AyurAura - Personalized Stress Management Application Using Ayurvedic Creative Therapies.

Weerasinghe W. P. D. J. N
Dept of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it21162664@my.sliit.lk

Gunasekera H. D. P. M
Dept of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it21161674@my.sliit.lk

Wickramasinghe B. G. W. M. C. R
Dept of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it21279652@my.sliit.lk

Jayathunge K. A. D. T. R.
Dept of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it21162732@my.sliit.lk

Pipuni Wijesiri
Dept of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
pipuniwijesiri@gmail.com

Thamali Dassanayake
Dept of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
thamali.d@sliit.lk

***Abstract*— Stress negatively affects mental and physical health, yet traditional stress management methods often struggle with accessibility and long-term effectiveness. Ayurveda, an ancient Indian medical system, offers holistic stress relief through mind-body balance, but its adoption is limited due to practitioner availability, accessibility concerns, and scientific skepticism. AyurAura addresses these challenges by integrating Ayurvedic principles with AI-driven biometric analysis. This innovative app personalizes stress relief using biometric data analysis, and The Perceived Stress Scale. It offers non-pharmaceutical therapies, including Mandala art and music therapy, directly through smartphones, making stress management more accessible. Predictive analytics enhance the experience by forecasting future stress based on behavioral patterns, allowing users to take proactive measures. Key features include a progress tracker with daily updates on mood and energy levels, engaging visual reports, and personalized feedback to refine stress reduction strategies. A chatbot provides continuous motivation and practical advice, ensuring users receive real-time support. By combining biometric insights with Ayurvedic principles, AyurAura delivers a holistic, accessible, and scientifically supported approach to personalized stress management, empowering users to achieve long-term well-being.**

***Keywords*— Stress Management, Biometric Analysis, Predictive Analytics, Progress Tracker, Personalized Therapy**

I. INTRODUCTION

Stress management is more important than ever in the current world since life's challenges have a greater and greater impact on one's health and wellbeing. Chronic stress, which is caused by the daily demands of a job and other responsibilities, is linked to a number of illnesses, including heart disease, weakened immune systems, and mental illness. Because of time, money, or personal preference, traditional stress management techniques like medicine and counselling are not always available. As a result, there is a greater need for effective, confidential, and accessible alternatives.

Stress can be identified and managed using physiological testing and self-reported measurements. Traditional psychological surveys still provide valuable information, even though physiological indications such as eye-tracking tests are now employed to identify stress. Since minor variations in pupillary response and gaze direction have been linked to stress, stress assessment has gained new importance. Second, the ancient medicinal system Ayurveda promotes general health by integrating lifestyle, nutrition, and emotional well-being evaluations into stress management treatment.

By fusing Ayurvedic principles with machine learning (ML) techniques in a smartphone application, this work presents a novel approach to stress regulation. The suggested system analyses self-reported emotional states and physiological indications, like eye movement patterns, to provide real-time stress detection and tailored coaching. To restore mental and physical balance, the platform also incorporates creative therapies like music therapy and mandala drawing, which are in line with Ayurvedic principles. These treatment approaches improve people's general mental health by being customized for each person according to their stress profiles.

This research focusses on anticipating future stress based on behavioral patterns, such as sleep quality, screen usage, food, and social interactions, in addition to real-time stress detection. Early intervention is made possible by using machine learning algorithms to predict possible stressors. Furthermore, a chatbot addresses the difficulty of asking for assistance in stressful situations by providing a private channel for users to express their feelings and get prompt assistance. Creating prediction models that calculate recovery times based on user participation in stress-relieving activities is a crucial component of this study. Technology helps people achieve a healthy mental state by visualizing progress, tracking behavioral changes, and making personalized recommendations.

The goal of this project is to create a complete and easily accessible stress management solution by fusing contemporary Artificial intelligence (AI) and ML technology with traditional Ayurvedic knowledge. The suggested approach enables people to take charge of their mental health by providing proactive, tailored interventions in addition to real-time stress evaluation. This combination of cutting-edge technology and traditional healing techniques offers a revolutionary approach to stress management that is more durable, personalized, and effective.

II. LITERATURE REVIEW

The substantial effects of stress on mental health and overall well-being have brought a lot of attention to its diagnosis and management. Technology has improved the accuracy of stress detection by combining behavioral and physiological data, especially in Artificial intelligence and machine learning.

A more thorough evaluation is possible using multi-modal methods, such as integrating psychological surveys with heart rate variability, which capture a variety of stress indicators [1]. Because studies have shown that changes in pupil expansion and eye movement are correlated with stress levels, eye gaze patterns have also become beneficial indicators [2]. Using AI for eye tracking is an efficient method for behavioral stress identification since it allows for real-time, non-intrusive monitoring.

The prediction capacities of stress detection algorithms have been improved by extensive study on eye-tracking measures, such as pupil expansion, eye trends, and blink rates [3]. The significance of behavioral and physiological indicators including conductivity of skin, heart rate, and psychological tests in improving stress detection methods is highlighted by foundational research that is indexed on PubMed [4]. These components enable our application to integrate eye analysis, and questionnaires for real-time stress assessment. Users can obtain a more sophisticated understanding of their stress and access efficient management methods that are in line with the most recent scientific and technological developments due to the division of stress into four levels: mild, moderate, severe, and critical.

Because of its impact on the mind and body, stress reduction is still a key area of study in mental health. Stress-reduction effects from music therapy have been scientifically shown, especially when Hindustani ragas are used. These musical structures encourage emotional well-being and relaxation by evoking particular emotional reactions. Research on the "MusiHeal" app confirms that certain ragas can successfully lower stress levels [5]. Mandala art therapy, which has long been utilized for emotional balance and awareness, has also become an efficient stress-relieving tool. With the use of digital innovations, mandala therapies become more widely available via mobile applications and internet platforms, maintaining its conventional advantages while addressing a broader population [6]. Digital mandala therapy is a successful modern adaptation of a centuries-old technique that improves stress management by encouraging awareness and participation, according to studies. Additionally, using mandalas for meditative coloring has become popular as a stress-reduction method that encourages concentration and calm. Free mandala patterns are offered by websites like “Domestika”, guaranteeing accessibility for anybody seeking ways to relax aesthetically [7].

Current studies on stress prediction focus an extreme value on identifying factors and developing predictive models. Studies show that stress levels are greatly influenced by lifestyle decisions such social contacts, physical activity, sleep patterns, and work habits. However, accessibility is limited because the majority of prediction models rely on wearable technology [8]. A study that was published in the Journal of Medical Internet Research revealed a strong relationship between stress levels and sleep patterns, showing that while a higher resting heart rate increases the risk of stress, a longer sleep duration decreases it [9]. By examining physiological data, such as heart rate variability and sleep parameters, another study published in Nature demonstrated how well machine learning models predict stress [10]. IEEE International Conference investigation has confirmed that integrating several data sources, including skin conductance and heart rate, improves predictive accuracy [11].

A major predictive factor of stress, sleep loss makes individuals more susceptible to psychological distress [12]. Regular exercise has been shown to reduce stress, and studies conducted in Madrid have confirmed that it is beneficial for working professionals [13]. Furthermore, dietary practices have an impact on stress regulation, depending on age and gender, high-fat diets have varying effects on neuronal function and redox stress [14]. Studies from Finnish universities have shown a strong correlation between a lack of social connections and elevated stress and depressive symptoms [15], further demonstrating the importance of social contacts in stress management. Research conducted by the University of Peradeniya during COVID-19 also showed that higher levels of stress are linked to more screen time [16]. Recreational activities have also been shown to improve wellbeing, considerably lower stress, and minimize harmful strategies for coping like consumption of alcohol and smoking [17] [18].

