HUMAN STRESS DETECTION AND PREDICTION USING ANN

By Amrutha TESR

INTRODUCTION

Stress during sleep significantly impacts health and well-being but often goes unnoticed. This project aims to develop a stress detection system that analyzes physiological data, such as heart rate variability, respiration rate, blood oxygen levels, and physical movements, to assess stress levels during sleep. An Artificial Neural Network (ANN) classifies sleep data into "Stressed" or "Not Stressed" categories.

The system features an interactive dashboard displaying real-time data, historical trends, and stress patterns, providing users with actionable insights. Personalized recommendations, such as breathing exercises or meditation, help users manage stress and improve sleep quality.

To protect sensitive physiological data, the platform incorporates robust encryption and complies with health standards. By leveraging advanced machine learning and user-friendly design, this project seeks to enhance individuals' ability to monitor and manage stress during sleep, promoting better overall health.

PROJECT OVERVIEW

- User-Friendly Dashboard: Provides real-time data, stress level scores, and historical trends through charts and graphs.
- Data Collection Options: Supports multiple data sources, including manual input, wearable device sync, and webcam-based analysis.
- Real-Time Stress Monitoring: Continuously monitors physiological and behavioral data to detect stress levels instantly.
- Personalized Stress Insights: Offers detailed analysis of stress triggers, patterns, and trends based on individual data.
- Machine Learning-Based Detection: Uses predictive algorithms to analyze data and classify stress levels as low, moderate, or high.
- Data Privacy and Security: Ensures secure data storage and encryption to protect user information.

TECH STACK

Frontend:

1. **HTML5**: This is used to create the structure of web pages.
2. **CSS3**: For styling the web pages.
3. **JavaScript**: This is for adding interactivity (if applicable, such as form validations).
4. **Django Templates**: To dynamically render data from the backend into the web pages.

Backend:

1. **Django (Python)**: For handling server-side logic, user input, and rendering templates.
2. **Django Forms:** To manage user input via forms in the frontend.
3. **MySQL**: This is for storing user data (if data is being saved).

TECH STACK

Machine Learning & Deep Learning:

1. Python: Programming language for ML model development.
2. Pandas: For data manipulation and analysis.
3. NumPy: For numerical computations.
4. Scikit-Learn: For data preprocessing (e.g., StandardScaler) and splitting the dataset.
5. TensorFlow/Keras: For building, training, and deploying the Artificial Neural Network (ANN).
6. Joblib: For saving and loading the scaler model.
7. Matplotlib & Seaborn: For visualizing training accuracy, loss curves, and data distribution.

