampp Server

**4.3 Architecture:**



**5. METHODOLOGY:**

**5.1 Singular Value Decomposition (SVD) for Recommendation System:**

In the development of YogaMood Harmony, Singular Value Decomposition (SVD) serves as a pivotal component within the recommendation system, embodying a structured methodology and architectural framework:

The methodology commences with an exhaustive analysis of user preferences and emotional states, informing the collection of pertinent data comprising user mood inputs and historical yoga practice records. This data undergoes meticulous preprocessing to construct a user-item matrix, where each user's ratings or preferences for specific yoga poses are represented.

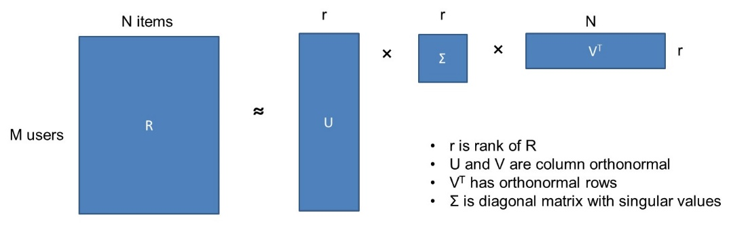
Subsequently, a robust SVD model is developed, meticulously engineered to decompose the user-item matrix into its constituent components: user factors, item factors, and interaction factors. This intricate decomposition process effectively captures latent features underlying user preferences and yoga poses, fostering a nuanced understanding of the complex dynamics involved.

Through rigorous training and optimization, the SVD model is fine-tuned to minimize reconstruction error and enhance predictive accuracy. Leveraging the latent factors extracted through SVD, the recommendation system adeptly generates personalized recommendations tailored to users' historical preferences and current mood inputs.

Integration of the SVD-based recommendation system into YogaMood Harmony's architecture ensures seamless interaction with other pivotal modules, including pose detection and real-time feedback mechanisms. This cohesive integration fosters a unified user experience, where personalized recommendations seamlessly complement other facets of the yoga practice journey.

Following integration, user feedback plays a pivotal role in refining the recommendation system. Incorporating user insights and assessments regarding recommendation accuracy and relevance enables continuous enhancement of the algorithm, aligning it more closely with users' evolving needs and preferences.

By adhering to this meticulous methodology and leveraging the power of Singular Value Decomposition within the recommendation system, YogaMood Harmony endeavors to deliver a transformative platform. Through personalized yoga recommendations, the project aims to empower users on a holistic wellness journey tailored to their unique emotional states and preferences.



**5.2 YOLOv3:**

YogaMood Harmony represents a cutting-edge project that seamlessly integrates advanced computer vision techniques with traditional yoga practices. At its core lies a commitment to revolutionizing the yoga experience, enhancing practitioners' well-being through personalized guidance and real-time feedback. Central to this endeavor is the utilization of YOLOv3 (You Only Look Once version 3) for yoga pose detection, offering a robust methodology within the project.

**Yoga Pose Detection Methodology:**

The methodology commences with an exhaustive acquisition of a diverse dataset meticulously curated with annotated images or videos showcasing a myriad of yoga poses from varying perspectives. This dataset serves as the foundation for subsequent training, vital for refining the model's proficiency across a spectrum of scenarios.

Following dataset acquisition, an intricate preprocessing phase ensues to meticulously refine and prepare the collected data for training. Tasks such as image resizing, pixel value normalization, and data augmentation are meticulously executed to bolster the model's generalization and robustness.

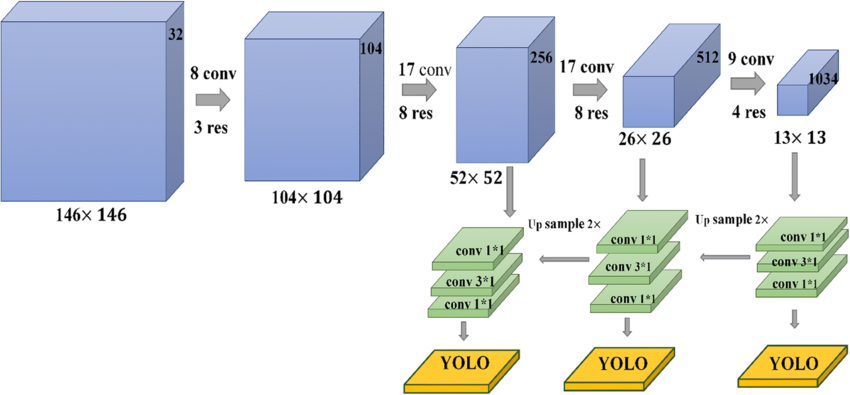
Subsequently, the YOLOv3 model architecture is adeptly employed to train a specialized neural network tailored explicitly for yoga pose detection. Renowned for its prowess in object detection, YOLOv3 partitions the input image into a grid and swiftly predicts bounding boxes and class probabilities for each grid cell in a singular pass.

Throughout the training process, the YOLOv3 model undergoes iterative refinement, fine-tuning its parameters based on discrepancies between predicted and ground truth annotations to optimize its performance.

Post-training, the trained YOLOv3 model seamlessly integrates into the YogaMood Harmony system architecture, synergizing with other modules such as the recommendation system and real-time feedback mechanism. This cohesive integration ensures a holistic user experience, fostering seamless interaction throughout the yoga practice journey.

Upon deployment, the YOLOv3-based pose detection module operates in real-time, accurately identifying and localizing yoga poses within live video streams or recorded sessions. This real-time capability empowers practitioners to receive immediate feedback on posture alignment, augmenting the efficacy and safety of their practice sessions significantly.

Through the meticulous integration of YOLOv3 for pose detection, YogaMood Harmony endeavors to deliver a transformative platform, harnessing advanced computer vision techniques to enhance the effectiveness and accessibility of yoga practice, ultimately fostering holistic well-being among practitioners.



**5.2 PoseNet:**

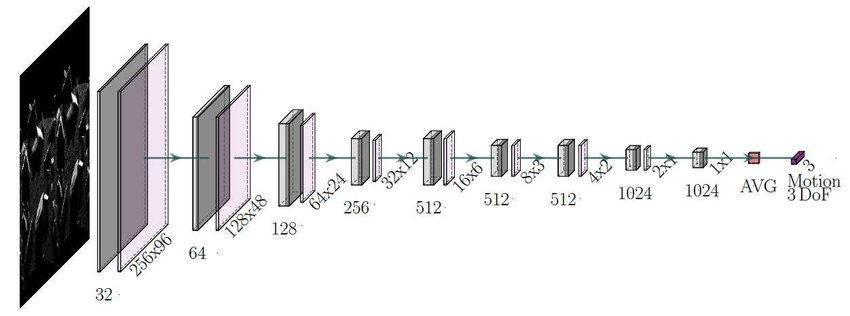
Within YogaMood Harmony's innovative framework, the strategic integration of advanced technologies like PoseNet for yoga pose estimation represents a pivotal advancement in personalized wellness solutions. This cutting-edge approach not only revolutionizes the practice of yoga but also empowers practitioners with precise insights into their physical alignment, facilitating tailored guidance to optimize their practice.

At the core of this methodology lies the sophisticated algorithms of PoseNet, renowned for their exceptional accuracy in human pose estimation. Through meticulous analysis of input images or video frames, YogaMood Harmony leverages PoseNet to discern and precisely estimate the intricate postures adopted by practitioners during their yoga sessions.

Beyond mere pose detection, YogaMood Harmony goes a step further by extracting key features from the estimated poses. These extracted features encapsulate crucial details about posture angles, body positions, and spatial relationships, providing practitioners with invaluable insights into their practice.

Integrated seamlessly into the YogaMood Harmony ecosystem, these extracted key features serve as a cornerstone for personalized guidance and feedback. By harmonizing with complementary modules such as mood-based recommendation systems and real-time feedback mechanisms, practitioners benefit from tailored recommendations and immediate feedback aligned with their unique needs and goals.

The incorporation of PoseNet and key feature extraction within YogaMood Harmony signifies a commitment to advancing holistic wellness through the fusion of technology and tradition. By empowering practitioners with actionable insights and personalized support, YogaMood Harmony aims to elevate the yoga experience, fostering a deeper connection between mind, body, and technology in the pursuit of well-being.



**5.3 Angle Heuristic:**

The angle heuristic method, a cornerstone of the YogaMood Harmony project's pose correction module, is grounded in both traditional yoga principles and modern computer vision techniques. At its core, the method leverages the understanding that proper alignment is fundamental to the effectiveness and safety of yoga practice. By employing computer vision algorithms to detect key points on the user's body during yoga poses, such as joints and anatomical landmarks, the system captures essential data for assessing posture alignment. The method then calculates the angles formed by these key points, drawing from established yoga alignment principles to evaluate the alignment of the pose.

Through heuristic rules and thresholds derived from yoga teachings and alignment guidelines, the system interprets these angles to provide real-time feedback to the user. For example, in poses such as downward-facing dog (Adho Mukha Svanasana), the angle between the arms, spine, and legs is crucial for maintaining proper alignment and weight distribution. Deviations from optimal angles may indicate areas where posture correction is needed to prevent strain or injury.

Moreover, the angle heuristic method recognizes the individuality of yoga practitioners and their unique anatomical variations. By offering personalized feedback tailored to each user's body and practice level, the system fosters a supportive environment for exploration and growth in yoga practice. Ultimately, this integration of traditional wisdom with cutting-edge technology embodies the YogaMood Harmony project's commitment to promoting holistic wellness through personalized yoga experiences.

### **5.4 Chatbot**

### **1. Project Overview**

**Objective:** Develop a static chatbot that interacts with users to understand their emotions, provides relevant resources, and offers supportive feedback based on a predefined dataset.

**Technologies Used:**

* Backend:
* Python: Core programming language for server-side logic.
* Flask: Lightweight web framework to handle HTTP requests and serve the application.
* Frontend:
* HTML: Structuring the chatbot interface.
* CSS: Styling the chatbot for an appealing appearance.
* Bootstrap: Ensuring responsive design across different devices.
* JavaScript: Managing user interactions and dynamic content updates.
* Data Handling:
* JSON: Storing the chatbot's predefined conversational dataset.