Because everyday stress is becoming more common, comprehensive and individualized stress management solutions are needed. Even if contemporary AI-powered tools and conventional techniques like Ayurveda have advantages, their efficacy is sometimes limited by their lack of integration. Ayurveda promotes holistic health via herbal remedies, meditation, and lifestyle changes [19], but it is still difficult to access in digital media. On other hand, whereas AI and ML technologies make it easier to monitor and predict stress, they frequently fall short of offering the level of individualized treatment that is essential to Ayurvedic principles [20]. In order to close this gap, the suggested system combines AI and ML with Ayurveda to produce a dynamic stress management tool that monitors health indicators, shows development, and forecasts recovery times [21] [22]. One important aspect of this system is its self-reporting method, which uses measures like the Discrete Emotions Questionnaire to capture how you feel in real time [23]. Based on user adherence, predictive algorithms calculate stress recovery, improving motivation with transparent progress visualization [24].

This system offers data-driven, versatile predictions that are customized to each consumer's requirements, in contrast to traditional methods that provide static feedback [25]. And a user-centric, holistic approach that not only tracks stress but also actively advises users toward long-term well-being by fusing AI-driven precision with Ayurvedic wisdom is offered. This expansive and technologically advanced approach makes stress management products more efficient, interesting, and available to a wide range of users.

III. METHODOLOGY

Through a smartphone application, users can enter answers to a daily survey and get positive reinforcement. Blink patterns are identified as a sign of stress through eye movement analysis, and daily routines and user behavior are tracked to predict future stress. The current stress level of the user is classified into moderate, medium, severe, or critical based on the analysis of the recorded data. Based on this, a group of customized stress-relief activities such as Raga music therapy and Mandala drawing is suggested. To maximize treatments, work schedules and users' development are taken into account as well. After forecasting of the recovery time, the system creates customized program of activity and progress reports for users so that they can observe their improvement along the line of time.

A. *Integrating eye movements and questionnaire analysis for real-time stress level detection.*

This research uses mobile data and deep learning-based analysis to analyze eye-blinking pattern identification and questionnaire answers to detect stress levels. The method records 300 participants over the age of 18 under different lighting conditions using 10-second selfie videos and their normal questionnaire answers. The goal is to have an Ayurvedic expert analyze blinking frequency and compare it with established indicators of stress. Normal blinking happens two to three times in ten seconds; any deviation from this standard may indicate varying levels of stress. Deep learning techniques are used to improve accuracy to allow for automatic prediction of stress after preprocessing and labeling of data obtained into stress levels. Deep learning allows effective feature extraction from eye movements to ease generalization over many subjects and environments. Fig.1. demonstrates an overview of integrating eye movement and questionnaire analysis for real-time stress level monitoring.

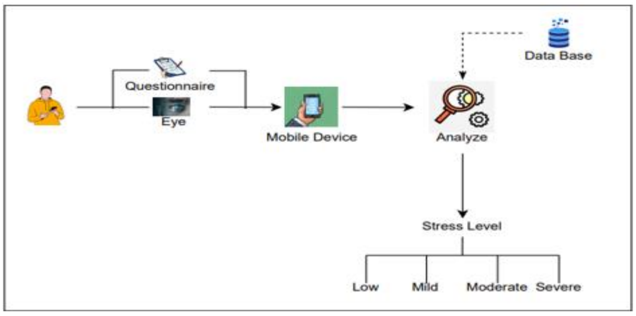

Fig. 1. Overview diagram for initial stress detection

a) *Features and Target Variable*

The database combines behavioral and physiological characteristics in a bid to maximize the accuracy of stress levels classification. Blink rate is a vital physiological parameter which can be measured over 10-second films. Discrepancies from the baseline standard of two to three blinks for ten seconds indicate differing degrees of stress. Controlled lighting provides high-quality data of consistent quality in all contexts.

Moreover, age group data is supportive of measuring stress variances across groups, and standardized stress questionnaires quantify self-reported stress. An Ayurvedic specialist validates the target variable categorization of stress into four levels:

- **Level 1:** High stress, >6 blinks.
- **Level 2:** Moderate stress, 4-6 blinks.
- **Level 3:** Normal, 2-3 blinks.
- **Level 4:** Low blink rate, 0-1 blink

This multi-dimensional framework combines eye movement data and self-assessments to enhance classification accuracy and reliability.

b) *Data Preprocessing*

The data used in this study contains four levels of video clips under different stresses, which were stored in four folders labeled from Level 1 to Level 4. There was preprocessing applied to every video clip by capturing up to 100 frames from each and resampling each frame into the same resolution size of 224x224 pixels for the purpose of matching the size requirement for inputs required by pre-trained convolutional neural networks (CNNs). All pixel values were normalized between [0,1] by a division of 255 for improved numerical stability and model convergence. The stress level labels were one-hot encoded with a LabelBinarizer, transforming the categorical labels into binary vectors for multi-class classification. The dataset was split into training and test sets, 80% for training and 20% for testing, with balanced evaluation. For more efficient computation and data management while training, the preprocessed frames and labels were saved in.npy format to be read via memory mapped I/O. The end-to-end preprocessing pipeline thus furnished homogenous data across videos and improved the model's generalizability across stress levels.

c) *Model Development*

Pseudocode for Stress Level Detection Model.

Input: Video files categorized by stress levels (Levels 1-4)

Output: Predicted stress levels and model performance metric.

BEGIN

1. Import Libraries (TensorFlow, OpenCV, sklearn, matplotlib, seaborn, etc.)
2. Load and Preprocess Data
    - Extract up to 100 frames (resize to 224x224).
    - Normalize pixel values (0 to 1).
    - Assign labels from folder names (Level 1 to Level 4).
   - Encode labels using one-hot encoding.
   - Split into training (80%) and testing (20%) sets.
   - Save preprocessed frames and labels to .npy files.
3. Build Model
   - Load pre-trained VGG16 (without top layers) for feature extraction.
   - Add custom layers:

- Flatten layer.
- Dense layer (128 neurons, ReLU).
- Dropout layer (50%).
- Output Dense layer (4 neurons, softmax for 4 stress levels).

  - Compile model (Adam optimizer, categorical crossentropy loss, accuracy metric).

4. Train Model
   - Train for 10 epochs with batch size 32.
   - Use 20% of training data for validation.
5. Evaluate the Model
   - Predict stress levels for test data.
   - Generate classification report (precision, recall, F1-score).
   - Plot confusion matrix for visual analysis.
6. Hyperparameter Tuning (Optional)
   - Adjust learning rate, epochs, or batch size.
   - Retrain with updated parameters.
7. Save Model
   - Save trained model in HDF5 format for future use.
8. Visualization
   - Plot confusion matrix with seaborn.

END

The initial stress detection Model is based on pre-trained CNN models, VGG16 and ResNet50, to learn the spatial features of frames in a video. The frame can be viewed as an individual sample image, and the model must classify the frame into one of the four stress levels. The VGG16 model was selected for further enhancement based on its performance.