DATA PARAMETERS

1)Snoring Rate

2)Respiratory Rate

3)Body Temperature

4)Limb Movement

5)Blood Oxygen

6)Eye Movement

7)Sleep Hours

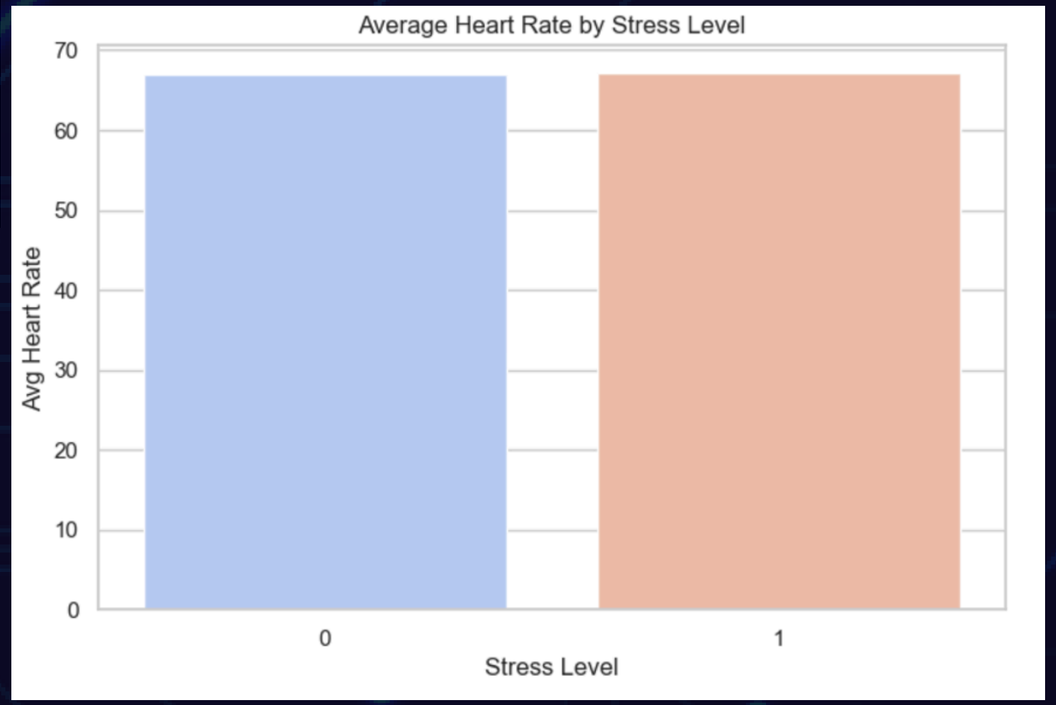
8)Heart Rate

9)Stress Level

DATA VISUALIZATION
PLOTS

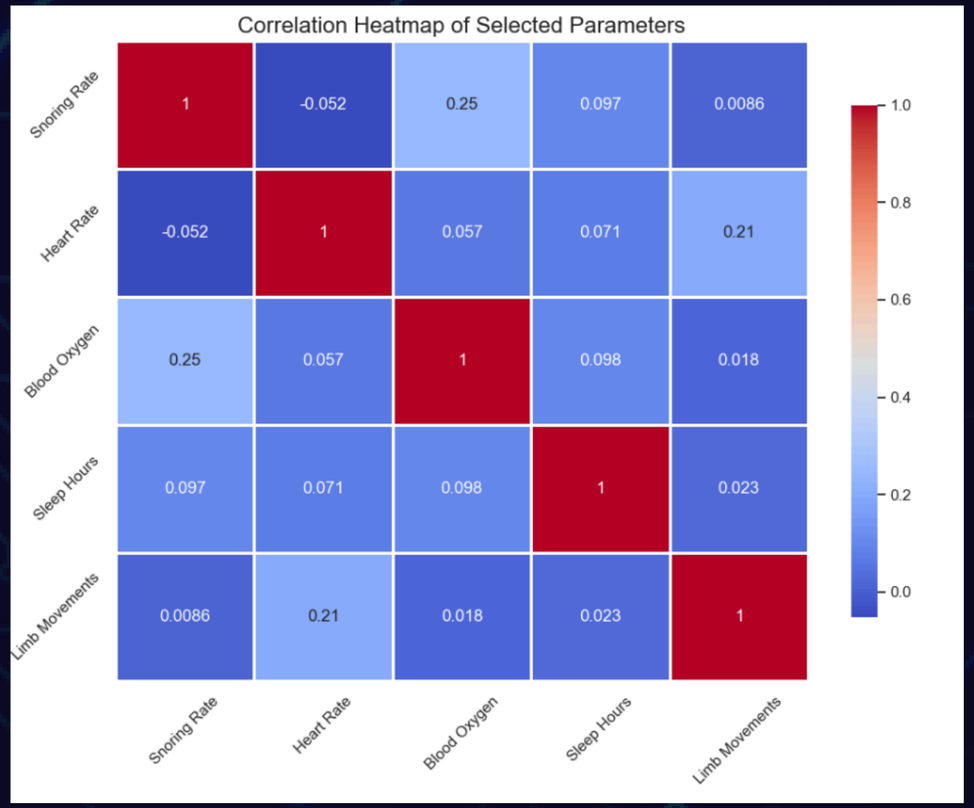
BARPLOT

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('generated_data1.csv')

sns.set(style="whitegrid")
average_heart_rate = df.groupby('Stress Level')['Heart Rate'].mean().reset_index()
plt.figure(figsize=(8, 5))
sns.barplot(data=average_heart_rate, x='Stress Level', y='Heart Rate', palette='coolwarm')
plt.title('Average Heart Rate by Stress Level')
plt.xlabel('Stress Level')
plt.ylabel('Avg Heart Rate')
plt.show()
```



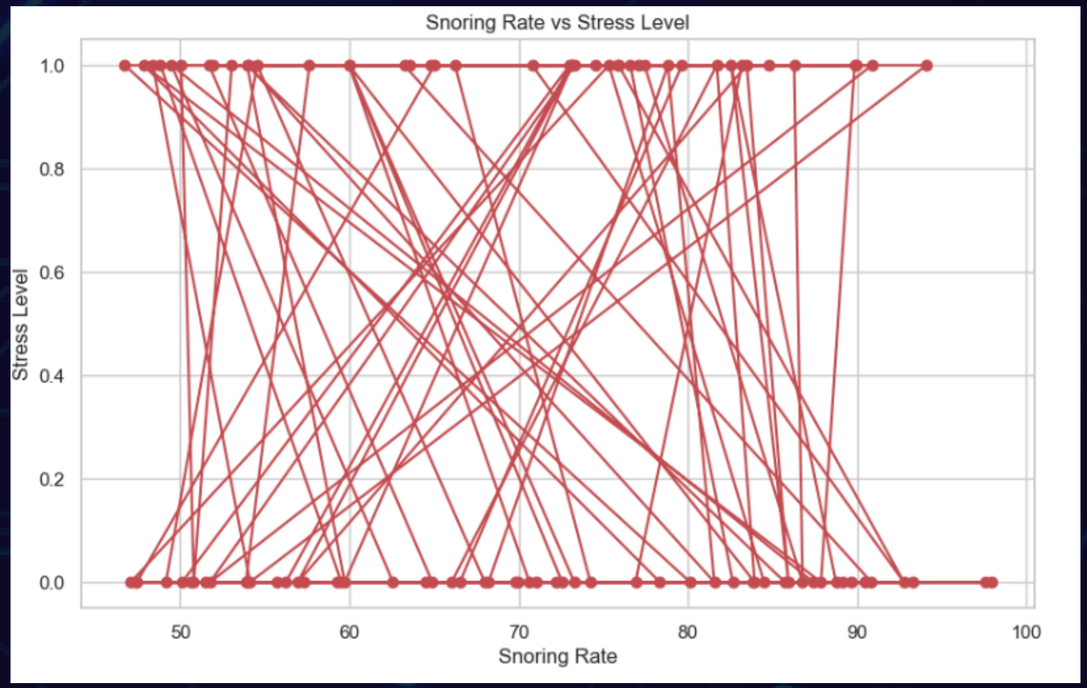
HEAT MAP

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('generated_data1.csv')
selected_columns = ['Snoring Rate', 'Heart Rate', 'Blood Oxygen', 'Sleep Hours', 'Limb Movements']

correlation = df[selected_columns].corr()

plt.figure(figsize=(10, 8))
sns.heatmap(correlation, annot=True, cmap='coolwarm', linewidths=1, linecolor='white', cbar_kws={"shrink": 0.8})

plt.title('Correlation Heatmap of Selected Parameters', fontsize=16)
plt.xticks(rotation=45)
plt.yticks(rotation=45)
plt.tight_layout()
plt.show()
```