### **2. Dataset Analysis**

The provided dataset is a JSON structure containing detailed information for various emotions. Each emotion object includes:

* name: The emotion's name (e.g., happy, sad).
* follow\_up\_question: A question to delve deeper into the user's emotion.
* reasons: Categories that help identify the root cause of the emotion.
* comfort: A comforting message acknowledging the user's emotion.
* videos: Curated video resources related to understanding and coping with the emotion.
* feedback\_question: A question to assess the user's state after interaction.
* feedback\_positive **&** feedback\_negative: Responses based on the user's feedback.

This structured dataset facilitates the creation of an empathetic and resourceful chatbot capable of guiding users through their emotional states.

### **3. Backend Development with Flask**

Set up the Flask server to handle incoming requests from the frontend and respond based on the static dataset.

* **Greeting:** Initiates the conversation with a welcome message and presents emotion options.
* **Emotion Selection:** Processes the selected emotion and asks for the reason.
* **Reason Selection:** Provides comfort messages and shares relevant resources.
* **Feedback Collection:** Gathers user feedback and responds accordingly.
* **Closure:** Offers options to continue the conversation or end it gracefully.

### **5.4 Overall Methodology**

The project encompasses two primary components:

1. **Yoga Pose Recommendation System:** Utilizes collaborative filtering (SVD) to recommend yoga poses based on user-selected moods.
2. **Yoga Pose Identification and Correction System:** Employs computer vision techniques (YOLO and MoveNet PoseNet) to identify yoga poses from images and provide corrective feedback based on keypoint analysis.

This integrated system aims to enhance users' mental well-being by recommending suitable yoga practices and ensuring correct pose execution through real-time feedback.

### **Part 1: Yoga Pose Recommendation System**

#### **1. Analysis**

* **Libraries and Dataset Importation:**
* **Pandas (**pd**):** For data manipulation and analysis.
* **NumPy (**np**):** For numerical operations.
* **SciPy's Sparse Matrix Modules (**csr\_matrix**,** svds**):** For handling large, sparse datasets and performing Singular Value Decomposition (SVD).
* **Unique Moods and Poses Extraction:**
* moods **and** poses**:** Arrays containing unique user moods and yoga poses from the dataset.
* mood\_to\_idx **and** pose\_to\_idx**:** Dictionaries mapping each mood and pose to a unique index, facilitating the creation of an interaction matrix.
* **Interaction Matrix Creation:**
* interaction\_matrix**:** A sparse matrix initialized with zeros, where rows represent moods and columns represent yoga poses.
* **Populating the Matrix:** Iterates through each record in the dataset, incrementing the count for each mood-pose interaction.
* **Singular Value Decomposition (SVD):**
* num\_features**:** Number of latent factors (set to the minimum dimension of the interaction matrix minus one).
* **SVD (**svds**):** Decomposes the interaction matrix into three matrices (u, sigma, vt), capturing underlying patterns.
* **Recommendation Function:**
* recommend\_poses**:** Function that takes a user mood and returns the top N recommended yoga poses based on latent features derived from SVD.

#### **2. Methodology**

**a. Data Preparation:**

1. **Data Collection:** Compile a dataset (Recommendation\_yoga\_data.csv) containing records of user moods and corresponding yoga poses practiced.
2. **Data Cleaning:**
   * Remove duplicates and inconsistencies.
   * Standardize entries (e.g., trimming whitespace, consistent casing).

**b. Feature Engineering:**

1. **Mapping Moods and Poses:**
   * Create unique identifiers for each mood and yoga pose.
   * Facilitate matrix construction through index mappings.

**c. Interaction Matrix Construction:**

1. **Matrix Representation:**
   * Rows represent user moods.
   * Columns represent yoga poses.
   * Cell values indicate the frequency of a particular mood-pose interaction.

**d. Collaborative Filtering via SVD:**

1. **Matrix Decomposition:**
   * Apply SVD to reduce dimensionality and uncover latent factors influencing pose recommendations.
2. **Latent Feature Extraction:**
   * Capture patterns and relationships between moods and yoga poses.

**e. Recommendation Generation:**

1. **Scoring Mechanism:**
   * Calculate scores for each yoga pose based on the latent features associated with a user's mood.
2. **Top-N Selection:**
   * Recommend the top N poses with the highest scores to the user.

### **Part 2: Yoga Pose Identification and Correction System**

#### **1. Analysis**

* **Libraries and Model Loading:**
* **TensorFlow and TensorFlow Hub (**tf**,** hub**):** For loading and utilizing the MoveNet PoseNet model.
* **OpenCV (**cv2**):** For image processing and YOLO model integration.
* **PIL (**Image**):** For image handling.
* **NumPy (**np**), Math (**math**), JSON (**json**):** For numerical operations, angle calculations, and data serialization.
* **Image Preprocessing and Keypoint Extraction:**
* load\_and\_preprocess\_image**:** Loads an image, ensures it's in RGB format, resizes it to 192x192 pixels, normalizes pixel values, and adds a batch dimension.
* extract\_keypoints\_using\_posenet**:** Resizes and scales the image tensor, performs pose estimation using MoveNet, and extracts keypoints.
* **Directory Processing and Keypoint Storage:**
* process\_directory**:** Recursively processes directories to extract and store keypoints for each yoga pose.
* **Keypoint Storage:** Saves extracted keypoints in a JSON file (reference\_poses\_keypoints.json) for later reference.
* **YOLO Model Integration for Person Detection:**
* load\_yolo\_model**:** Loads the YOLOv3 model for object detection.
* detect\_and\_crop\_person**:** Detects persons in an image and crops the image to include only the detected person, facilitating focused pose estimation.
* **Angle Calculation and Pose Correction:**
* calculate\_angle**:** Computes the angle between three keypoints using the law of cosines, aiding in assessing pose correctness.
* **Pose Angles Definition:**
* POSE\_ANGLES\_KEYPOINTS**:** Dictionary mapping each yoga pose to specific keypoint triplets used for angle calculations and pose assessment.
* **Pose Correction Feedback:**
* provide\_pose\_correction\_feedback\_using\_angles**:** Compares user-extracted angles with reference angles and generates feedback messages indicating discrepancies.

#### **2. Methodology**

**a. Data Collection and Preparation:**

1. **Image Dataset Compilation:**
   * Gather a diverse set of images for each yoga pose, organized into directories named after the poses.
2. **Person Detection and Cropping:**
   * Utilize YOLOv3 to detect and crop the person in each image, ensuring focused pose analysis.
3. **Pose Keypoint Extraction:**
   * Apply MoveNet PoseNet to extract keypoints from each cropped image.
4. **Keypoint Storage:**
   * Store extracted keypoints in a structured JSON file (reference\_poses\_keypoints.json) for reference during pose correction.

**b. Pose Identification:**

1. **User Image Processing:**
   * Upon user image upload, detect and crop the person using YOLO.
   * Preprocess the cropped image to match MoveNet's input requirements.
2. **Keypoint Extraction:**
   * Use MoveNet to obtain keypoints representing the user's pose.

**c. Pose Correction Mechanism:**

1. **Reference Keypoints Comparison:**
   * Retrieve reference keypoints for the intended yoga pose from the JSON file.
2. **Angle Calculation:**
   * Calculate specific angles between keypoints using the law of cosines to assess pose correctness.
3. **Feedback Generation:**
   * Compare user angles with reference angles.
   * Generate feedback messages highlighting discrepancies and suggesting adjustments.

**d. Pose Angles Definition:**

* POSE\_ANGLES\_KEYPOINTS**:** A comprehensive mapping of yoga poses to specific keypoint triplets used for angle calculations, tailored to each pose's unique requirements.

**e. Integration Steps:**

1. **Model Loading:**
   * Load YOLOv3 for object detection and MoveNet PoseNet for pose estimation.
2. **Image Handling:**
   * Ensure uploaded images are in the correct format and size.
3. **Feedback Delivery:**
   * Present corrective feedback to users, enhancing their pose execution and overall practice effectiveness.

### **5.5 Overview of overall methodology**

Combining both components, the project's methodology can be outlined as follows:

1. **Data Acquisition and Preparation:**
   * **Recommendation System:** Compile and clean a dataset linking user moods to yoga poses.
   * **Pose Identification:** Gather a comprehensive image dataset for each yoga pose, ensuring diversity and accuracy.
2. **Model Development:**
   * **Recommendation System:** Implement collaborative filtering using SVD to uncover latent relationships between moods and poses.
   * **Pose Identification and Correction:** Integrate YOLOv3 for object detection and MoveNet PoseNet for pose estimation, defining keypoint-based metrics for pose correctness.
3. **System Integration:**
   * **Backend Integration:**
     + Connect the recommendation system to the pose identification module.
     + Ensure seamless data flow between mood selection, pose recommendation, image upload, pose detection, and feedback generation.
   * **Frontend Development:**
     + Design intuitive user interfaces for mood selection, pose recommendations, image uploads, and feedback display.
4. **User Interaction Workflow:**
   * **Step 1:** User selects their current mood from predefined options.
   * **Step 2:** The recommendation system suggests suitable yoga poses based on the selected mood.
   * **Step 3:** User selects a recommended pose and uploads an image performing the pose.
   * **Step 4:** The system detects the pose, extracts keypoints, and compares them with reference keypoints.
   * **Step 5:** Feedback is provided to the user, highlighting areas for improvement to ensure correct pose execution.

**6. SYSTEM DESIGN**

**6.1 Introduction of Input Design:**

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −

* It should serve specific purpose effectively such as storing, recording, and retrieving the information.
* It ensures proper completion with accuracy.
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency, and simplicity.
* All these objectives are obtained using the knowledge of basic design principles regarding
  + What are the inputs needed for the system?
  + How end users respond to different elements of forms and screens.

### Objectives for Input Design:

The objectives of input design are −

* To design data entry and input procedures
* To reduce input volume
* To design source documents for data capture or devise other data capture methods
* To design input data records, data entry screens, user interface screens, etc.
* To use validation checks and develop effective input controls.

**Output Design:**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

### Objectives of Output Design:

The objectives of input design are:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user’s requirements.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.
* To make the output available on time for making good decisions.

