# Project Report

## 1. INTRODUCTION

### 1.1 Project Overview

The project, "Sleep Oracle Anticipating Health and Lifestyle," aims to revolutionise sleep analysis by integrating lifestyle factors. This overview provides a snapshot of the project's significance in enhancing well-being.

### 1.2 Purpose

The purpose is to develop a comprehensive system that goes beyond traditional sleep analysis. By considering lifestyle factors, the Sleep Oracle provides personalised insights for users to improve sleep patterns and overall health.

## `````` 2. LITERATURE SURVEY

### 2.1 Existing problem

Current sleep analysis systems lack a holistic approach, often neglecting crucial lifestyle factors. The literature survey identifies this gap and emphasises the need for a more integrated solution.

### 2.2 References

References include key works on sleep analysis, health datasets, and systems incorporating lifestyle factors. Notable sources are cited to provide a foundation for the project's development.

### 2.3 Problem Statement Definition

The problem lies in the inadequacy of existing sleep analysis tools to account for lifestyle variables. The Sleep Oracle seeks to address this by creating a unified system for comprehensive sleep insights.

## 3. IDEATION & PROPOSED SOLUTION

### 3.1 Empathy Map Canvas

The empathy map canvas explores user needs and pain points related to sleep and health. It serves as a foundation for ideation, guiding the development of features that cater to user perspectives.

### 3.2 Ideation & Brainstorming

Ideation involves collaborative brainstorming to generate innovative features. The team explores solutions considering data accuracy, user engagement, and advanced technologies for a cutting-edge system.

# 4. REQUIREMENT ANALYSIS

## 4.1 Functional requirement

Functional requirements detail user-centric features like sleep data input, personalized insights, and feedback mechanisms. Each requirement is linked to specific user stories for clarity.

## 4.2 Non-Functional requirements

Non-functional requirements encompass security measures, scalability, response times, and user interface responsiveness. Compliance with data protection regulations is emphasized.

# 5. PROJECT DESIGN

## 5.1 Data Flow Diagrams & User Stories

Data flow diagrams illustrate how information flows through the system. User stories capture specific interactions, detailing how users input data and receive personalized insights.

## 5.2 Solution Architecture

The high-level architecture includes components like data collection modules, machine learning algorithms, and a user-friendly interface. Integration points between these modules are outlined for clarity.

# 6. PROJECT PLANNING & SCHEDULING

## 6.1 Technical Architecture

Technical architecture details technologies chosen for the backend, frontend, and database. Considerations include data encryption, API integration, and system dependencies.

## 6.2 Sprint Planning & Estimation

Sprint planning involves breaking down features into tasks, estimating effort, and prioritizing based on user value. This iterative process ensures continuous delivery of valuable increments.

## 6.3 Sprint Delivery Schedule

A detailed schedule outlines deliverables for each sprint, including feature implementation, testing, and user acceptance.

## 7. CODING & SOLUTIONING
## (Explain the features added in the project along with code)

### 7.1 Feature 1

This feature includes a seamless interface for users to input sleep-related data. Code snippets demonstrate form validation, data submission, and integration with backend services.

### 7.2 Feature 2

Personalized Insight Generation

The code for personalized insight generation showcases the integration of machine learning models, considering sleep metrics and lifestyle factors. Visualization techniques may be highlighted.

### 7.3 Database Schema

The database schema outlines the organization of sleep-related data, ensuring efficiency in storage and retrieval. It includes tables for user profiles, sleep records, and analysis results.

## 8. PERFORMANCE TESTING

### 8.1 Performance Metrics

Performance testing metrics include accuracy, confusion matrix, and classification report. Scalability tests ensure the system can handle increasing loads.

## 9. RESULTS

### 9.1 Output Screenshots

Screenshots exhibit the system's output, featuring personalized insights presented in a user-friendly interface. Examples showcase various scenarios, including different lifestyle inputs.

**Logistic Regression:**

```
[76] accuracy

     0.6666666666666666

[77] confusion_matrix(y_test,y_pred)

     array([[ 7, 13,  0],
            [ 3, 37,  1],
            [ 0,  8,  6]])

[78] print(classification_rep)

                   precision    recall  f1-score   support

                0       0.70      0.35      0.47        20
                1       0.64      0.90      0.75        41
                2       0.86      0.43      0.57        14

         accuracy                           0.67        75
        macro avg       0.73      0.56      0.60        75
     weighted avg       0.70      0.67      0.64        75
```

**XGB Boost:**

```
[85] accuracy_score(y_test,y_pred1)

     0.9066666666666666

[86] confusion_matrix(y_test,y_pred1)

     array([[16,  2,  2],
            [ 1, 40,  0],
            [ 1,  1, 12]])

[87] print(classification_report(y_test,y_pred1))

                  precision    recall  f1-score   support

               0       0.89      0.80      0.84        20
               1       0.93      0.98      0.95        41
               2       0.86      0.86      0.86        14

        accuracy                           0.91        75
       macro avg       0.89      0.88      0.88        75
    weighted avg       0.91      0.91      0.91        75
```