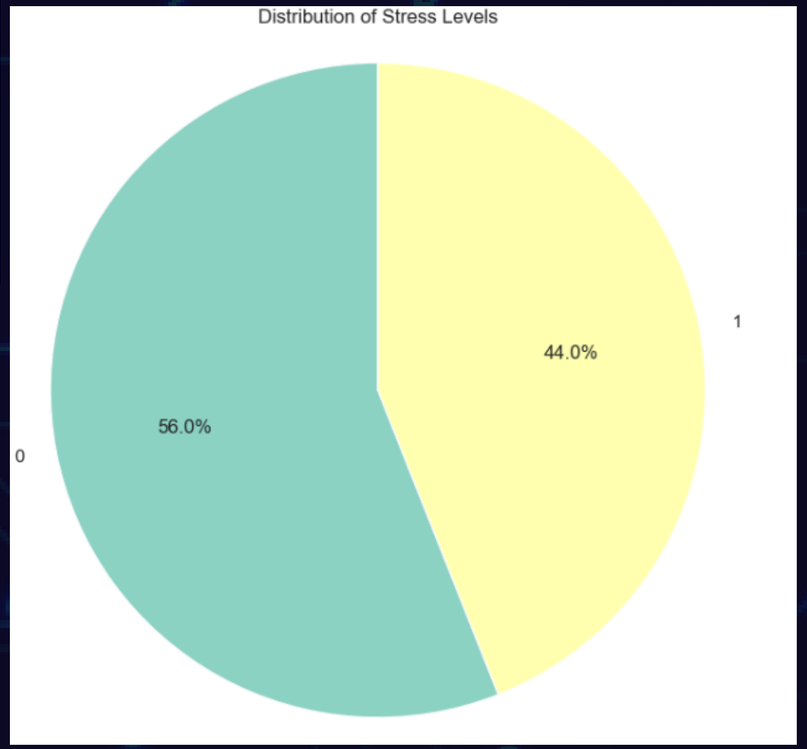
LINE PLOT

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('generated_data1.csv')

sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
plt.plot(df['Snoring Rate'], df['Stress Level'], marker='o', linestyle='-', color='r')

plt.title('Snoring Rate vs Stress Level')
plt.xlabel('Snoring Rate')
plt.ylabel('Stress Level')

plt.grid(True)
plt.show()
```



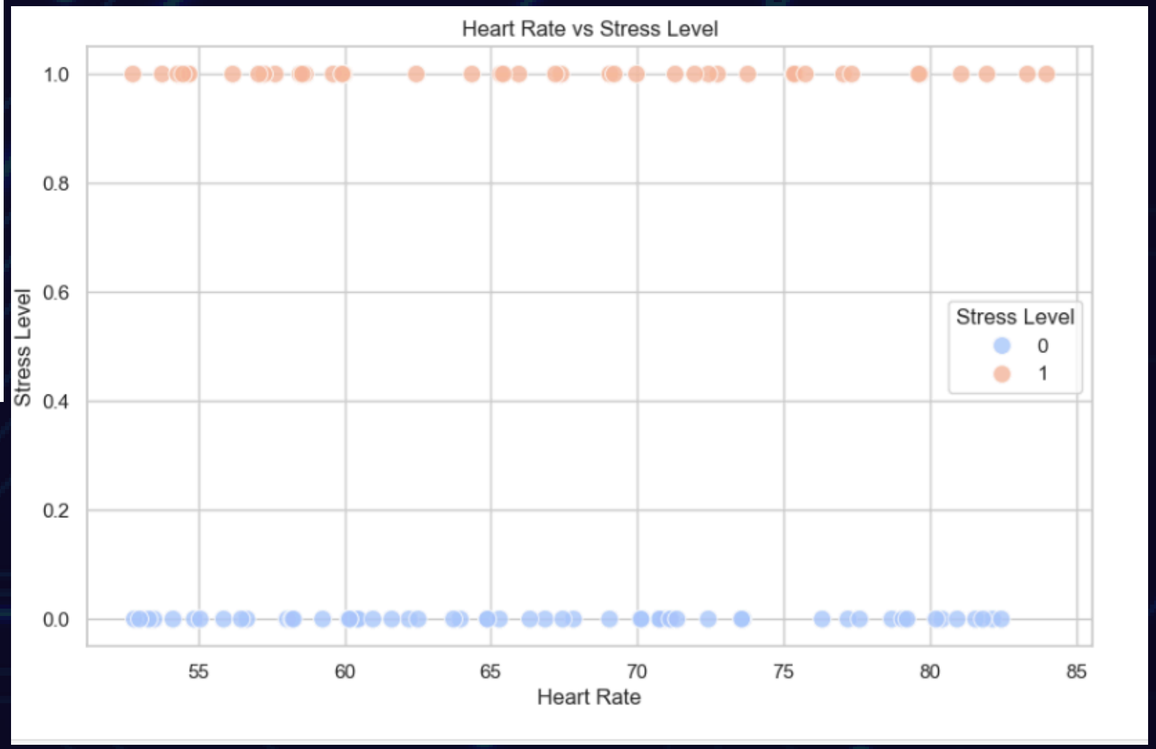
PIE CHART

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('generated_data1.csv')

stress = df['Stress Level'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(stress, labels=stress.index, autopct='%1.1f%%', startangle=90, colors=sns.color_palette('Set3'))

plt.title('Distribution of Stress Levels')
plt.axis('equal')
plt.show()
```