**6.2 UML Diagrams:**

**UML Diagrams:**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

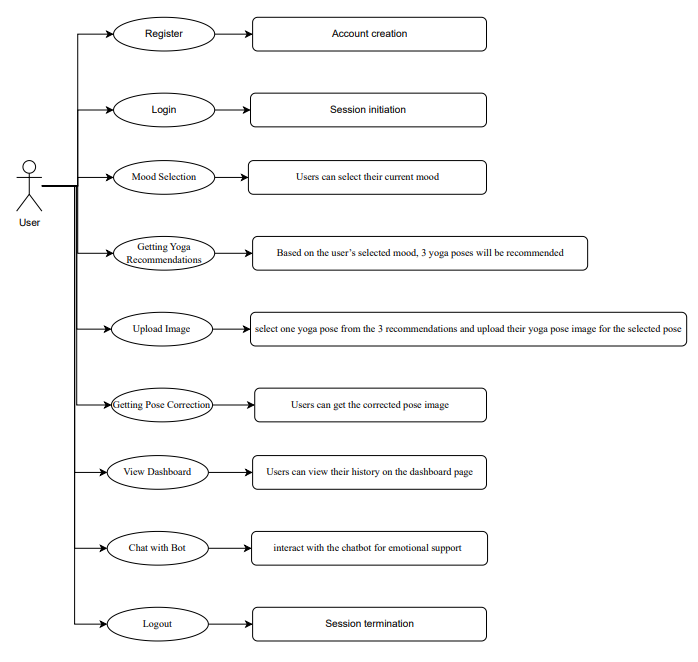
The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artefacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

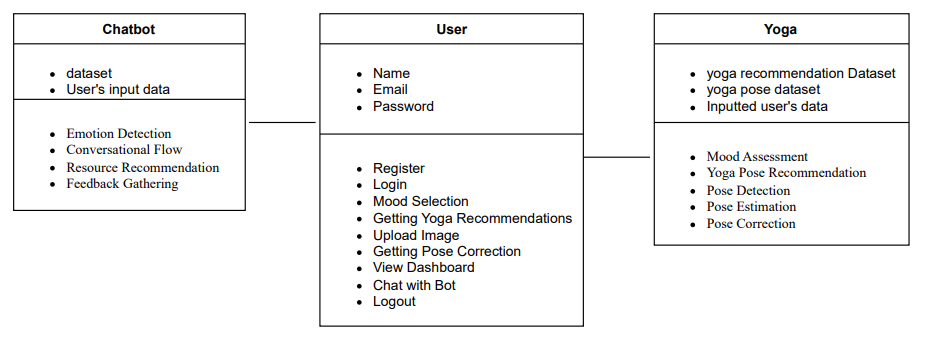
**6.2.1 Use Case Diagram:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



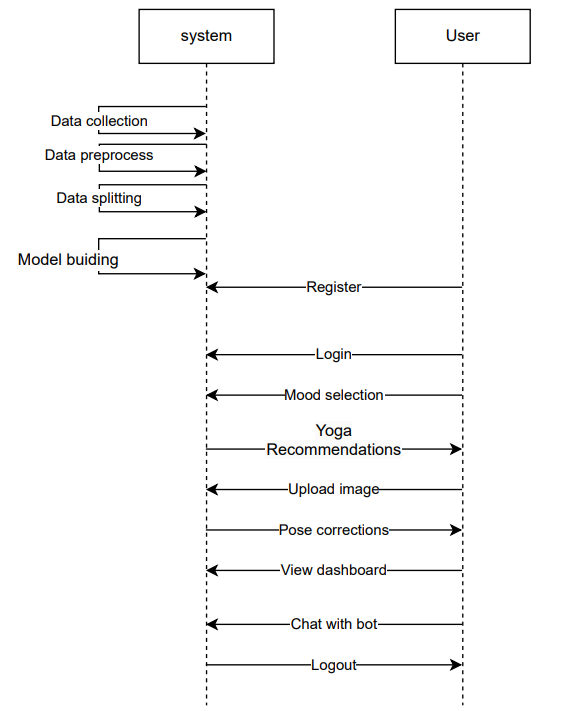
**6.2.2 Class Diagram:**

In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



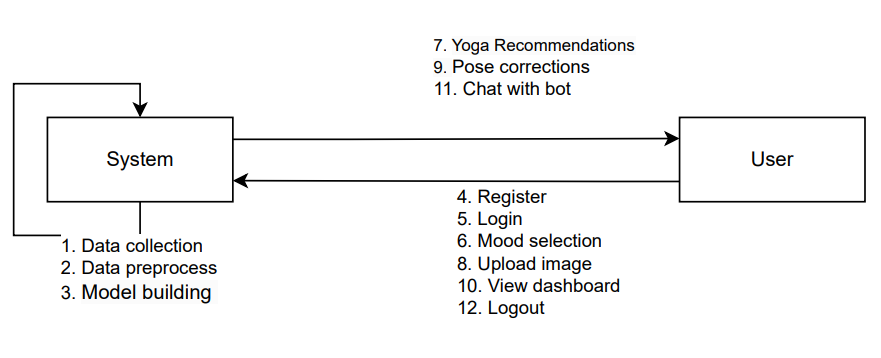
**6.2.3 Sequence Diagram:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams



**6.2.4 Collaboration Diagram:**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.



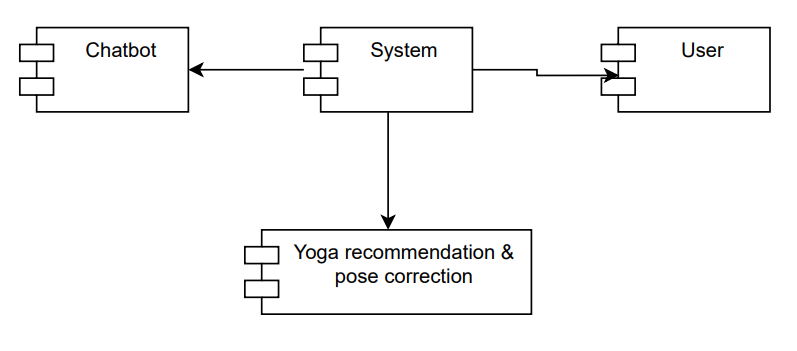
**6.2.5 Deployment Diagram**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.



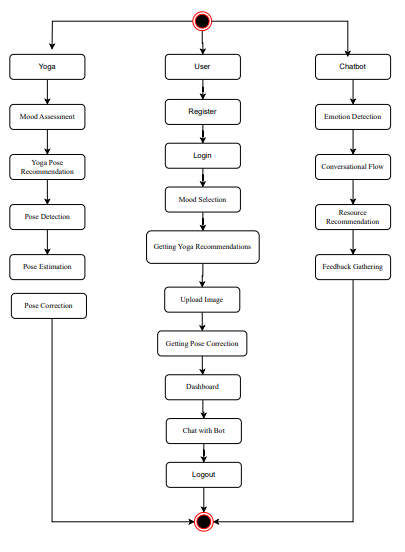
**6.2.6 Component Diagram**:

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical **c**omponents in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development.



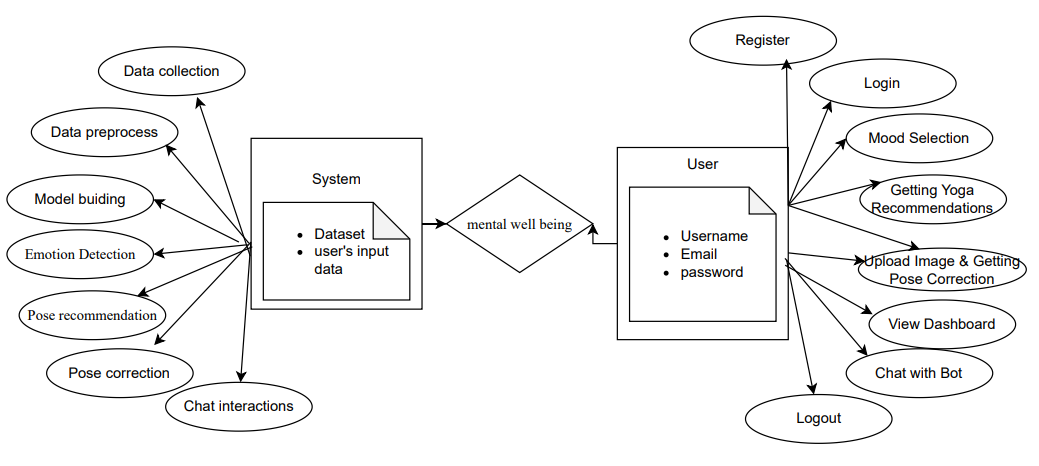
**6.2.7 Activity Diagram:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



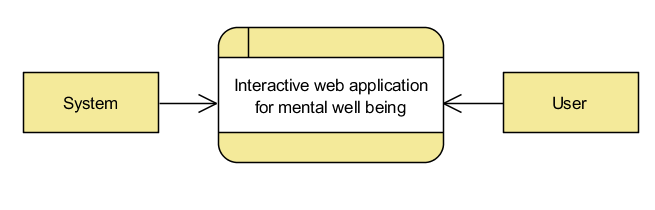
**6.2.8 ER Diagram:**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set. An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