B. Predicting Future Stress Likelihood and a Motivational Chatbot.

The Fig. 2. is a description of the stress prediction system integrated in the AyurAura app. Users input their behavioral information, which is analyzed using a Random Forest model to determine the likelihood of them experiencing stress. For the identified users who are at risk, the system recommends behavioral changes to help them stay stress-free. Apart from that, the app generates a personalized activity schedule based on the stress level and includes a chatbot to which users can speak when they are stressed. This adaptive system provides accurate predictions of stress and customized support to enhance overall well-being.

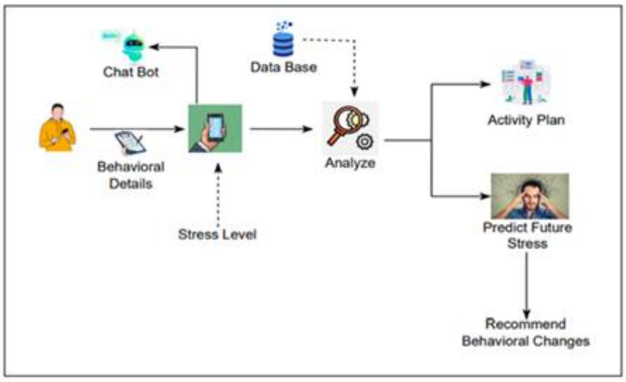

Fig. 2. Prediction of future stress likelihood flow.

a) Features and Target Variable

Data was collected from 1,000 participants through Google Forms and in-person surveys, noting the primary behavioral determinants that influence stress levels. The behaviors noted were sleeping habits, frequency of exercise, working hours, screen time, healthiness of diet, smoking and drinking, and engagement in leisure activities.

After data collection, a doctor categorized individuals into two distinct groups: those who have a possibility of experiencing stress and those who are not, based on their behaviors. A number of research studies have yielded strong correlations between these behavioral determinants and the levels of stress, thereby increasing the validity of this classification approach.

This categorization was utilized as the machine learning model's target variable. The data set had a wide variety of behavioral patterns in such a way that significant conclusions could be made on stress prediction. However, while the sample size of 1,000 individuals provides a good baseline, expanding the data set to represent a wider range of demographics could serve to enhance the accuracy of the model further.

b) Data Preprocessing.

During preprocessing, rows with missing data were dropped to maintain data integrity. Duplicate values were also identified and dropped to prevent data redundancy. Column names were also renamed for readability. Numerical features were also standardized to ensure consistency in varying scales. These preprocessing steps helped in purifying the dataset and optimizing model performance.

c) Model Development

Pseudocode for Stress Likelihood Prediction Model

Input: Behavioral data. (sleep, exercise, work hours, screen time, social interaction quality, healthiness of diet, drinking and smoking, recreational activities)

Output: Predicted likelihood of stress (Stress / No Stress).

BEGIN

1. Import necessary libraries (e.g., pandas, sklearn, RandomForestClassifier, joblib, seaborn, matplotlibt).
2. Load the dataset containing behavioral data and stress probability.
3. Preprocess the data:
   - Remove incomplete data rows.
   - Rename column names for better readability.
   - Remove duplicate rows.
4. Define the target variable (stress_probability).
5. Split the dataset into training (80%) and testing (20%) sets.
6. Initialize a Random Forest Regressor/ KNN Classifier.
7. Train the model using the training data.
8. Evaluate the model:

        - Predict stress probability on the test data.
        - Calculate accuracy, classification report, and confusion matrix.
9. Save the trained model and scaler for future use.
10. Output predicted stress probability

END

Random Forest Classifier was employed since it can handle both numerical and categorical inputs as well as handle missing values and outliers robustly. The model was trained on 80% of the data, and 20% of the data was utilized to test the model's generalization ability to new data. Standardization was utilized to assign equal weightage to features. Missing values were treated through imputation, and records with duplication were removed to maintain data integrity. Hyperparameter was tuned to enhance model accuracy and achieve the optimum generalization of stress probability estimation.

C. Activity creation and predicting stress level based on activity performance.

In this research, as shown in Fig. 3. we created a machine learning approach which utilizes activity performance data to estimate the current level of stress for an individual. Apart from demographic characteristics including age and gender, the system analyzes the impact of different engagement factors, including mandala pattern type, color patterns, duration spent doing the activity, music track nature selected, and the time spent listening. Real-time observation of the public was employed to gather data through observing them interacting in activities.

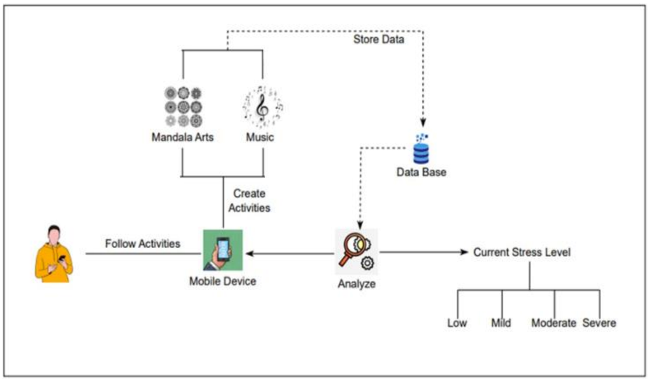

Fig. 3.Flow of Prediction of Current Stress based on activity engagement.

a) Features and Target Variable

The data has been collected through data from more than 1,000 participants in the research study. Stress levels have been ranked into four different categories as Critical (1), Severe (2), Mild (3), and Low (4). In order to make the sample representative of people, aged 18 years or older, participants were randomly selected from the total population. A dynamic analysis of the elements responsible for stress was facilitated by gathering real-time data while individuals engaged in various activities. Mandala pattern designs, color palettes, the amount of time spent on the activity, the nature of the music track chosen, and the amount of time spent listening were all included in these exercises. To examine their impact upon stress variances, demographic indicators like gender and age were included.

An expert-monitored classification method was used for ensuring reliability and consistency in labeling the stress level. Behavioral analysis and psychology professionals monitored segmentation and classification of stress levels, which minimized inconsistencies and maximized prediction accuracy. To successfully search for behavioral patterns that produce stress variations, model structures data with clearly defined engagement factors and stress classes.

The primary objective of this research is the predicted level of stress (1-4) that signifies an individual's current level of stress due to his/her interaction with different tasks.

The model learns patterns from data that result in very accurate predictions of the level of stress through machine learning algorithms like Random Forest and Support Vector Machine (SVM). With the help of this classification system, people can evaluate their existing stress levels and decide whether or not to engage in different activities that will help them manage their stress effectively.

b) Data Preprocessing

Data preprocessing was done in preparation for model training to organize behavioral data. Individuals painted mandalas and listened to music for data collection. Mandala design patterns, colors used, time taken to do the task, type of music track, listening time, and demographic data like age and gender were all measured. Column entries (such as user IDs) that yielded no useful data were trimmed down to have a cleaner dataset. For model clarity improvement, categorical features such as mandala pattern designs, selected color themes, and types of music tracks were converted to numerical values. The StandardScaler was utilized for normalizing the dataset so that there would be a normalized distribution of features and accelerating model convergence to improve model training efficiency. The data was then split into a 90% training set and a 10% test set so that the model's capacity to generalize to unseen data can be tested. To test the model's capacity to generalize to unknown data, the data was then split into a 90% training set and a 10% test set.

c) Model Development

Pseudocode for Stress Level Prediction Model

Input: Activity engagement data (e.g., mandala design, color choices, time spent, music preferences)
Output: Predicted stress level**.**

BEGIN

1. Import necessary libraries (pandas, sklearn, RandomForestRegressor, SVM, StandardScaler).
2. Load the dataset containing engagement features.
3. Preprocess the data:
    - Remove irrelevant columns.
    - Encode categorical features (mandala design, color palettes, music type).