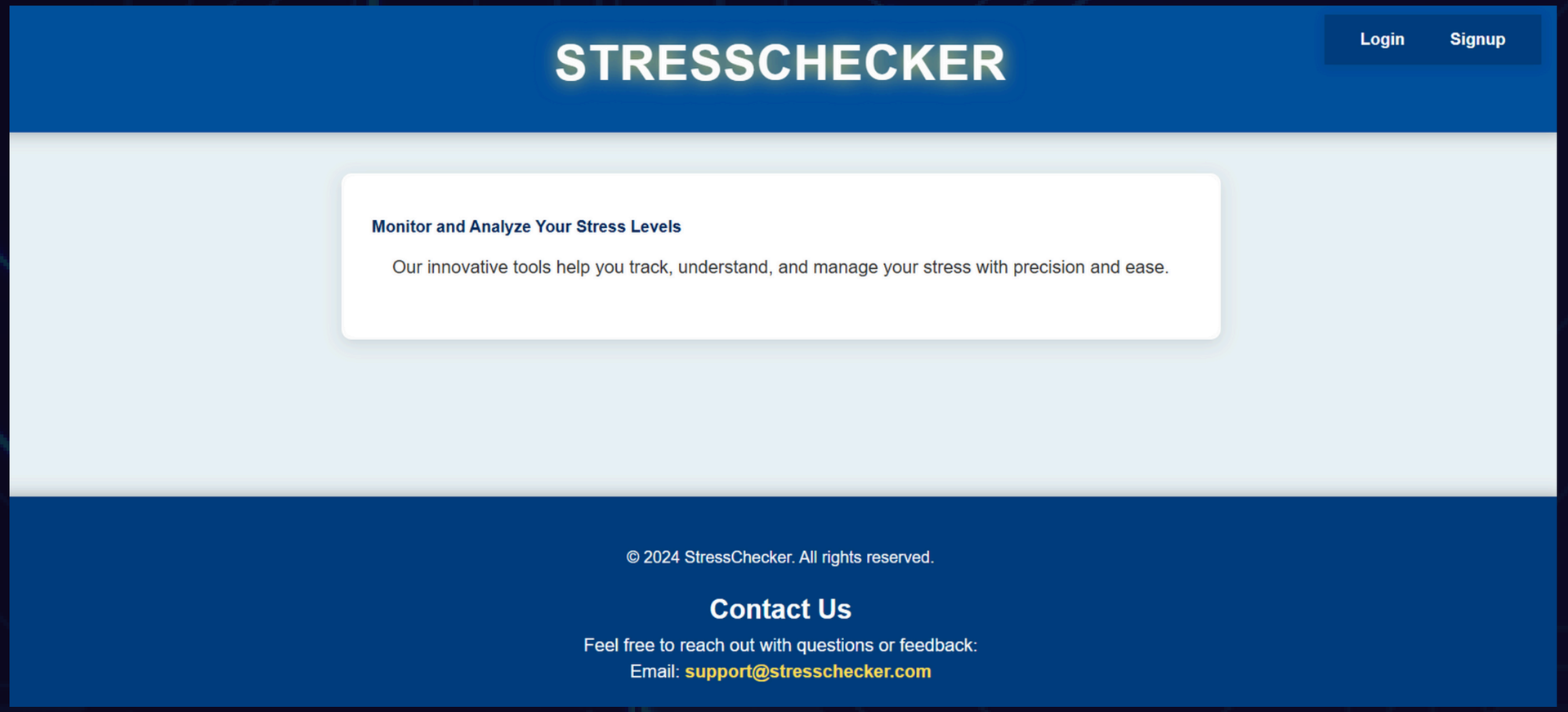
SCATTER PLOT

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('generated_data1.csv')
sns.set(style="whitegrid")

plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Heart Rate', y='Stress Level',
                hue='Stress Level', palette='coolwarm', s=100, alpha=0.8)
plt.title('Heart Rate vs Stress Level')
plt.xlabel('Heart Rate')
plt.ylabel('Stress Level')