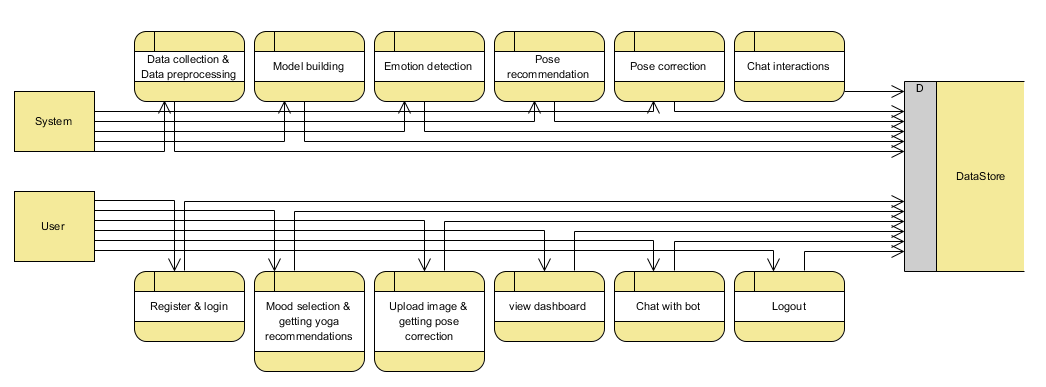


**6.3 DFD Diagram:**

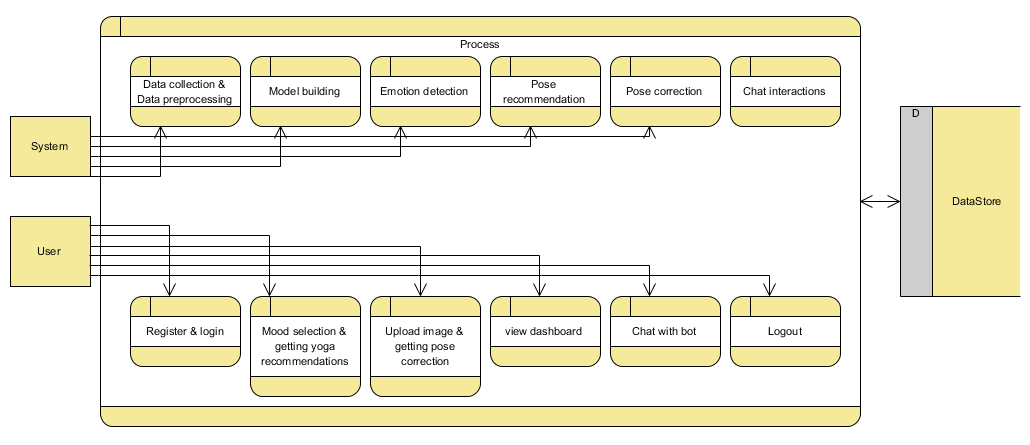
A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.



**Level 1 Diagram:**

****

**Level 2 Diagram:**

****

**7. IMPLEMENTATION**

**7.1 MODULES:**

**1. System:**

* 1. **Yoga system module**
     1. Mood Assessment: Collects user input on their current mood or emotional state.
     2. Yoga Pose Recommendation: Recommends 3 yoga poses based on the user's selected mood.
     3. Pose Detection: Detects yoga poses from images uploaded by the user.
     4. Pose Estimation: Estimates key features of detected yoga poses, such as joint positions and angles.
     5. Pose Correction: Provides real-time feedback and correction guidance on pose execution.
  2. **Chabot Module:**
     1. Emotion Detection: Recognizes and responds to 59 distinct emotions based on user input.
     2. Conversational Flow: Guides the user through their emotions, asks for reasons behind their feelings, and provides empathetic support.
     3. Resource Recommendation: Recommends relevant YouTube videos to aid in emotional regulation.
     4. Feedback Gathering: Collects user feedback post-interaction to refine responses and improve user experience.

**2. User:**

**2.1** Register: Users can register with their credentials such as email and password.

**2.2** Login: Users can log in with their registered credentials.

**2.3** Mood Selection: Users can select their current mood.

**2.4** Getting Yoga Recommendations: Based on the user’s selected mood, 3 yoga poses will be recommended.

**2.5** Upload Image: Users can select one yoga pose from the 3 recommendations and upload their yoga pose image for the selected pose.

**2.6** Getting Pose Correction: The uploaded yoga pose image will be analyzed by the trained model, which provides corrections for the pose. Users can then view the corrected pose image. This information (selected mood, uploaded image, pose-corrected image) will be stored in the database.

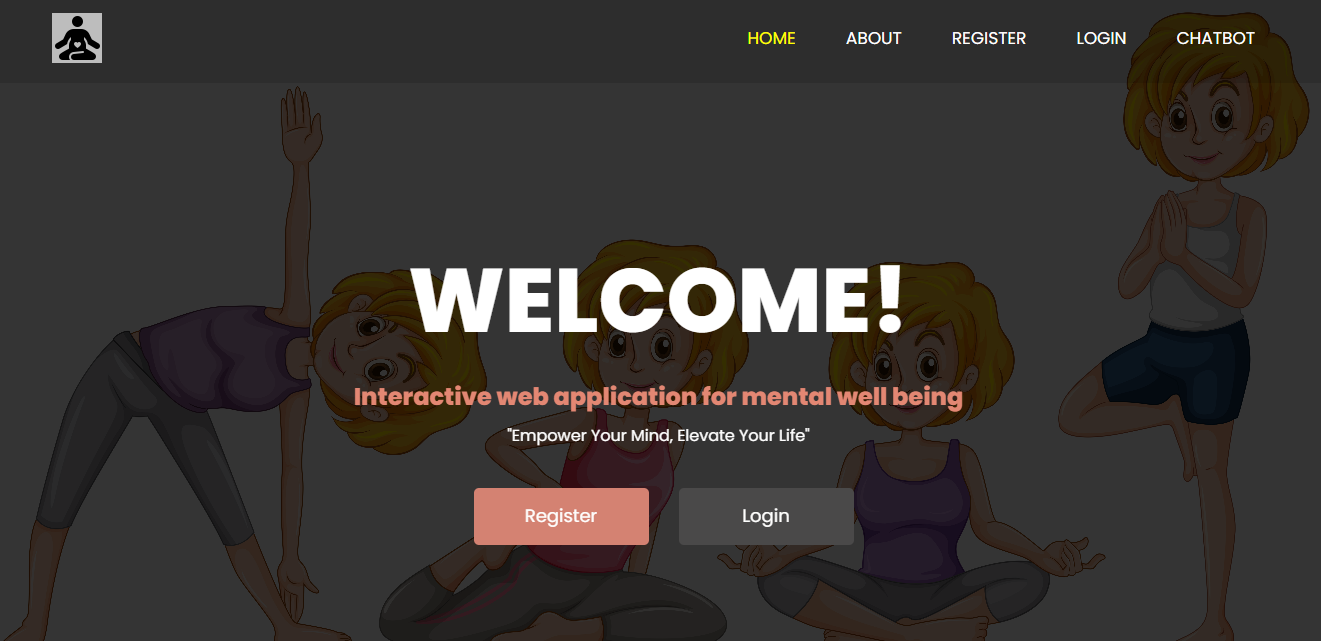
**2.7** Dashboard: Users can view their history on the dashboard page. Here they can view their selected mood, uploaded image, and pose-corrected image ordered by date. Users can also retrieve history data by a specific date or period.

**2.8** Chat with Bot: Users can interact with the chatbot for emotional support and recommendations.

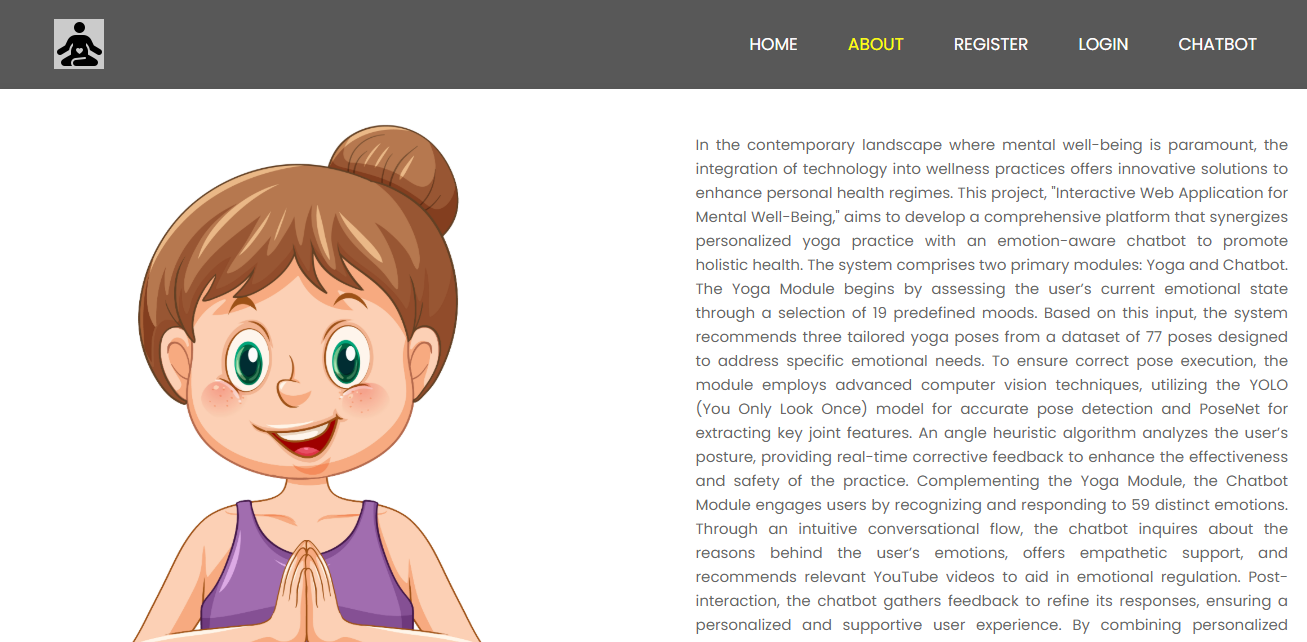
**2.9** Logout: After completing their activities, users can log out from the website.