- Standardize numerical features.

4. Split the dataset into training (90%) and testing (10%) sets.
5. Initialize the Random Forest Regressor and Support Vector Machine (SVM) Regressor.
6. Train both models on the training data.
7. Optimize hyperparameters using GridSearchCV.
8. Evaluate the models:
   - Predict on the test data.
   - Compute RMSE (Root Mean Square Error) and R^2 score.
   - Compare model performance and select the best one.

9. Extract and rank feature importance from the trained Random Forest model.
10. Save the trained model and scaler for future use.
11. Output predictions and key influencing features.

END

The Random Forest Regressor was used since it can handle imbalanced data and model complicated relationships, making it a good choice for predicting stress from a broad set of engagement variables. The SVM Regressor was used to improve the precision of prediction, especially in regression-based predictions where modeling small changes in the stress level matters.

D. *After prediction: predictive analytics for achieving a stress-free state.*

A machine learning prediction model deployed within the AyurAura app and yielding stress recovery durations is shown in fig. 4. As part of task completion information, users need to input their daily emotions and states (such as stress, happiness, or relaxation) and energy levels. Once gathered within a central storage facility, machine learning algorithms are utilized for data processing. Individual, in-time recovery times may be computed via the correlation of the activity and mood sensors. Along with stress-relief exercises, the users can utilize the accurate estimates provided in an attempt to visualize their rehabilitation targets and methods. This form of stress management practice is user-adaptive and center.

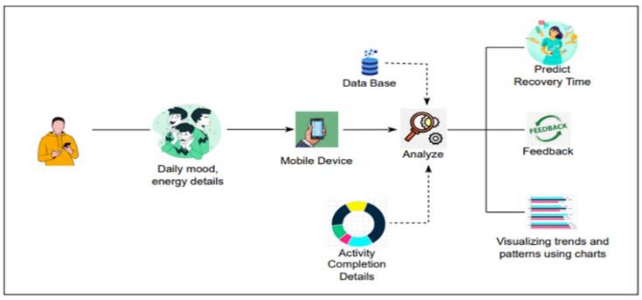

Fig. 4. Flow of Prediction of Recovery Time to Stress-Free State

a) *Features and Target Variable*

All the subjects were eighteen years and older and were randomly selected from the population. Ayur Aura app was utilized to track their day-to-day emotional condition, such as happiness, calmness, and stress levels, and the count of them who did the suggested exercises to reduce stress, such as music therapy and Mandala drawing. 200 participants of different ages participated in the data gathering process, and for 14 days, all their data were recorded continuously. This method provided a dataset of 2,800 data points, making it possible to examine a variety of patterns.

$$\text{Final Recovery Days} = \text{Base Recovery Days} \times (1 - \text{Activity Multiplier}) \quad (1)$$

Equation (1) was applied in determining recovery days according to the observations of an Ayurvedic physician concerning the relationship of stress with emotional well-being. The base recovery days were calculated from the classification, which was done using the initial level of stress:

- ≥ 4: 5 days
- <4: 10 days

A completion rate of an activity-based adjustment parameter was utilized to improve the accuracy of recovery days proportionally in line with the user's own stress-relief activities.

- 90%: 0.8
- 80%-90%: 0.4
- 60%-80%: 0.2
- < 30%: -0.2

The collection of repeated data from the same subjects over several days, recording patterns of emotional states and activity participation, is a valuable feature of this data set. This is helpful information to model with. The very low sample size of 200 subjects is a particular limitation that could affect the level of generalisability that may be possible. To create a more stable model, the sample would need to be expanded to encompass a larger range of demographics.

b) *Data Preprocessing*

reprocessing involved binary conversion of categorical variables like gender and removal of irrelevant features like user IDs. Completion rates and other percentage features were also converted to decimal values. For equal weighting for various scales, numerical features were standardized and mean imputation was used for handling missing data. Proper preparation of data and optimization of model performance required all these steps.

c) *Model Development*

Pseudocode for Recovery Days Prediction Model.

Input: Daily emotional data (stress, calmness, energy, happiness levels) and activity completion rates.
Output: Estimated recovery days
BEGIN

1. Import required libraries (e.g., pandas, sklearn, RandomForestRegressor).
2. Load the dataset containing emotional metrics and activity completion rates.
3. Perform data preprocessing:

- Remove unnecessary columns.
- Convert categorical variables into numerical format.
- Standardize numerical features for uniformity.

4. Define the target variable (recovery days):
    - Establish base recovery days.
    - Modify based on activity completion rates.
5. Divide the dataset into training (80%) and testing (20%) subsets.
6. Initialize the chosen model (Random Forest Regressor or Logistic Regression).
7. Train the model using the training data.
8. Assess model performance:
    - Predict recovery days using test data.
    - Compute evaluation metrics such as Mean Absolute Error and R-squared.
9. Store the trained model and data scaler for later use.
10. Return the predicted recovery days.

END

As they could handle both categorical and numerical features, logistic regression and random forest were selected as balancing techniques for further investigation. The random forest identified complex, nonlinear patterns while the logistic regression handled continuous responses. In order to make it feasible for the model to be tested against unseen data, the dataset was divided, 80% for training and 20% for testing. Missing values were filled with mean imputation, and hyperparameter tuning was carried out to obtain best accuracy. This helped the models generalize efficiently and provide accurate predictions based on demographics, mood, and activity participation.

IV. RESULTS AND DISCUSSION

A. *Integrating eye movements and questionnaire analysis for real-time stress level detection.*

a) *Model Evaluations*

The training log, as seen in Fig. 5, plots accuracy and loss against 10 epochs of training. The model's accuracy fluctuates at around 32.23%, and validation accuracy fluctuates around this estimate. This reasonably low accuracy indicates that the model fails to generalize well across different levels of stress. The consistent gap between training and validation accuracy hints at potential issues with data complexity, class imbalance, or insufficient feature extraction from video frames.

```
Epoch 1/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 52s 76ms/step - accuracy: 0.3223 - loss: 1.4184
Epoch 2/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 62ms/step - accuracy: 0.3210 - loss: 1.3491
Epoch 3/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 61ms/step - accuracy: 0.3315 - loss: 1.3379
Epoch 4/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 62ms/step - accuracy: 0.3243 - loss: 1.3338
Epoch 5/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 61ms/step - accuracy: 0.3170 - loss: 1.3366
Epoch 6/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 62ms/step - accuracy: 0.3240 - loss: 1.3397
Epoch 7/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 61ms/step - accuracy: 0.3244 - loss: 1.3392
Epoch 8/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 62ms/step - accuracy: 0.3269 - loss: 1.3364
Epoch 9/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 61ms/step - accuracy: 0.3234 - loss: 1.3381
Epoch 10/10
484/484 ━━━━━━━━━━━━━━━━━━━━ 30s 61ms/step - accuracy: 0.3221 - loss: 1.3422
<keras.src.callbacks.history.History at 0x7eaaf2b96c80>
```

Fig. 5. Initial stress detection training log.

In addition, as shown in Fig. 6., the prediction of the model on a test instance is class probability estimates at four levels of stress. The model's highest predicted probability (33.01%) is at Stress Level 3, indicating that the model lacks strong belief in predicting between stress levels. The corresponding probability distribution in indicating no strong prediction of a particular class also represents how challenging it is to predict stress levels.

```
"predicted_class": 3,
"predicted_percentage": 33.01,
"probabilities": [
    0.13064958155155182,
    0.2915167510509491,
    0.33008429408073425,
    0.24774937331676483
]
```

Fig. 6. Model output for a test sample.