plt.legend(title='Stress Level')
plt.show()
```

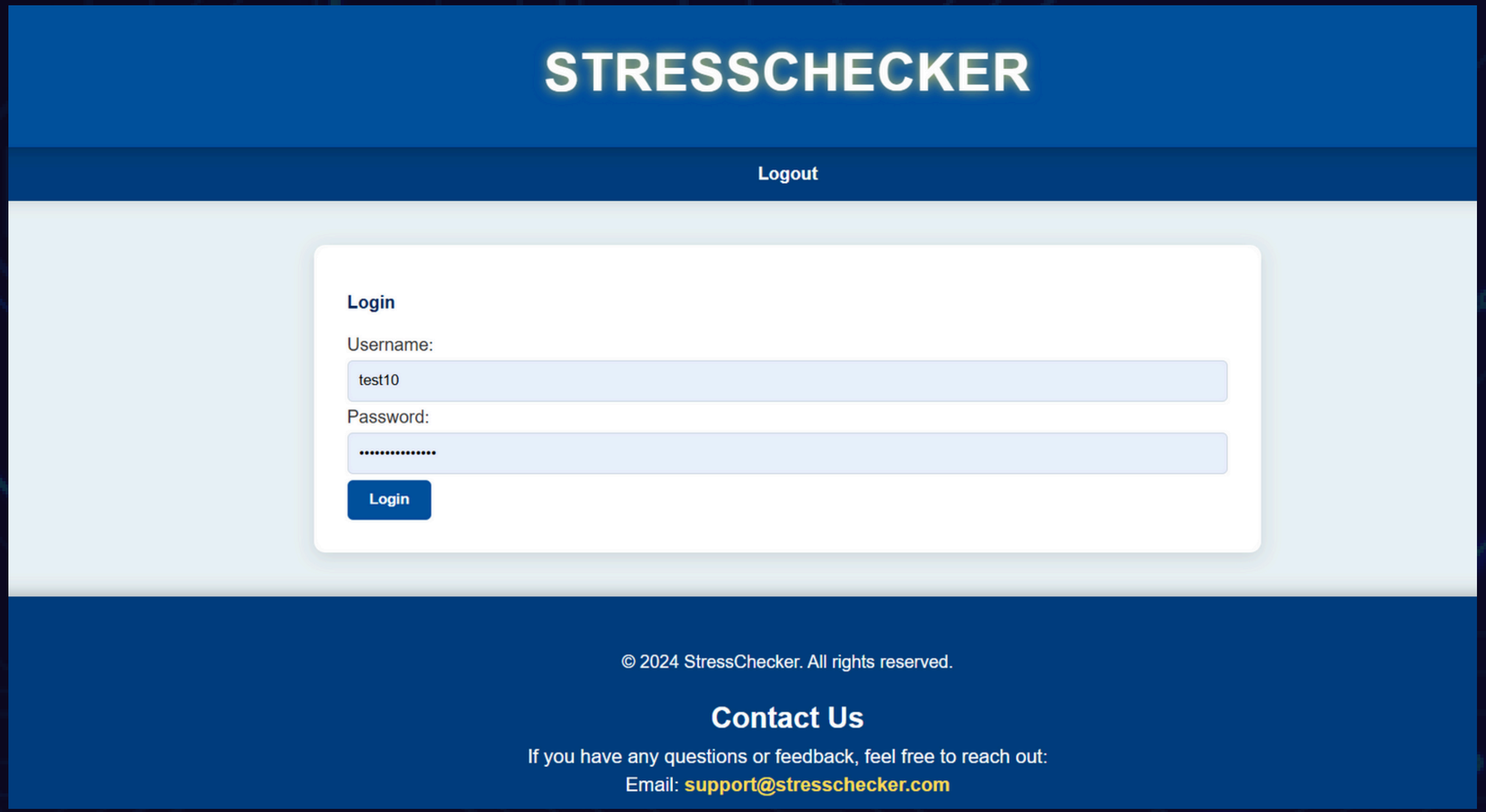


HUMAN STRESS PREDICTION WEBISTE

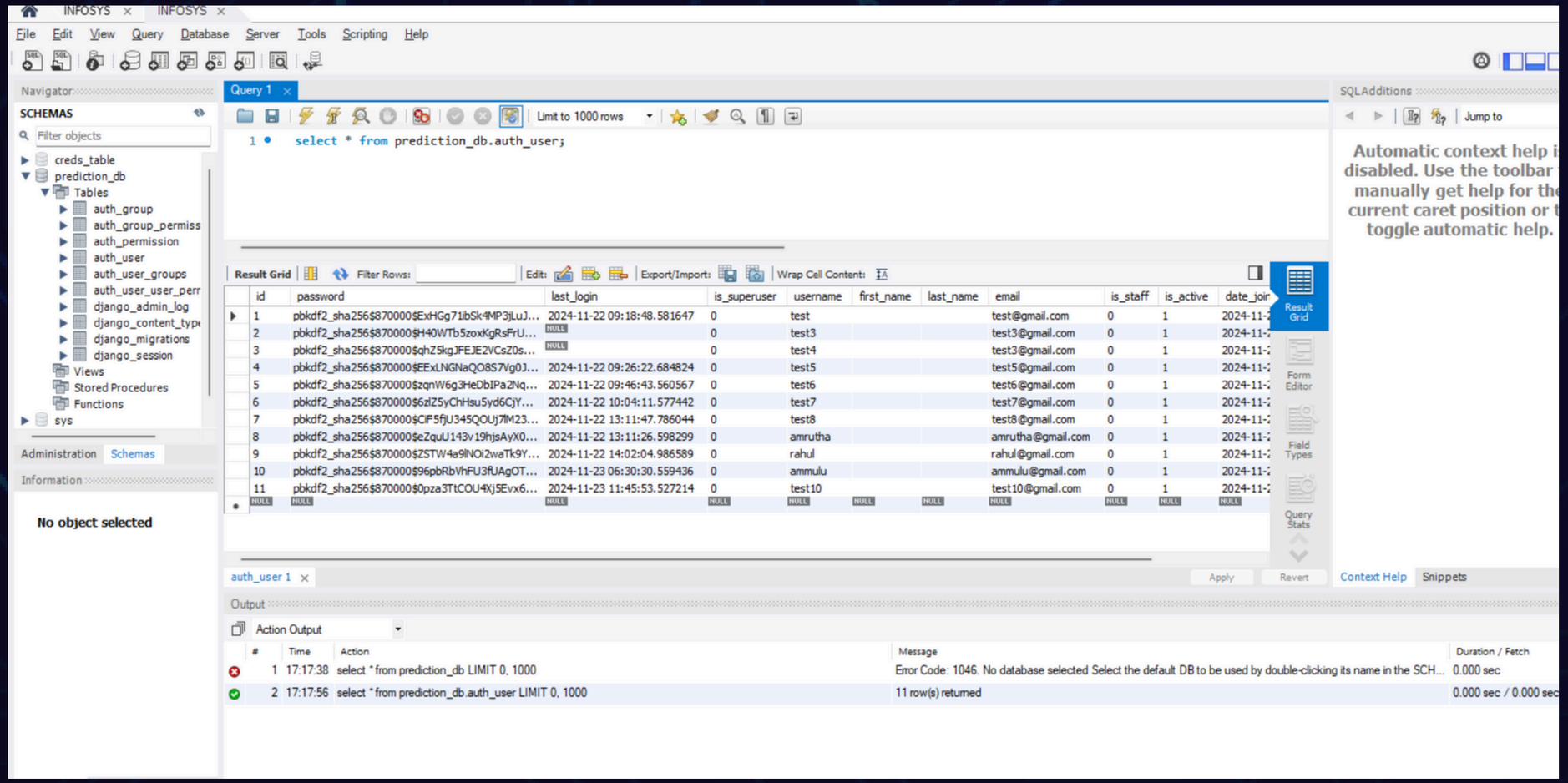
MAIN PAGE


STRESSCHECKER
Login
Signup
Monitor and Analyze Your Stress Levels
Our innovative tools help you track, understand, and manage your stress with precision and ease.
© 2024 StressChecker. All rights reserved.
Contact Us
Feel free to reach out with questions or feedback:
Email: support@stresschecker.com

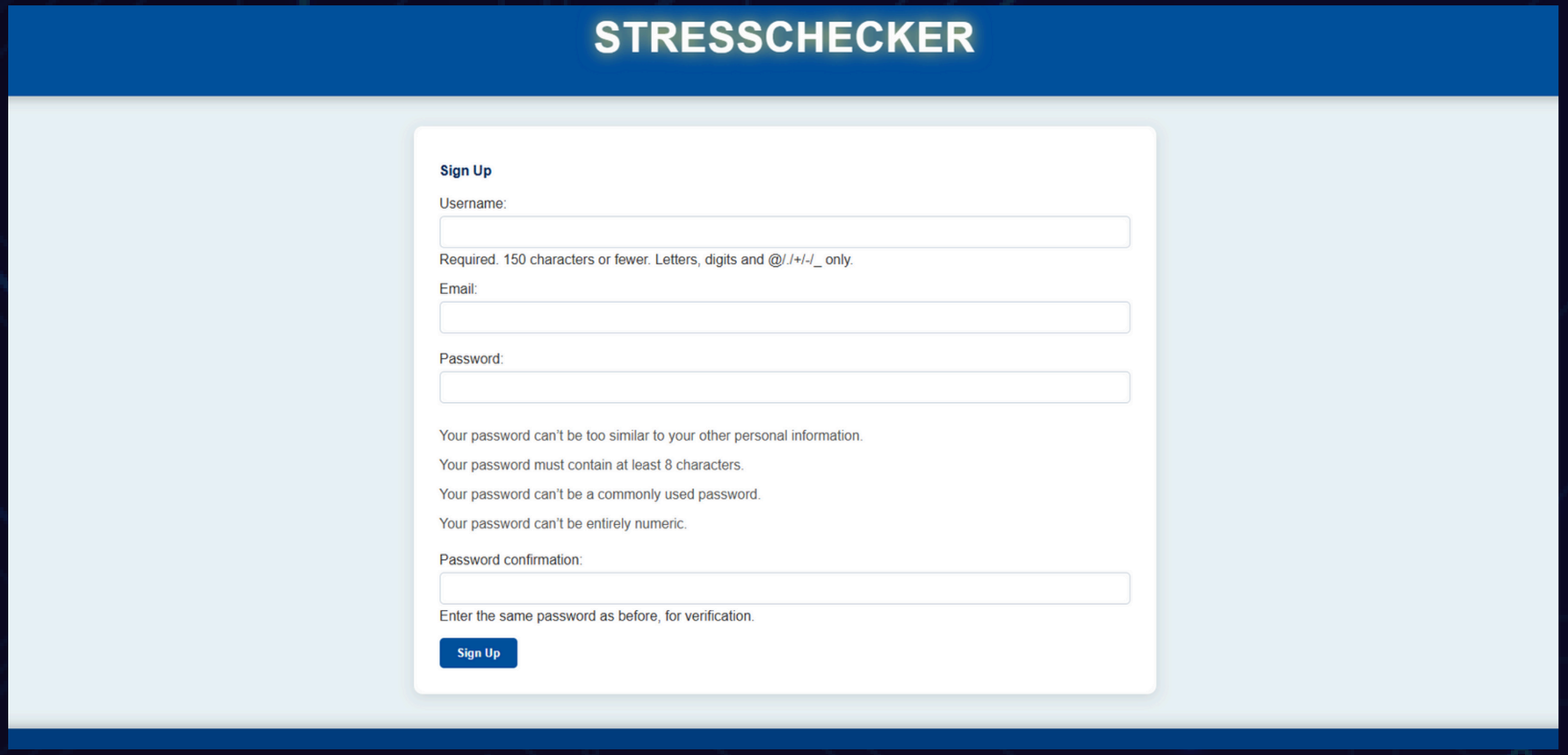
LOGIN PAGE


STRESSCHECKER
Logout
Login
Username:
test10
Password:
Login
© 2024 StressChecker. All rights reserved.
Contact Us
If you have any questions or feedback, feel free to reach out:
Email: support@stresschecker.com

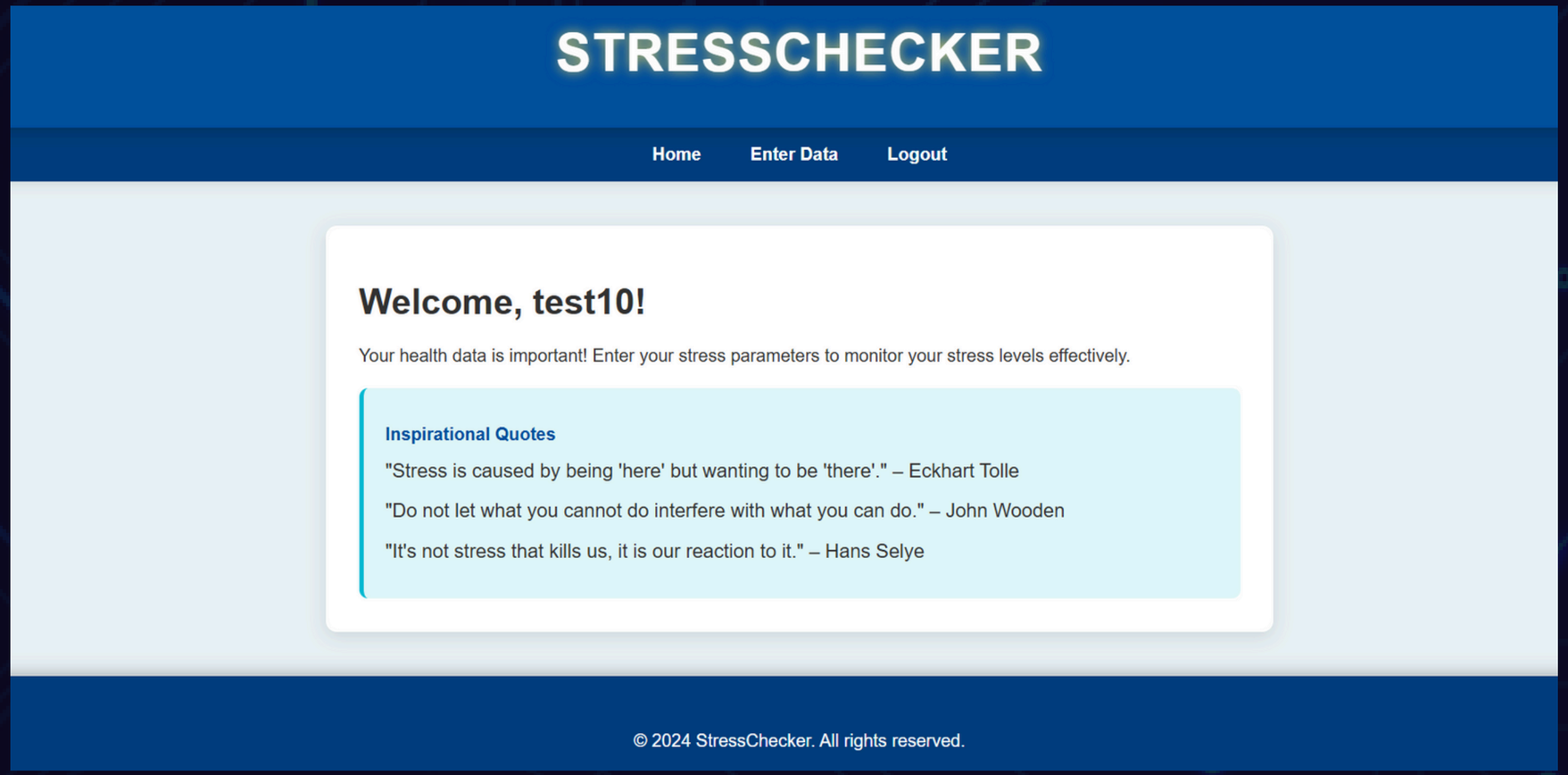
DATABASE CONNECTION