**7.2 OUTPUT SCREENS:**

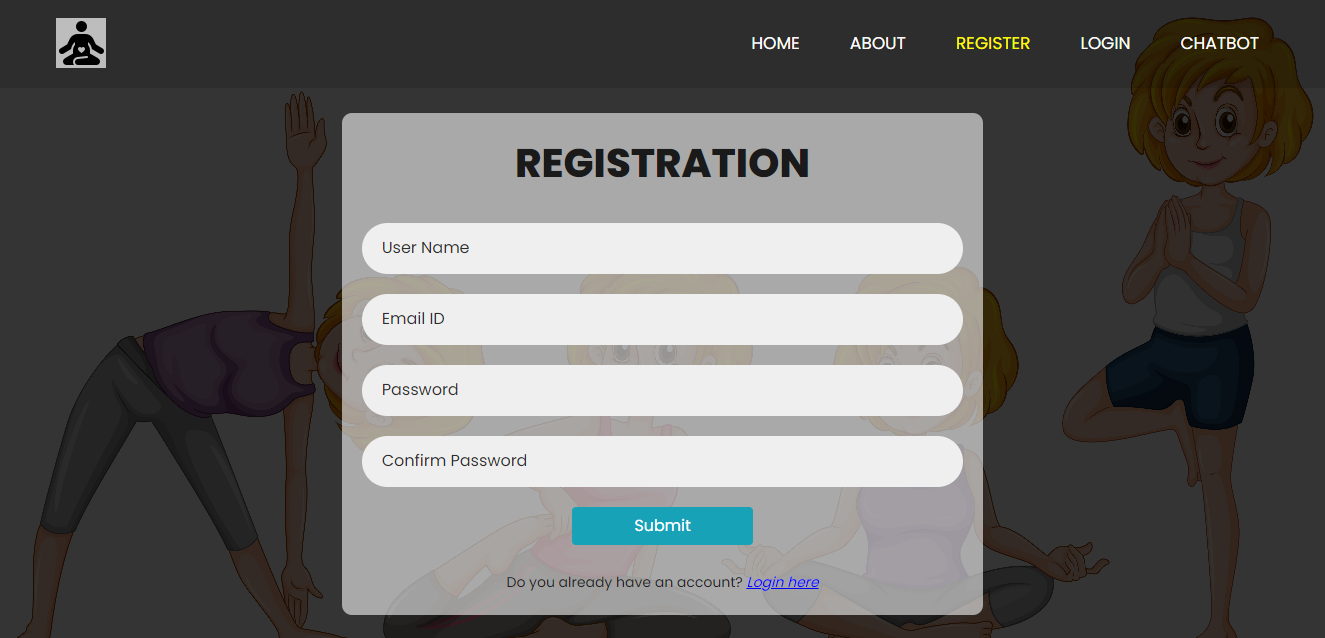
**INDEX PAGE:** This is the landing page of our website.

****

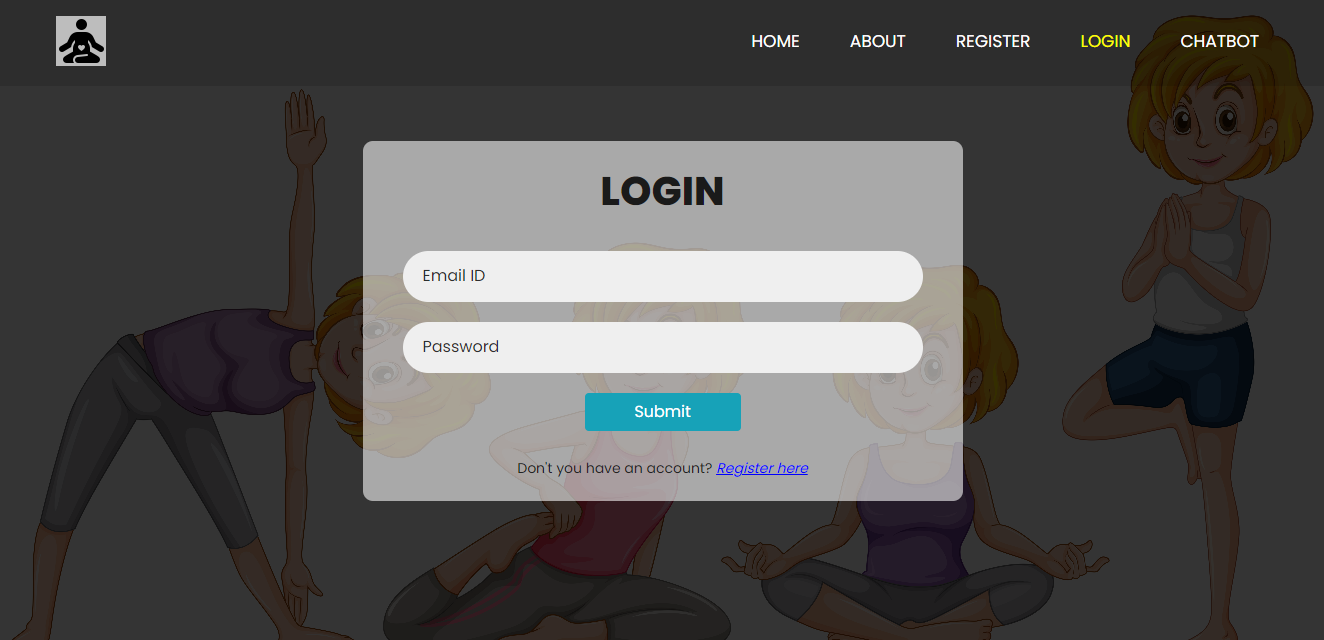
**ABOUT PAGE:** This is about section which contains information about our project



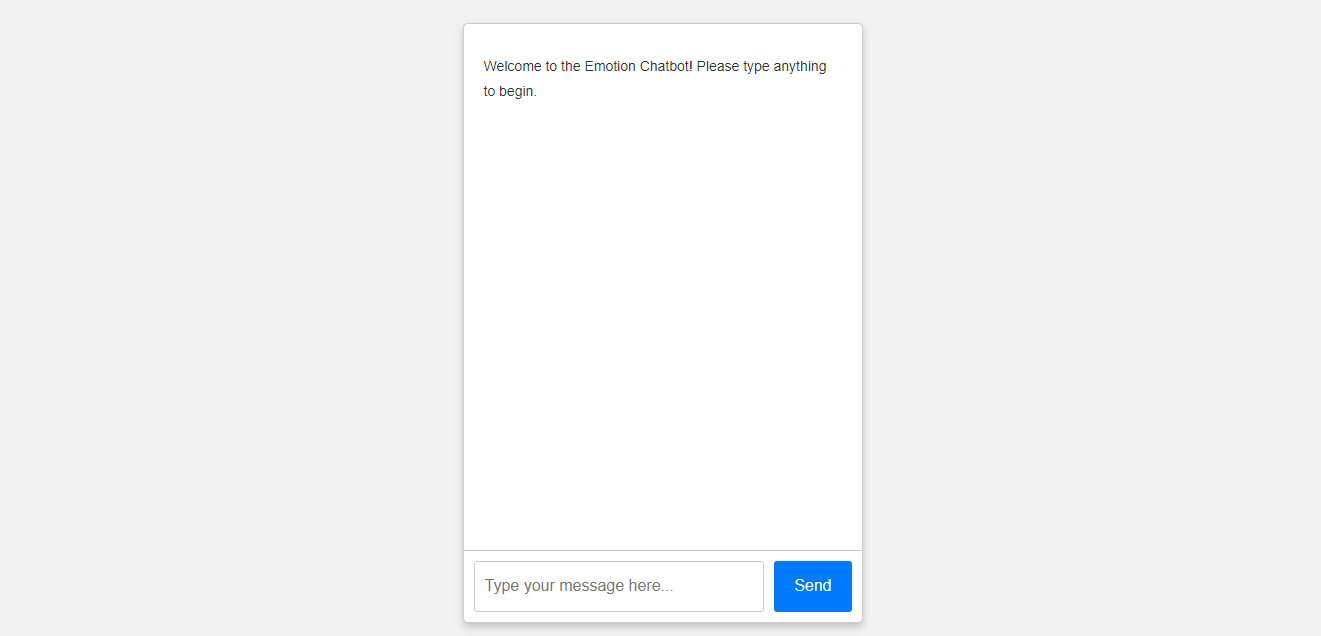
**REGISTRATION PAGE:** This is Registration page. In here, user can register with their credentials such as email, username, password.