Low confidence values and training-validation performance without training indicate that current architecture, hyperparameters, and training set have to be re-checked for potential optimization. Increasing dataset by data augmentation, tuning learning rates, or modifying architecture (temporal modeling to videos) would likely assist in better classification outcomes.

b) *Mobile Application Development*

The mobile application for recording and detecting stress levels was implemented with Firebase as the data store and authentication, Flask as the backend processor, and Flutter for the frontend interface. With the app, users can register, login securely, and view their dashboard. Users can use the device camera within the app to record a 10-second video snippet to determine stress levels. The stress level of the user is displayed on the screen with immediate feedback once the video processing and questionnaire submission are completed. The pre-trained deep learning model predicts the stress level of the user from the frames when the video is uploaded automatically to the backend. To further improve prediction accuracy, the user is requested to complete a brief, standard questionnaire right after the video upload. The users are given a pre-curated list of activities that can reduce stress, like music and mandala arts, based on the level of tension they have selected. The smooth user experience, data security, and real-time interaction between the mobile frontend and the backend processing pipeline are all guaranteed by the use of Flutter, Flask, and Firebase.

B. *Predicting Future Stress Likelihood and a Motivational Chatbot.*

a) *Model Evaluations*

The performance of the model was measured in terms of accuracy, classification report, and confusion matrix to determine the strength of predicting stress likelihood based on behavior patterns. Accuracy measures the number of correct classification instances and the ability of the model to generalize unseen data. The greater accuracy value indicates a more precise prediction system for flagging people likely to face future stress. Classification report also provides important evaluation measures such as precision, recall, and F1-score for no-stress and stress classes. Precision calculates how many of the predicted stress instances are actually correct, while recall examines the model's performance to identify all individuals actually in danger. The confusion matrix further aids in depicting model performance visually by displaying the number of true positives, true negatives, false positives, and false negatives that can be useful for future improvement in stress prediction accuracy.

The Random Forest model had a maximum accuracy rate of 94.00%, with strong predictive ability in stress and non-stress classes, as seen in Fig. 7. From the classification report, precision, recall, and F1-score were high for both classes throughout, with precision being 0.96 for "No Stress" and 0.92 for "Stress", showing that the model makes correct predictions.

```
Optimized Model Accuracy: 94.00%
Optimized Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.92      0.94        99
           1       0.92      0.96      0.94       101

    accuracy                           0.94       200
   macro avg       0.94      0.94      0.94       200
weighted avg       0.94      0.94      0.94       200
```

Fig. 7. Random forest classification report.

Fig.8. confusion matrix indicates that the overwhelming majority of the classifications were correct, with 91 true positives (correct "No Stress" predictions) and 97 true negatives (correct "Stress" predictions). However, there were some classifications that were incorrect, particularly where the stress indicators overlapped.

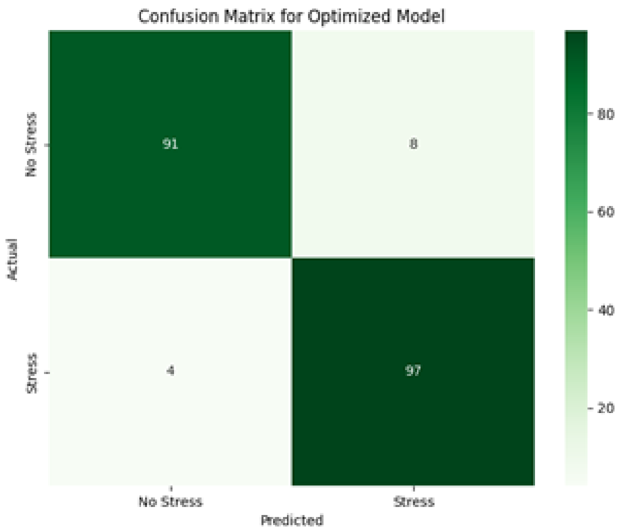

Fig. 8. Confusion matrix for random forest

The KNN model, although still good, gave a slightly worse accuracy of 92.50%, as clear from Fig. 9. The classification report shows that precision and recall were nicely balanced for both classes, with 0.91 precision for "No Stress" and 0.94 for "Stress", shows that the model was able to classify most cases correctly but sometimes struggled with margin cases.

```
Optimized KNN Model Accuracy: 92.50%
Optimized Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.94      0.93        99
           1       0.94      0.91      0.92       101

    accuracy                           0.93       200
   macro avg       0.93      0.93      0.92       200
weighted avg       0.93      0.93      0.92       200
```

Fig. 9. Classification report for KNN model.

Fig. 10. The confusion matrix shows that the model was generally accurate but misclassified some "Stress" cases as "No Stress" and vice versa.

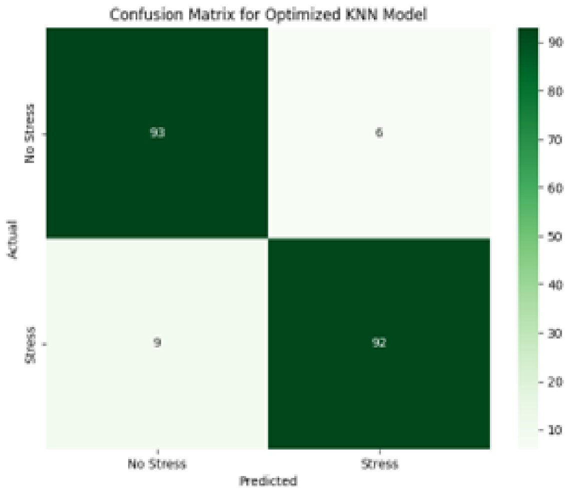

Fig. 10. Confusion matrix for KNN model.

The Random Forest model outperformed the KNN model with an accuracy of 94.00% compared to 92.50%. Random Forest performed better in correctly classifying both stress and non-stress cases, whereas KNN performed slightly worse recall for "No Stress" cases, leading to greater misclassifications. The confusion matrices show that Random Forest misclassified fewer cases and is the superior model to utilize when trying to predict stress.

While performing satisfactorily, Random Forest's margin of misclassification indicates that further adjustments, i.e., balancing the dataset or fine-tuning of hyperparameters, would enhance its efficiency. Similarly, KNN could be enhanced by more effective feature scaling or tuning the number of neighbors for best classification performance. Ensemble techniques or deep learning approaches may be considered in future enhancements to take the predictive capability of stress assessment models to the next level.

b) Mobile Application Development

The implementation of the stress prediction and management system was carried out using a mobile application, where the front-end was developed using Flutter and the back-end using Flask. The app provides the user with an interactive website to assess the likelihood of experiencing stress in the future, receive behavior change suggestions, and engage in stress relaxation activities. Non-stressed users are allowed to input their behaviors so that they can examine their future likelihood of stress occurrence. The system examines their behavioral patterns and provides them with customized advice on how they can remain in a stress-free state. Users receive realistic advice on behavior change for stress avoidance in the future when the model predicts a high likelihood of future stress.