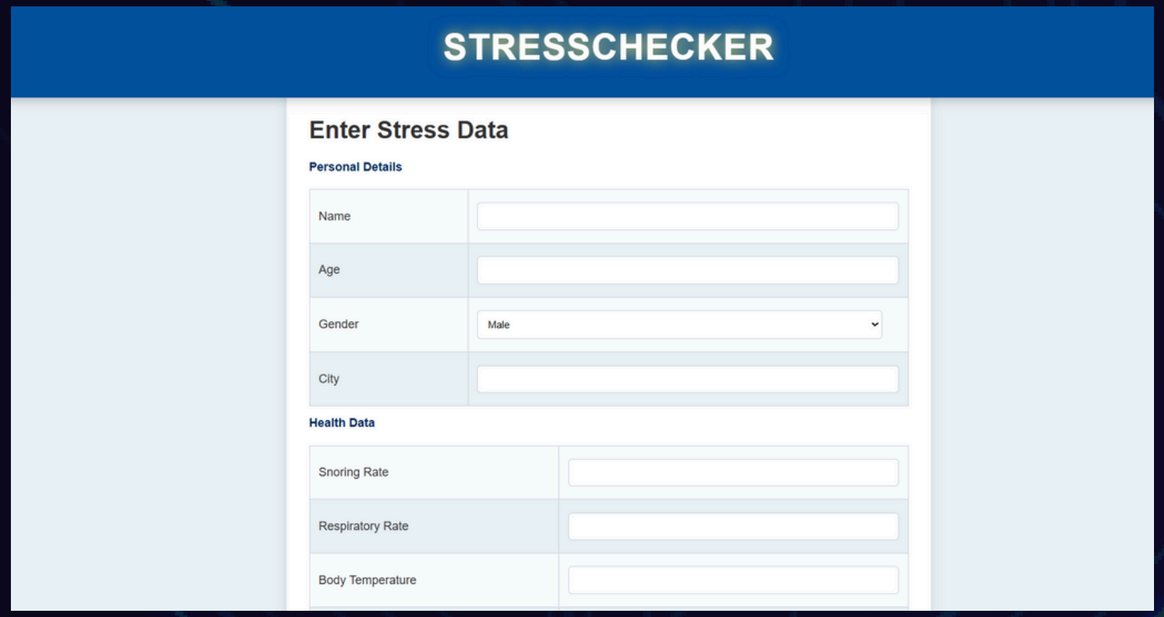
SIGNUP PAGE

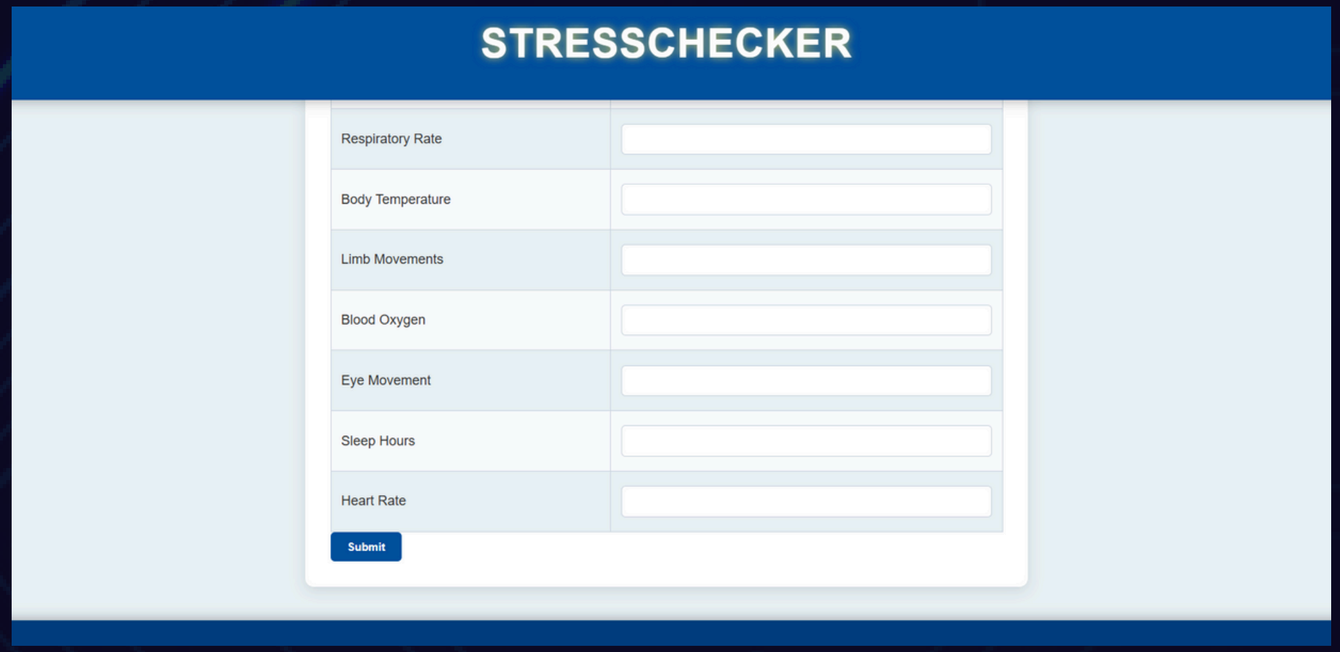

STRESSCHECKER
Sign Up
Username:
Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.
Email:
Password:
Your password can't be too similar to your other personal information.
Your password must contain at least 8 characters.
Your password can't be a commonly used password.
Your password can't be entirely numeric.
Password confirmation:
Enter the same password as before, for verification.
Sign Up

HOME PAGE

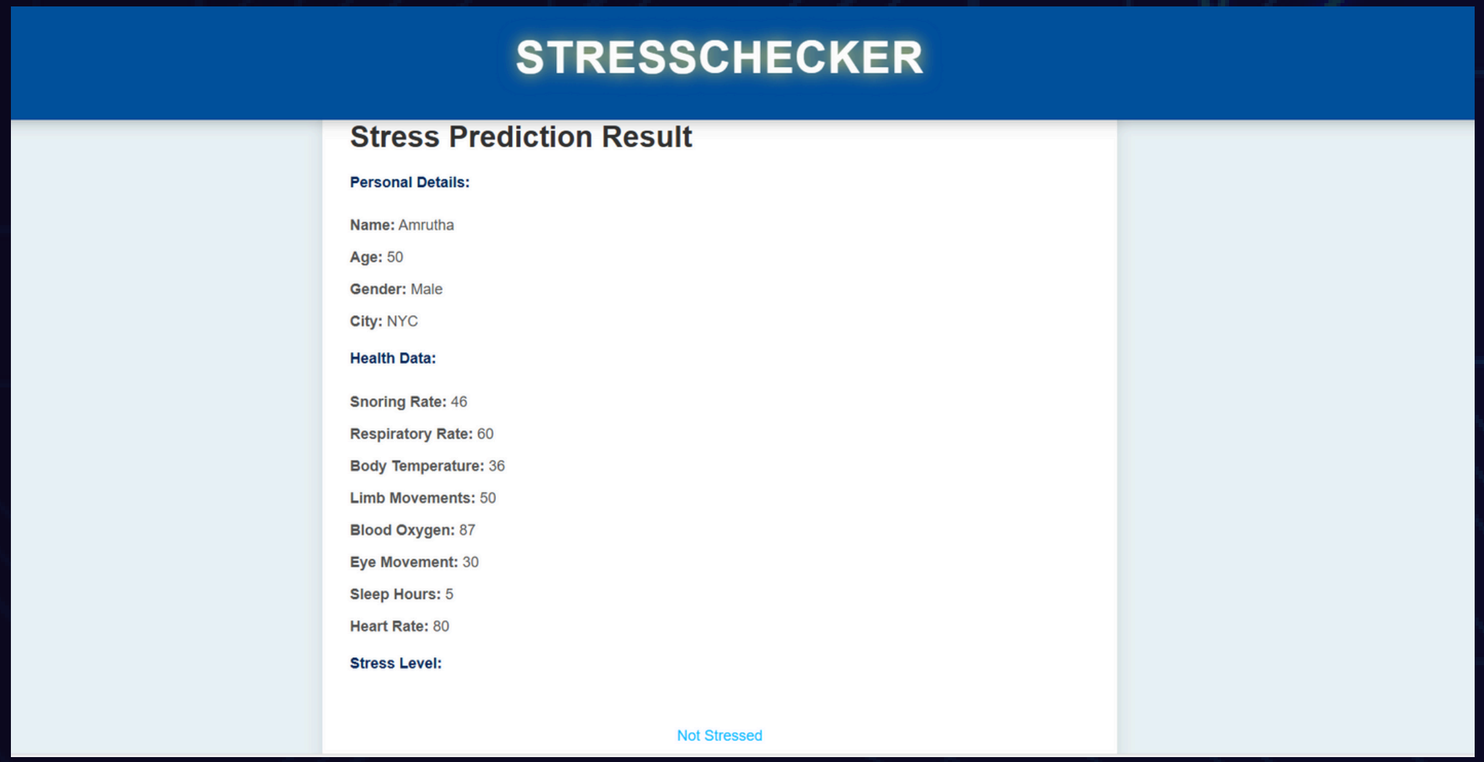

STRESSCHECKER
Home
Enter Data
Logout
Welcome, test10!
Your health data is important! Enter your stress parameters to monitor your stress levels effectively.
Inspirational Quotes
"Stress is caused by being 'here' but wanting to be 'there'." – Eckhart Tolle