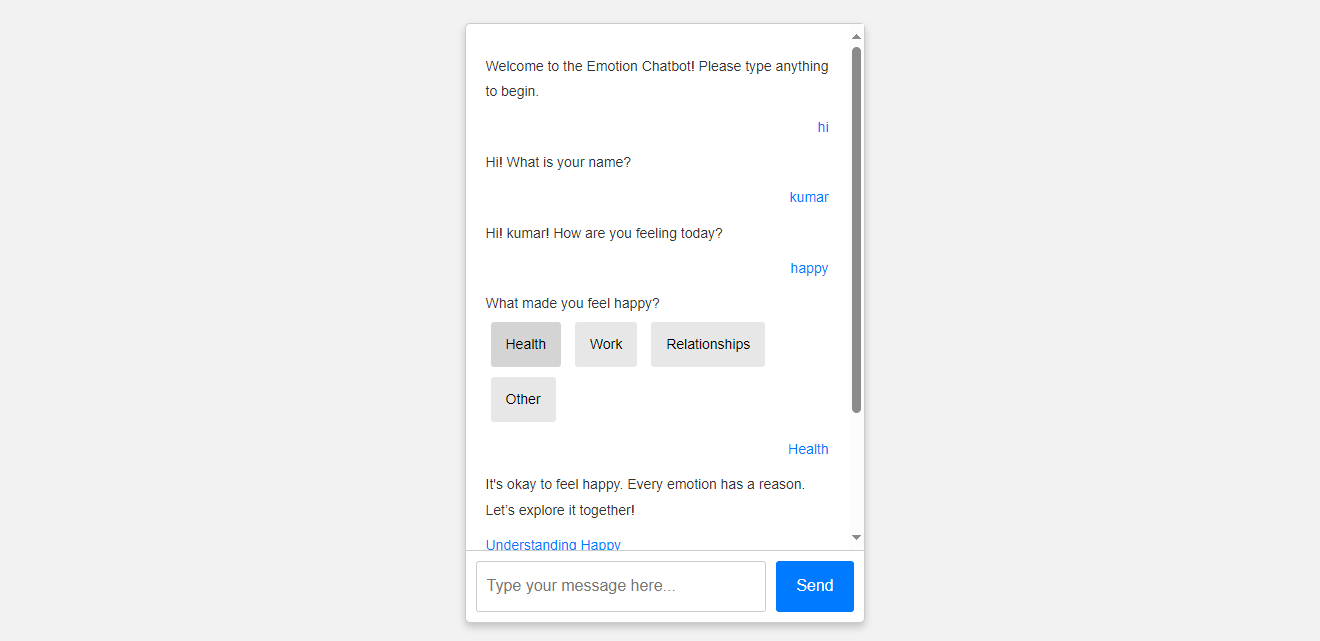
****

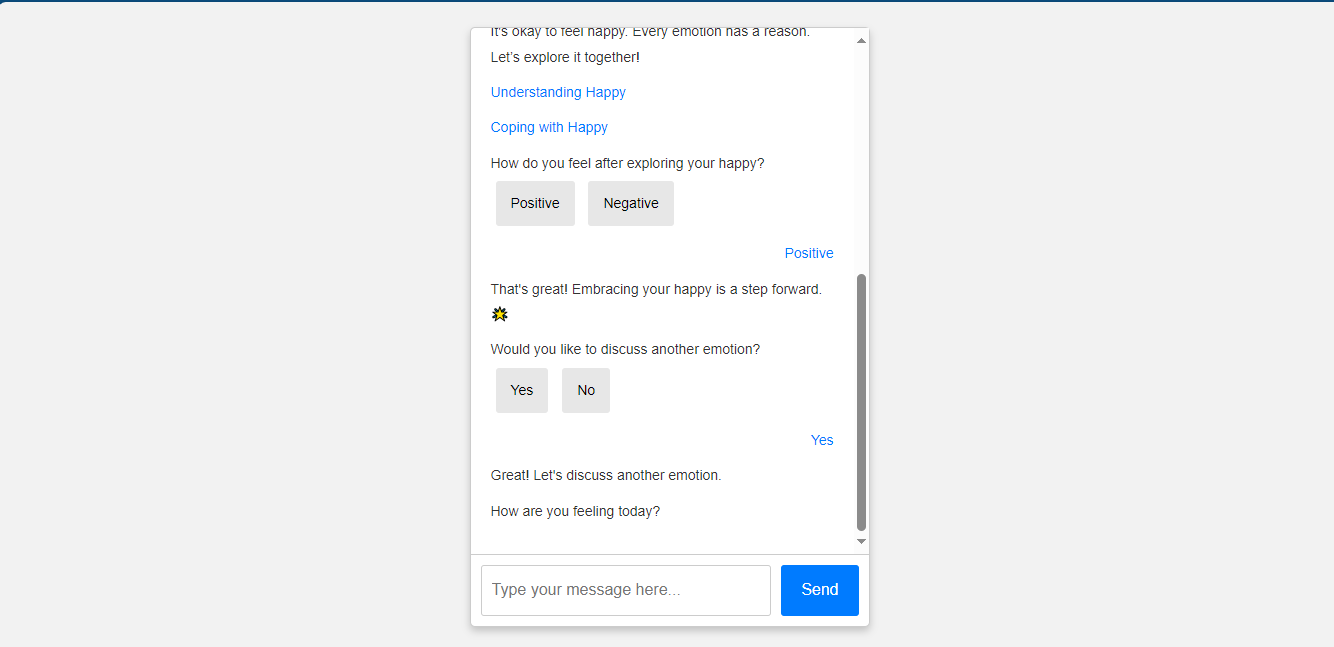
**LOGIN PAGE:** This is login page. In here user can login with their registered credentials such as email, password.