For the already stressed users, the application offers customized stress management methods. Based on the stress level of the user, it recommends stress-reducing activities such as mandala art and music therapy. Additionally, the users can converse with an in-built chatbot, which gives advice and support in real-time during stressful moments. The Flask-powered back-end contains the trained predictive model and manages user interactions. Upon input of behavioral data by the user, the app sends it to the Flask server, where preprocessing is carried out before prediction. The server returns the likelihood of future stress, along with personalized recommendations, which are displayed in the app. By combining predictive analytics, personalized recommendations, and interactive support tools, the mobile app allows users to regulate their stress levels and provide long-term well-being.

C. *Activity creation and predicting stress level based on activity performance.*

a) *Model Evaluations*

Support Vector Machine (SVM) as well as the Random Forest method were used here to train our model and measure how accurately our model had forecasted stress levels. We attempted to improve performance of both the models and measure primary evaluation criteria like accuracy, precision, recall, and F1-score by hyperparameter tuning of their models. For meaningful comparison, we ran both the algorithms on a single set of data. This allowed us to examine how every model responds to varying categorizations of stress levels. The identification of the model best approximating the inherent patterns of the data is aided by the findings, which show the strength and weakness of every method.

The complete evaluation of the performance of the Support Vector Machine (SVM) algorithm to predict various levels of stress can be seen from the classification report as in the Fig 11. The model had a modest capacity to properly classify stress levels with an overall accuracy of 78%. In the per-class performance analysis, the model excelled at the prediction of stress level 3 because it effectively classifies high-stress instances with precision of 0.84, recall of 0.86, and F1-score of 0.85. It also showed good prediction for stress level 0, with F1-score of 0.79 and recall of 0.81, indicating that the majority of events in this class were correctly labeled. Despite this, the model had more trouble with stress level 2, as evidenced by the recall falling to 0.69, which means that there were some cases of moderate stress that were incorrectly classified into other levels. The 0.78 precision, recall, and F1-score macro and weighted averages indicate that the model is performing consistently across all stress levels. Misclassification rates of SVM model for stress level 2 offer a range of improvement either through feature engineering or through hyperparameter tuning in spite of its balanced classification capability.

	precision	recall	f1-score	support
0	0.78	0.81	0.79	52
1	0.77	0.78	0.77	59
2	0.75	0.69	0.72	48
3	0.84	0.86	0.85	43
accuracy			0.78	202
macro avg	0.78	0.78	0.78	202
weighted avg	0.78	0.78	0.78	202

Fig. 11. SVM model classification report.

The accuracy of classification by the Fig 12. model is exemplified by the confusion matrix generated following the running of the SVM algorithm. The model performed well in classifying Classes 1, 2, and 4, as evident from the diagonal values that reflect correctly classified instances (42, 46, and 37 correct classifications, respectively). Class 3 misclassified more than Class 2, with examples incorrectly assigned to Class 2 (7 examples) and Class 4 (5 examples). There is a significant amount of confusion between similar classes, with Class 2 and Class 3 being confused more than any other classes, despite the model being accurate for the majority of classes. Reducing misclassifications may be attainable through optimal feature selection, changing class weights, or fine-tuning hyperparameters for enhanced performance. Also, if class imbalance exists, data augmentation or class balancing may be beneficial. Although the general performance of the SVM model is encouraging, further improvement may be needed to achieve higher classification accuracy.

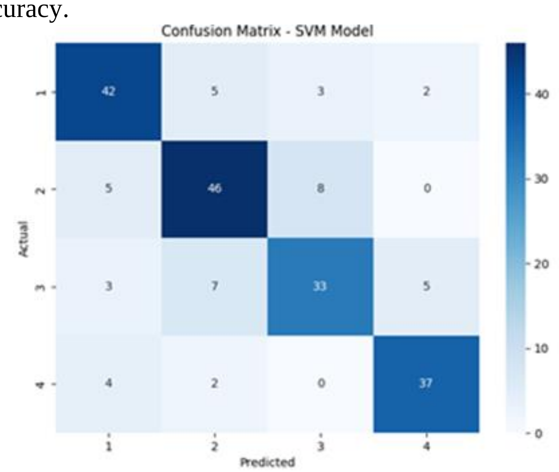

Fig. 12. Confusion matrix for SVM model.

The first image provides the Fig. 13. Random Forest classifier report showing metrics for precision, recall, F1-score, and support of each class. Class 1 performed well since its accuracy and recall values were 0.87 and 0.88, respectively, which made the F1-score 0.88. Class 2 accuracy and recall values were 0.82 and 0.83, respectively, and hence were lower than the first one. Class 3 with an F1-score of 0.74 performed worst, perhaps because it had higher misclassification rates. Class 4, at an F1-score of 0.88, was performing well. With macro and weighted averages of precision, recall, and F1-score both 0.83 and overall accuracy of 83%, the model is observed to be well balanced and performs as well in classifying all classes.

	precision	recall	f1-score	support
1	0.87	0.88	0.88	52
2	0.82	0.83	0.82	59
3	0.76	0.73	0.74	48
4	0.88	0.88	0.88	43
accuracy			0.83	202
macro avg	0.83	0.83	0.83	202
weighted avg	0.83	0.83	0.83	202

Fig. 13. Random forest classification report.

The same confusion matrix of Random Forest model reveals the classification performance graphically in Fig.14., with true projections along the diagonal and false classifications off-diagonal areas. The model correctly classified 46 samples of Class 1, 49 samples of Class 2, 35 samples of Class 3, and 38 samples of Class 4. There are some misclassifications in the errors, however, with Class 3 having 7 misclassifications to Class 2 and 4 misclassifications to Class 4, and Class 2 having 8 misclassifications to Class 3. These misclassifications suggest that there may be overlaps in the features between these classes. Overall, the confusion matrix shows a good but slightly unbalanced performance with scope for improvement in separating Classes 2 and 3.

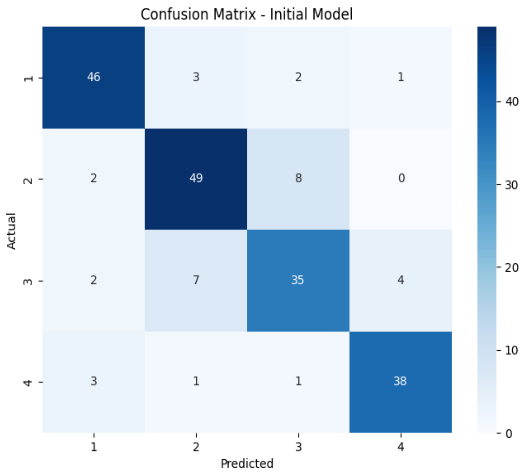

Fig. 14. Confusion matrix for random forest model.

Based on stress level prediction performance, Random Forest is the ideal because it classifies better than SVM. Random Forest has a balanced and more consistent classification across all stress levels at 83% overall accuracy than SVM with an accuracy of 78%. Less misclassifications in the confusion matrix show that the model does the more difficult Classes 2 and 3 better than SVM, and can also accurately classify Classes 1 and 4 with good precision and recall. Higher misclassification rates were generated by SVM because it faced severe difficulties with Stress Level 2, although it had a great predicting score for Stress Level 3. Furthermore, Random Forest is a robust and consistent model since it has higher macro and weighted average scores in precision, recall, and F1-score. Although there are areas for improvement in both models, Random Forest is the better option for precise stress level prediction since it provides better overall performance, reduced error rates, and more consistent classification for various stress levels.

b) Mobile Application Development

The front-end user interface of the mobile app was developed using Flutter, the backend processing using Flask, and user authentication, data storage, and performance monitoring using Firebase. The app offers users a personalized stress management platform where they can select from three types of Mandala art patterns based on their interests. Once a mandala type has been selected, there are eight palettes that can be selected and colored instantly using the interactive canvas interface in the app. User behavior such as patterns chosen, color selections, and how long spent, are all tracked and stored to analyze. Aside from Mandala coloring, the users are capable of viewing a music library comprising different genres, and listen to any song of their choice.