## Decision Tree:

```
[90] accuracy_tree = accuracy_score(y_test, y_pred_tree)
     print("Decision Tree Accuracy:", accuracy_tree)

     Decision Tree Accuracy: 0.9066666666666666

[91] classification_report_tree = classification_report(y_test, y_pred_tree)
     print("Decision Tree Classification Report:\n", classification_report_tree)

     Decision Tree Classification Report:
                   precision    recall  f1-score   support

                0       0.89      0.80      0.84        20
                1       0.93      0.98      0.95        41
                2       0.86      0.86      0.86        14

         accuracy                           0.91        75
        macro avg       0.89      0.88      0.88        75
     weighted avg       0.91      0.91      0.91        75


     confusion_matrix(y_test,y_pred_tree)

     array([[16,  2,  2],
            [ 1, 40,  0],
            [ 1,  1, 12]])
```

**Random forest:**

```
[110] print(f"Accuracy: {accuracy}")
```

```
Accuracy: 0.9066666666666666
```

```
[111] print("Confusion Matrix:")
      print(conf_matrix)
```

```
Confusion Matrix:
[[16  2  2]
 [ 1 40  0]
 [ 1  1 12]]
```

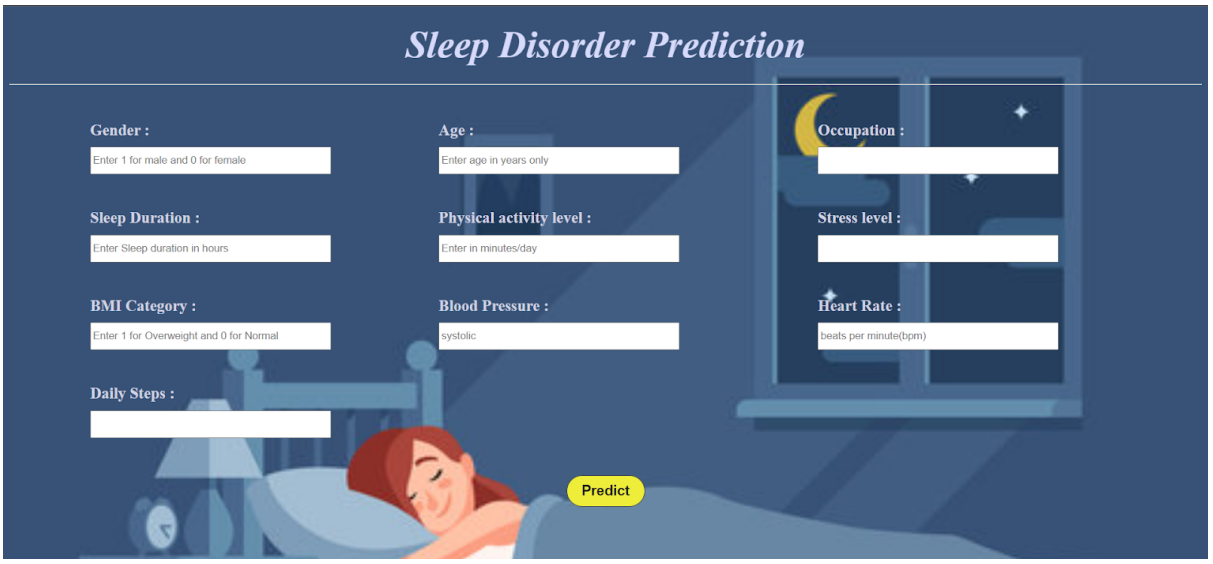
```
print("Classification Report:")
print(classification_rep)
```

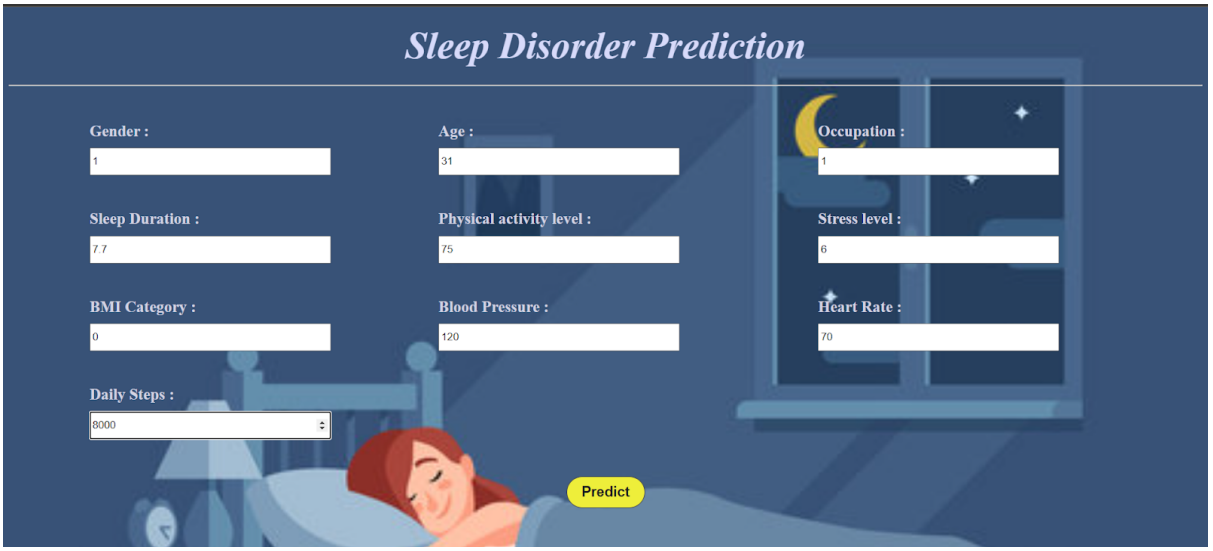
```
Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.80      0.84        20
           1       0.93      0.98      0.95        41
           2       0.86      0.86      0.86        14

    accuracy                           0.91        75
   macro avg       0.89      0.88      0.88        75
weighted avg       0.91      0.91      0.91        75
```