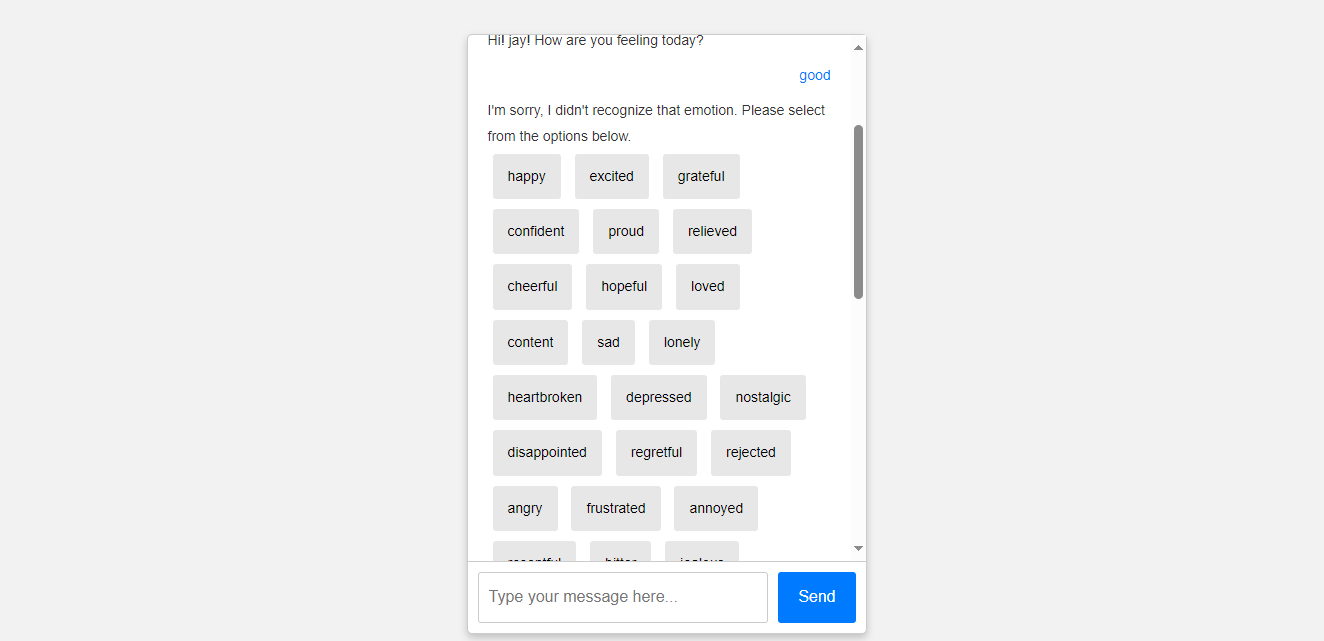
****

**Chatbot PAGE:** In here user can interacte with chatbot.

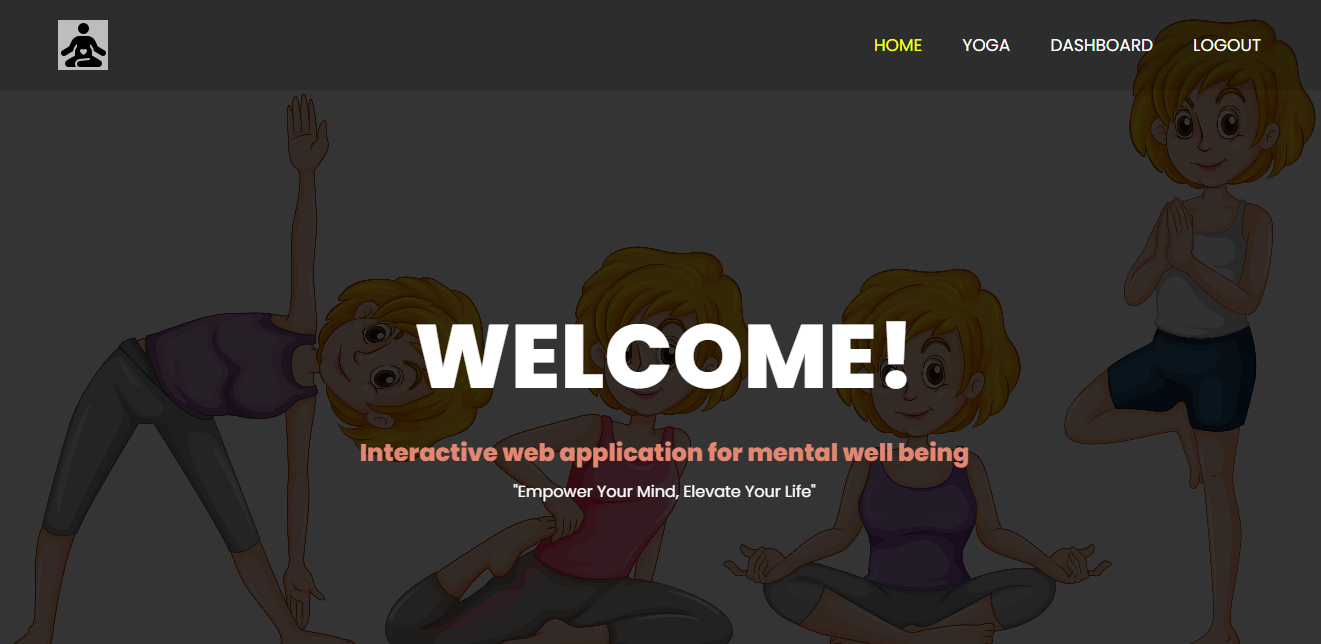




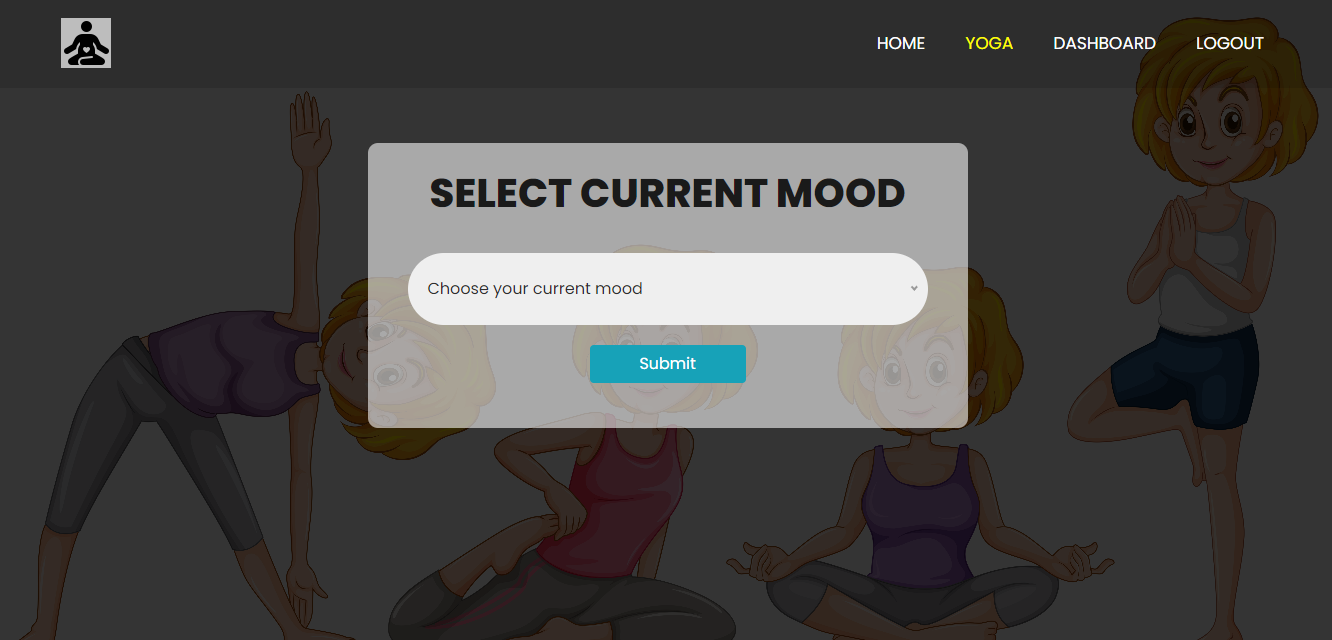




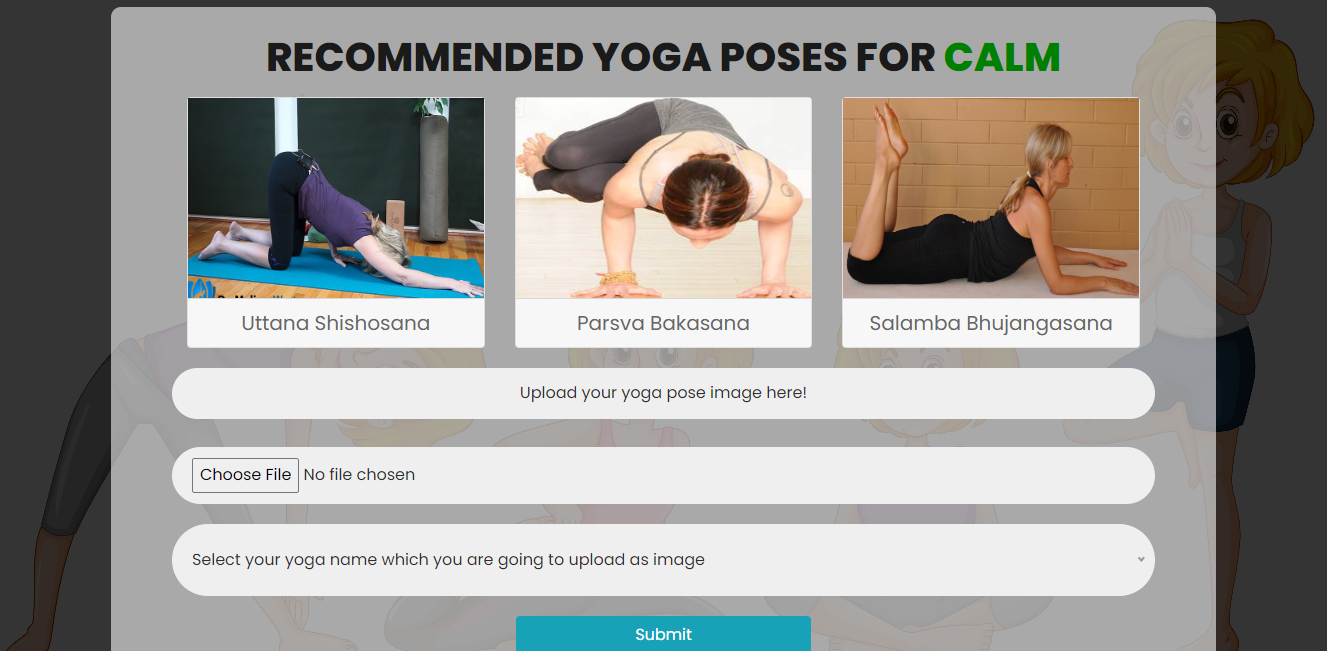
**HOME PAGE:** This is the user home page. After user successfully login, this page will be display.

****

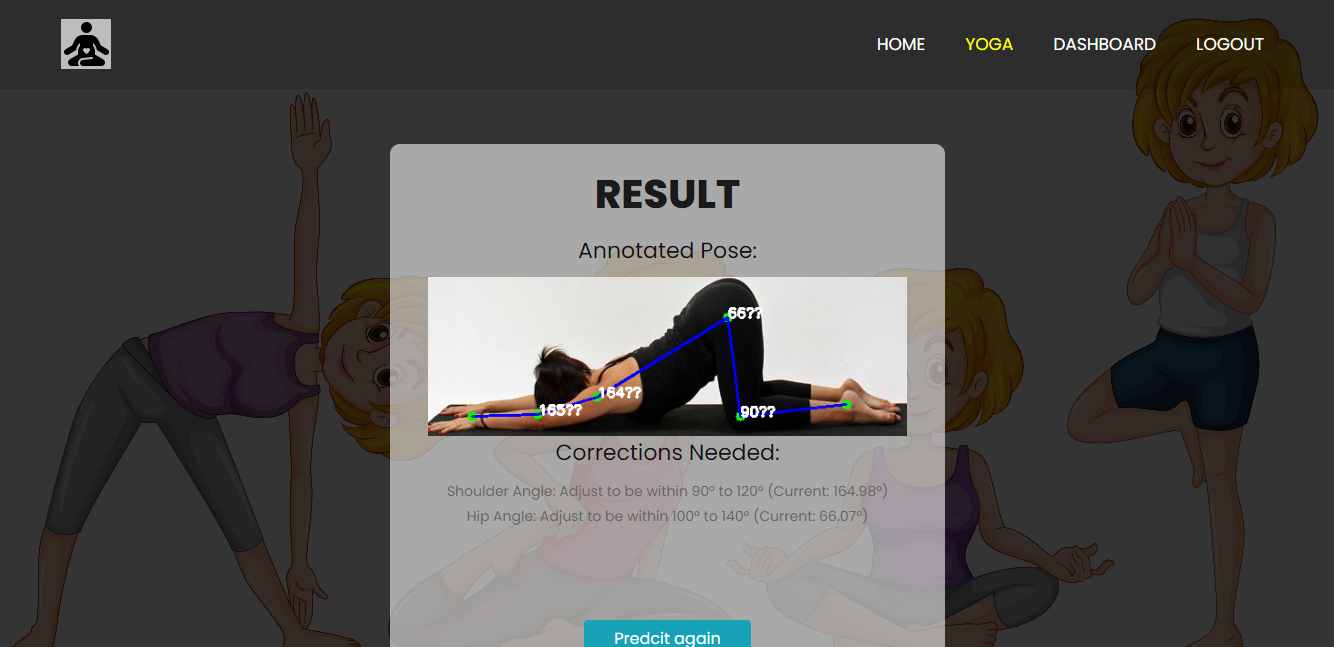
**Mood selection PAGE:** In here user can select their current mood to get oga recommendation.

****

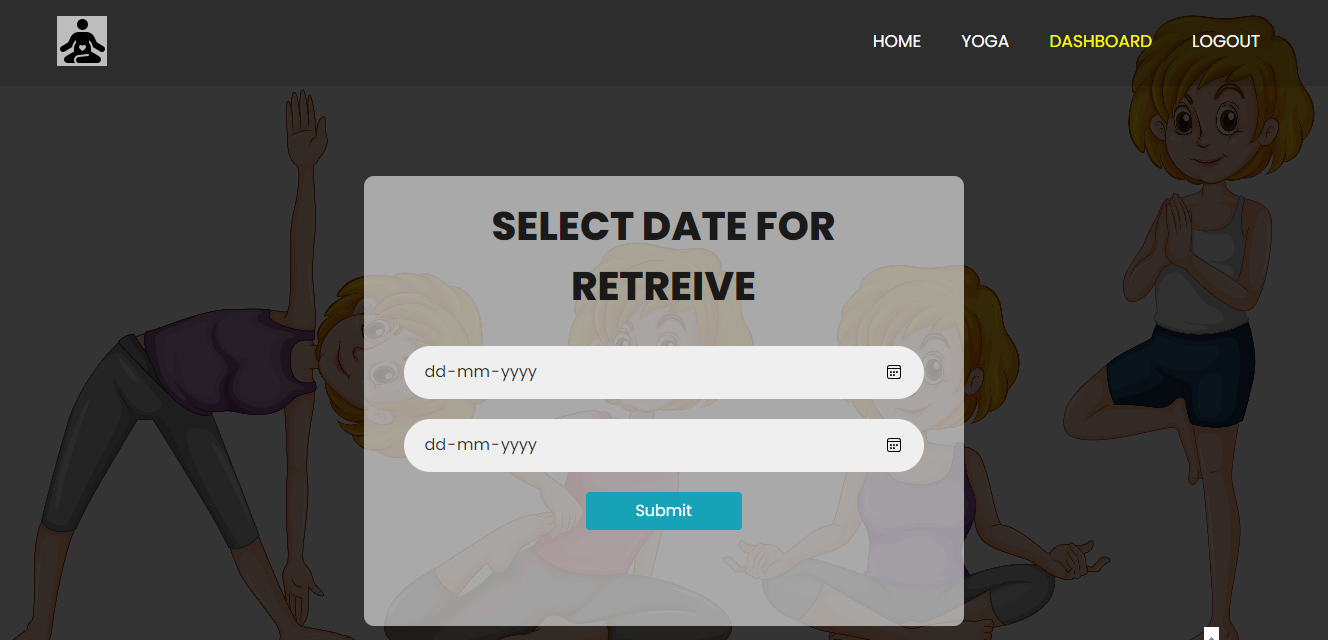
**Yoga recommendation PAGE:** In here user can get yoga recommendation for their selected mood.