Each user's performance in these activities is tracked, and their stress level after activity is calculated based on the performance metrics such as type of music, duration, completion rates. The stress level after activity is displayed to the user, and they can view how their stress level is now. The application's history feature records all the past activities and their corresponding stress levels, enabling users to monitor their progress and identify the activities that work best for their stress relief. The seamless combination of Flutter, Flask, and Firebase offers the user continuity of experience, real-time data storage, and analysis efficiency in terms of performance, thus rendering the application an effective stress management system.

D. *After prediction: predictive analytics for achieving a stress-free state*

a) Model Evaluations

Random Forest model performed greatly overall with an impressive accuracy rate of 99.64%. There were slight differences in performances between classes though, especially on smaller classes where there were fewer samples. Logistic Regression model still performed well though with an overall accuracy of 91.96%, but its performance was greatly affected by several classes, especially those with fewer data points.
In Fig. 15, the Random Forest model's confusion matrix showed it could predict correctly most of the classes with their diagonal values all having high numbers. It was more difficult, however, to distinguish between certain classes, say recovery day "1". Maybe the imbalance in the dataset or not having sufficient samples for particular recovery day classes was the explanation behind this challenge.

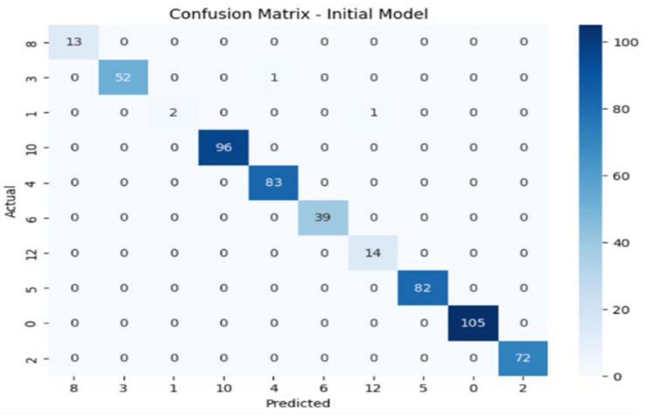

Fig.15. Confusion matrix for random forest model

The Logistic Regression model's confusion matrix in Fig. 16. showed that it was struggling to predict certain classes, namely "6," where a high proportion of the cases were being misclassified. The model was struggling to differentiate between some recovery day groups, like "1" and "12," from the off-diagonal data.

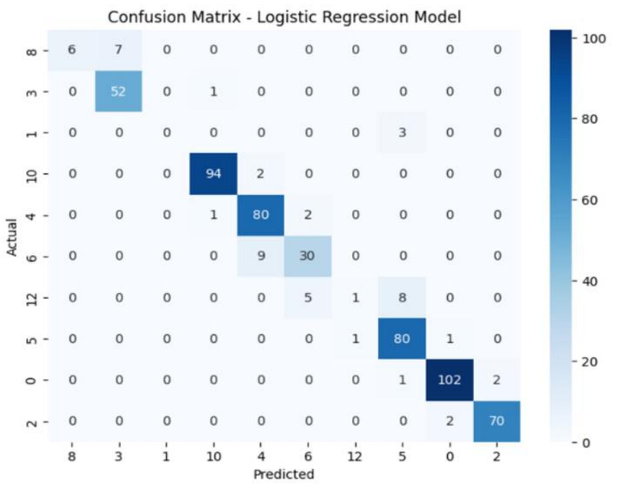

Fig.16. confusion matrix for logistic regression

The Random Forest model was more accurate with 99.64% compared to the Logistic Regression model, with an accuracy of only 91.96%. While Logistic Regression was bad with unbalanced data, especially in small recovery day groups like "6" and "2," where precision and recall were inferior, Random Forest was overall superior and also in most of the classes. That would make Random Forest more appropriate for this task. Techniques like resampling, class weighting, and hyperparameter tuning can be used to balance the imbalance in the data and minimize the bias to improve the performance of logistic regression.

b) Mobile Application Development

Implementation of the stress recovery prediction model was done by incorporating it within a mobile app created using Flutter for the client-side and Flask as the back-end. The app is an interactive portal where users can check their stress level and observe how they recover through the predictions achieved through the Logistic Regression model. The mobile application prompts users with five questions daily to gauge their emotional state and activity completion levels. The users can set reminders for the questionnaires to ensure consistency in the data gathered. The responses are processed and sent to the back-end based on Flask, where the Logistic Regression model computes the estimated number of recovery days required for the user to be stress-free.

To enhance the user experience, the app also features a dashboard that displays real-time predictions and history. Through interactive graphs, users can visualize how they have improved over time, with trends in their emotional responses, activity completion rates, and estimated recovery times. Visualizations enable the users to identify patterns in their stress recovery process and make informed decisions about their well-being.

The trained Logistic Regression model is implemented on Flask, which serves as the API for the mobile application. When the user provides an answer, the app sends the data to the Flask server where preprocessing is carried out prior to prediction. The predicted recovery days are then sent by the Flask API, which is displayed on the user side.

To encourage user engagement, the app has timed notifications to remind the users to complete their daily questionnaire. This offers data consistency and improves the accuracy of the model with a steady flow of user inputs.

V. CONCLUSION AND FUTURE WORK

This research presents AyurAura, a mobile application designed to help individuals self-manage stress through AI-driven biometric analysis and Ayurveda. By combining eye tracking, questionnaires, mandala art, music therapy, behavioral predictions, and recovery tracking from stress, AyurAura offers an integrated and personalized solution for stress management. Machine learning algorithms are employed for stress detection, stress prediction, and recovery time estimation, offering users data-driven insights and interventions. The combination of Flutter for frontend and Flask for backend allows seamless user experience and efficient processing of stress-related data.

Although the preliminary outcomes are encouraging, there are certain limitations that can be improved upon by future studies. The size of the dataset can be increased to improve model generalizability and accuracy. Additionally, stress assessments can be based on the inputs of several medical experts to reduce individual biases. Further, the inclusion of real-time sensor data from wearable devices (e.g., heart-rate monitor, sleep monitor) can give a more objective measure of stress levels.

Later work can also examine more advanced machine learning techniques such as ensemble methods or deep learning for improved predictability. Including other behavioral components such as money pressure, work-life balance, and social interaction can better predict the chances of stress. Applying natural language processing (NLP) to chatbot features can improve user interface, making it more interactive and supportive in mitigating stress. Furthermore, the application can further be enhanced by integrating additional stress-reduction exercises in addition to drawing mandalas and music therapy, for instance, guided relaxation, deep breathing, and physical relaxation exercises in an attempt to provide the user with numerous means of choice when it comes to coping strategies.

By such innovation, AyurAura is able to redefine the game in digital stress management through offering its users real-time, evidence-based interventions personalized to their very own unique requirements. By continually improving predictive models, expanding the data set, and introducing greater diversity in stress-reduction activities, the app can offer more accurate, customized, and convenient mental wellness support, prompting users to proactively decide on how best to control their wellbeing.