"Do not let what you cannot do interfere with what you can do." – John Wooden
"It's not stress that kills us, it is our reaction to it." – Hans Selye
© 2024 StressChecker. All rights reserved.

DATA ENTRY PAGE





RESULT PAGE


STRESSCHECKER
Stress Prediction Result
Personal Details:
Name: Amrutha
Age: 50
Gender: Male
City: NYC
Health Data:
Snoring Rate: 46
Respiratory Rate: 60
Body Temperature: 36
Limb Movements: 50
Blood Oxygen: 87
Eye Movement: 30
Sleep Hours: 5
Heart Rate: 80
Stress Level:
Not Stressed

CONCLUSION

A human stress detection system empowers users to monitor and manage stress levels during sleep by leveraging physiological data such as heart rate, respiratory rate, and body movements. Using binary classification (1 = Stressed, 0 = Not Stressed), the system predicts stress levels with the help of an Artificial Neural Network (ANN). Real-time monitoring and performance metrics, including accuracy, precision, recall, and F1-score, ensure reliable analysis.

The platform features visualizations of data distributions, training progress, and model performance, enhancing user understanding of the results. Key modules implemented include data preprocessing and visualization, model architecture design, training and evaluation with visual feedback, and model deployment for seamless integration.

Through a user-friendly dashboard and personalized recommendations, the system aids users in recognizing stress patterns and improving mental well-being. With robust privacy measures, it is a secure and effective tool for achieving better stress management and overall health.

THANK YOU