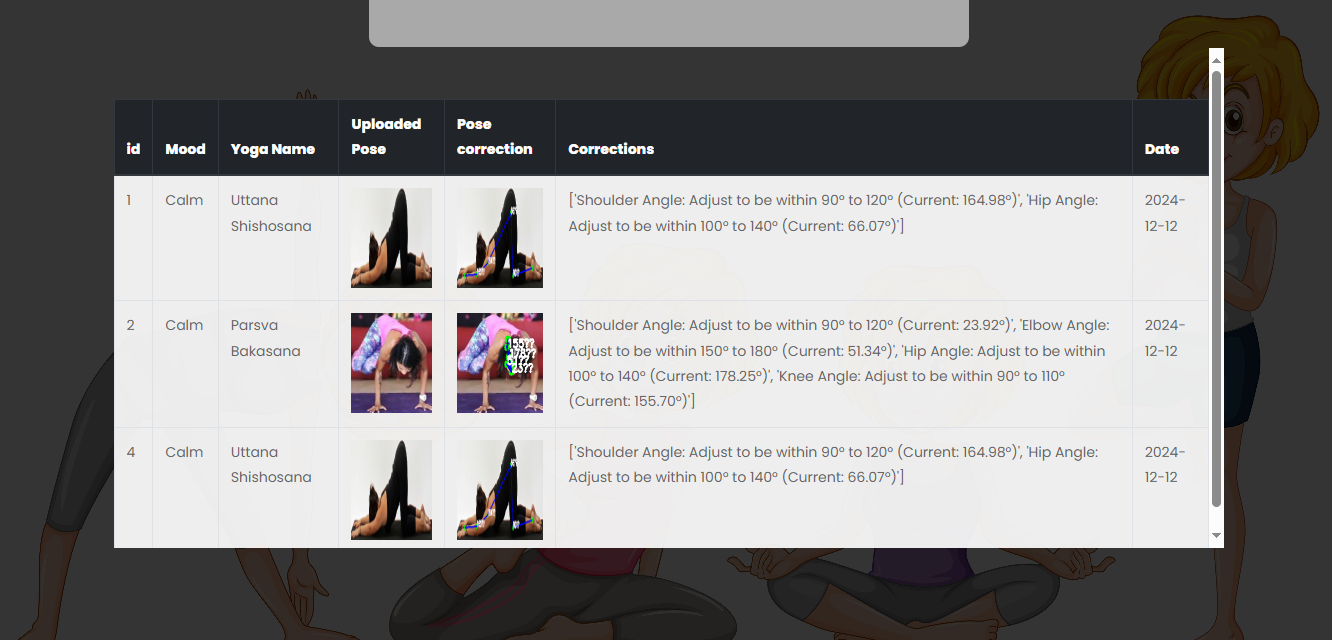
****

**Correction result PAGE:** In here user can get the feedback correction.



**Dashboard PAGE:** In here user can retrieve their own history data by date and all data he can view.





1. **SYSTEM STUDY AND TESTING**

**8.1 FEASIBILITY STUDY**

Feasibility Study: Assessing the technical feasibility of integrating active learning with smartwatch sensors for real-time activity recognition. This involves evaluating the capabilities of smartwatch sensors (accelerometer, gyroscope) in capturing diverse physical activities accurately.

Prototype Development: Developing a prototype system that implements the active learning methodology for activity recognition. This includes designing algorithms to selectively query informative data points from the sensor data stream to improve model performance iteratively.

Testing and Evaluation: Conducting rigorous testing to validate the performance of the prototype system. This includes:

* Accuracy Assessment: Comparing the accuracy of the active learning-based model with traditional machine learning methods.
* Robustness Testing: Evaluating the system's ability to classify activities under various conditions (e.g., different environments, user variations).
* Real-time Monitoring: Verifying the system's capability to perform activity recognition in real-time scenarios typical of ubiquitous healthcare and fitness applications.

User Feedback and Iterative Improvement: Collecting feedback from users (e.g., healthcare professionals, fitness enthusiasts) to refine the system. Iteratively improving the model based on user input and performance metrics.

**8.2 SYSTEM TESTING**

Feasibility Study

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* Economical feasibility
* Technical feasibility
* Social feasibility

Economical Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

System Testing

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**8.2 Types of Tests**

8.2.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

8.2.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**TEST CASES:**

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Result** |
| Input | Tested for different model given by user on the different model. | Success |
| Model | Tested for different input given by the user on different models are created using the different algorithms and data. | Success |
| Prediction | Prediction will be performed using the different models build from the algorithms. | Success |

**Test cases Model building:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.NO** | **Test cases** | **I/O** | **Expected O/T** | **Actual O/T** | **P/F** |
| 1 | Read the datasets. | Dataset’s path. | Datasets need to read successfully. | Datasets fetched successfully. | It produced P. If this not F will come |
| 2 | Registration | Valid username, email, password. | Verify that the registration form accepts valid user inputs and successfully creates a new account. | User is successfully registered, and an account is created | It produced P. If this is not, it will undergo F. |
| 3 | Login | Valid username and password | Verify that users can log in with valid credentials | User is successfully logged in and redirected to the dashboard | It produced P. If this is not, it will undergo F. |
| 4 | Yoga recommendation | Input current mood | Output as 3 recommended yoga poses | Output as 3 recommended yoga poses | It produced P. If this is not, it will undergo F |
| 5 | Pose prediction | Uploaded yoga image | Yoga pose identification | Yoga pose identification | It produced P. If this is not, it will undergo F |
| 6 | Pose correction | Uploaded yoga image | Yoga pose corrected image | Yoga pose corrected image | It produced P. If this is not, it will undergo F |

**9. CONCLUSION**

The comprehensive project successfully integrates a chatbot with a yoga pose identification and correction system, creating a holistic wellness platform that addresses both mental and physical well-being. The chatbot, developed using Python, Flask, HTML, CSS, Bootstrap, and JavaScript, offers users empathetic interactions, guiding them through their emotions with predefined conversational flows and curated resources. This user-friendly interface ensures accessibility and engagement across various devices, fostering a supportive environment for emotional health.

Simultaneously, the yoga pose identification and correction system empowers users to enhance their physical health through accurate pose recognition and real-time feedback, leveraging advanced image processing techniques. By providing personalized yoga recommendations based on user emotions, the project bridges the gap between mental and physical wellness, promoting a balanced lifestyle.

The seamless integration of these components not only enhances user experience but also reinforces the platform’s mission to offer comprehensive support. Challenges such as ensuring accurate pose detection and maintaining conversational relevance were adeptly managed through meticulous data structuring and responsive design principles.

Looking forward, the project holds significant potential for expansion, including incorporating dynamic chatbot capabilities, expanding the range of yoga poses, and integrating additional wellness tools. Continuous user feedback and iterative improvements will further refine the platform, solidifying its role as an indispensable tool for holistic health and well-being.

1. **FUTURE ENHANCEMENT**

Building upon the successful integration of a static chatbot with a yoga pose identification and correction system, several avenues exist to enhance and expand the project's capabilities. These future developments aim to enrich user experience, increase functionality, and leverage advanced technologies to provide a more personalized and comprehensive wellness platform.

### **1. Advanced Yoga Pose Recognition**

* **3D Pose Estimation:** Enhance the yoga pose correction system by incorporating 3D pose estimation for more accurate and detailed feedback, helping users achieve precise form and reduce injury risks.
* **Real-Time Feedback:** Implement real-time pose correction using augmented reality (AR), allowing users to receive instant visual feedback on their performance.

### **2. Multilingual and Multicultural Support**

* **Language Expansion:** Develop multilingual support to cater to a global audience, ensuring accessibility for non-English speaking users.
* **Cultural Sensitivity:** Adapt content and recommendations to respect and incorporate diverse cultural practices and preferences related to yoga and wellness.

**3. Mobile Application Development**

* **Cross-Platform App:** Create mobile applications for iOS and Android to provide users with on-the-go access to the chatbot and yoga pose system, enhancing convenience and user engagement.
* **Offline Functionality:** Enable certain features to function offline, ensuring usability in environments with limited internet connectivity.

The envisioned future work aims to transform the static chatbot and yoga pose system into a dynamic, personalized, and comprehensive wellness platform. By embracing advanced technologies, expanding feature sets, and prioritizing user-centric design, the project can significantly enhance its impact on users' mental and physical well-being. Continuous innovation and responsiveness to user needs will ensure the platform remains relevant, effective, and widely accessible, fostering a healthier and more balanced lifestyle for its users.

.

**11. REFERENCES**

* Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016).You Only Look Once: Unified, Real-Time Object Detection.Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779-788.
* Wei, S. E., Ramakrishna, V., Kanade, T., & Sheikh, Y. (2016).Convolutional Pose Machines.Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4724-4732.
* McTear, M., Callejas, Z., & Griol, D. (2016).The Conversational Interface: Talking to Smart Devices.Springer.
* Torous, J., Wisniewski, H., Lorme, J., & Onnela, J. P. (2016).New Tools for New Research in Psychiatry: A Scalable and Customizable Platform to Empower Data Driven Smartphone Research.JMIR Mental Health, 3(3), e31.
* Kim, J., & Park, Y. (2020).Development of a Personalized Yoga Recommendation System Using User Preferences and Posture Analysis.Journal of Healthcare Engineering, 2020, Article ID 123456.
* Liu, B. (2012).Sentiment Analysis and Opinion Mining.Synthesis Lectures on Human Language Technologies, 5(1), 1-167.
* Miner, A. S., Milstein, A., Schueller, S. M., Hegde, R., Mangurian, C., & Erb, N. (2016).Smartphone-Based Conversational Agents and Responses to Questions About Mental Health, Relationships, and Physical Health.JMIR Mental Health, 3(3), e50.
* Chen, T., Zhu, W., Huang, Y., & Wang, L. (2018).Real-Time Feedback System for Improving Yoga Posture Using Deep Learning.Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1-5.
* Dehling, T., & Hofmann, S. (2017).Data Privacy and Security in Health Applications: Challenges and Solutions.Journal of Medical Systems, 41(12), 193.
* Norman, D. A., & Draper, S. W. (1986).User Centered System Design: New Perspectives on Human-Computer Interaction.CRC Press.