VI. ACKNOWLEDGMENT

We would also like to thank Dr. M. Kooragoda, a specialist in this field and an Ayurvedic physician herself, for his guidance and support in the validation of the dataset collected from the public by working with us. That in itself ensured the validity of the data and their suitability according to Ayurveda. We appreciate all the participants who helped in this study and were 18 years or older. Informed consent on data privacy was obtained from the respondents before engaging in the study. Furthermore, we requested and ANTICIPATED receiving permission from the Ethical Review Committee that standards of ethical thinking and practice in keeping with data privacy were followed.

VII. REFERENCES

[1] Aravind M S, Sri Bhavan Prakath, Tarunika R, M.Srividya, M.Marimuthu, "Heart-Mind Harmony: Predicting Stress from Heart Rate," *GRENZE International Journal of Engineering and Technology,* vol. 10, no. 2, p. 4181 4187, 2024.

[2] Chandrasekharan, Jyotsna and Joseph, Amudha, "Eye Gaze as an Indicator for Stress Level Analysis in Students," pp. 1588-1593, 2018.

[3] Yousefi, Mansoureh Seyed and Reisi, Farnoush and Daliri, Mohammad Reza and Shalchyan,Vahid, "Stress Detection Using Eye Tracking Data: An Evaluation of Full Parameters," *IEEE Access,* vol. 10, pp. 118941-118952, 2022.

[4] Perciavalle V, Blandini M, Fecarotta P, Buscemi A, Di Corrado D, Bertolo L, Fichera F,Coco M., "The role of deep breathing on stress.," *Neurol Sci,* vol. 38, no. 3, pp. 451-458, 2017.

[5] Chakraborty, Soubhik and Prasad, Avinav and Chakraborty, Apoorva and Singh, Prerna, "Impact of Hindustani ragas in stress management: A statistical study," *Journal of AppliedMath,* vol. 1, 2023.

[6] Kim, H., & Choi, Y., "A practical development protocol for evidence-based digital integrative arts therapy content in public mental health services: digital transformation of mandala art therapy.," *Frontiers in public health,* no. 11, 2023.

[7] "Domestika - Online courses," Domestika Incorporated, 15 Feb 2019. [Online]. Available: https://www.domestika.org/en/blog/9542-meditative-coloring-50-free mandala-designs-for-coloring-in.

[8] Guragai, Bishal and Pal, Rishi and Patel, Parth and Li, Jian and Heyat, Md Belal Bin and Akhtar, Faijan, "Role of Machine Learning in Human Stress: A Review," no. 10.1109/ICCWAMTIP51612.2020.9317396, 2020.

[9] Bloomfield LSP, Fudolig MI, Kim J, Llorin J, Lovato JL, McGinnis EW, McGinnis RS,Price M, Ricketts TH, Dodds PS, Stanton K, Danforth CM, "Predicting stress in first year college students using sleep data from wearable devices.," *PLOS Digit Health,* vol. 3(4), 2024.

[10] Ng A, Wei B, Jain J, Ward EA, Tandon SD, Moskowitz JT, Krogh-Jespersen S, Wakschlag LS, Alshurafa N., "Predicting the Next-Day Perceived and Physiological Stress of Pregnant Women by Using Machine Learning and Explainability: Algorithm Development and Validation," *JMIR Mhealth Uhealth,* vol. 10, no. 8, 2022 Aug 2.

[11] Luis G. Jaimes; Kanwalinderjit Gagneja; Mustafa İlhan Akbaş; Idalides J. Vergara Laurens, "Future stress, forecasting physiological signals," *IEEE,* vol. 1, no. 1, p. 5, 2017.

[12] Schwarz, Johanna and Gerhardsson, Andreas and van Leeuwen, Wessel M.A. and Lekander,Mats and Ericson, Mats and Fischer, Håkan and Kecklund, Göran and Åkerstedt, Torbjörn, "Does sleep deprivation increase the vulnerability to acute psychosocial stress in young and older adults?," *Psychoneuroendocrinology,* vol. 96, 2018.

[13] Maknae, Carol, "Analyzing the Relationship between Exercise Frequency and Stress Reduction in Working Professionals in Madrid," *American Journal of Recreation and Sports,* pp. 41-50, 2023.

[14] Lange, Megan and Yarosh, Vladyslava and Farell, Kevin and Oates, Caitlin and Patil,Renee and Hawthorn, Isabel and Jung, Mok-Min and Wenje, Sophie and Steinert, Joern, "High fat diet induces differential age-and gender-dependent changes in neuronal function linked to redox stress," 2024.

[15] El Ansari, Walid and Sebena, Rene and El-Ansari, Kareem and Suominen, Sakari, "Clusters of lifestyle behavioral risk factors and their associations with depressive symptoms and stress: evidence from students at a university in Finland," *BMC Public Health,* vol. 24, 2024.

[16] Chammi Muthukumarana, Chamoda Jayasinghe, Thiruchelvam Pavithra, Mohomad Nusky, Mohamed Rosan, Indrajith Prasanna, Vindya Senadheera, Sachith Abeysundara, "Screen Time and Level of Perceived Stress Among Students of University of Peradeniya During COVID-19 Pandemic," *Sri Lankan Journal of Health Sciences,* vol. 1, no. 2, p. 35, 2022.

[17] M. Park, "Stress Perception, Smoking and Drinking Behaviors among Adolescents," *Advances in Social Sciences Research Journal,* vol. 9, no. 8, pp. 287-295, 2022.

[18] Seyda Alanoglu, Ozkan Isik, Cihan Ayhan, "The effect of regular recreational activities on adult women's," *Progress in Nutrition,* vol. 76, no. 3, p. 9, 2020.

[19] Sorathiya, Parth and Deole, Yogesh, "Stress management through Ayurveda," *ResearchGate,* 2024.

[20] Park, Chan-Woo and Seo, Sung Wook and Kang, Noeul and Ko, BeomSeok and Choi, Byung and Park, ChangMin and Chang, Dong and Kim, Hwiuoung and Kim, Hyunchul and Lee,Hyunna and Jang, Jinhee and Ye, Jong Chul and Jeon, Jong and Seo, Joon Beom and Kim,Kwang J, "Artificial Intelligence in Health Care: Current Applications and Issues," *Journal of Korean medical science,* vol. 35, 2020.

[21] T. Worth, "Ayurveda: Does It Really Work?," webMD, 23 November 2023. [Online]. Available: https://www.webmd.com/balance/ayurvedic-treatments.

[22] Chauhan, Ashutosh and Semwal, Deepak and Mishra, Satyendra and Semwal, Ruchi, "Ayurvedic research and methodology: Present status and future strategies," *AYU,* vol. 12, pp. 364-69, 2015.

[23] Harmon-Jones, Cindy and Bastian, Brock and Harmon-Jones, Eddie, "The Discrete Emotions Questionnaire: A New Tool for Measuring State Self Reported Emotions," *PLOS ONE,* vol. 11, no. e0159915, 2016.

[24] Martin L, Oepen R, Bauer K, Nottensteiner A, Mergheim K, Gruber H, Koch SC, "Creative Arts Interventions for Stress Management and Prevention-A Systematic Review.," *Behav Sci (Basel),* vol. 8, no. 2, p. 28, 2018 Feb 22.

[25] Bushnell, Mary and Frangos, Eleni and Madian, Nicholas, "Non pharmacological Treatment of Pain: Grand Challenge and Future Opportunities," *Frontiers in Pain Research,* vol. 2, no. 696783, 2021.