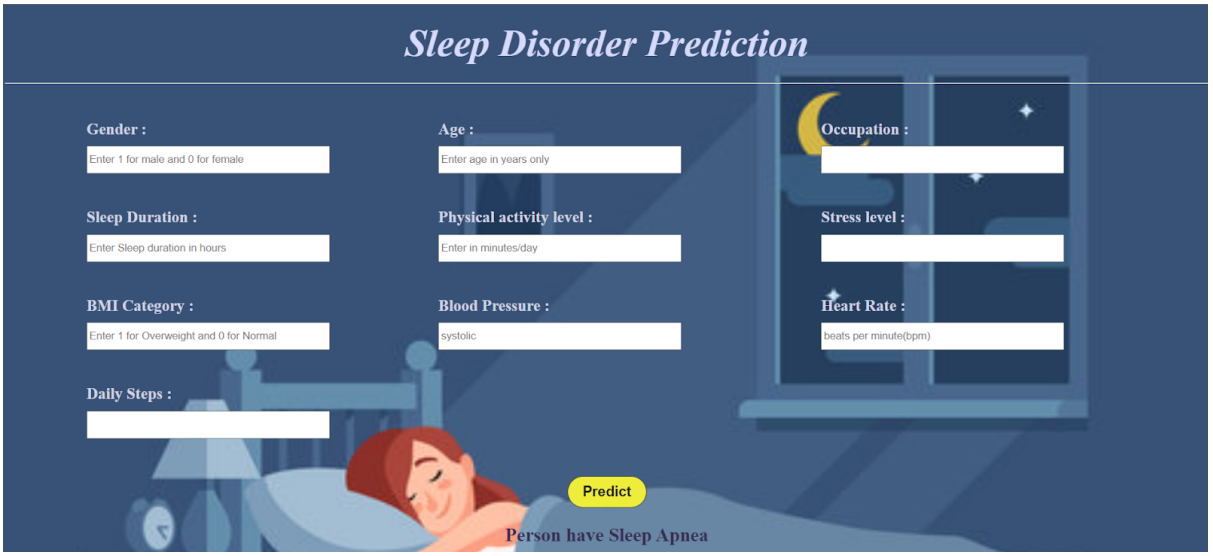
## Random Forest ( Hyper Parameter Tuning )

```
[109] print(f"Accuracy: {accuracy}")

      Accuracy: 0.9066666666666666

      print("Confusion Matrix:")
      print(conf_matrix)

      Confusion Matrix:
      [[16  2  2]
       [ 1 40  0]
       [ 1  1 12]]

[111] print("Classification Report:")
      print(classification_rep)

      Classification Report:
                    precision    recall  f1-score   support

                 0       0.89      0.80      0.84        20
                 1       0.93      0.98      0.95        41
                 2       0.86      0.86      0.86        14

          accuracy                           0.91        75
         macro avg       0.89      0.88      0.88        75
      weighted avg       0.91      0.91      0.91        75
```

## Website:






## 10. ADVANTAGES & DISADVANTAGES

Advantages highlight the system's accuracy in personalized insights, improved user well-being, and potential integration with wearable devices.

Disadvantages may include system dependencies and potential privacy concerns.

## 11. CONCLUSION

The conclusion summarizes achievements, emphasizing how the Sleep Oracle identifies the problem. It discusses user feedback and system improvements over the development lifecycle.

## 12. FUTURE SCOPE

The future scope envisions enhancements such as incorporating additional health metrics, expanding user engagement strategies, and potential collaborations with healthcare professionals.

## 13. APPENDIX

Source Code :

Sleep Oracle project-flask.ipynb

GitHub & Project Demo Link :

https://github.com/smartinternz02/SI-GuidedProject-611515-1698735855