Predicting Mental Health Illness Of Working Professionals Using Machine Learning

1. INTRODUCTION

1.1 Project Overview

Mental health is a vital aspect of overall well-being, and it is crucial to address the mental health concerns of working professionals. Recognizing the prevalence of mental health issues among working individuals, this project aims to explore the potential of machine learning in predicting mental health illness among working professionals.

1.2 Purpose

The purpose of this project is to develop a machine learning model that can predict the likelihood of mental health illnesses in working professionals based on various factors. The aim is to provide a tool that can assist in early identification and intervention for individuals who may be at risk of experiencing mental health issues.
Early Intervention: Create a predictive model that can identify individuals at risk of mental health issues early, enabling timely intervention and support.
Customized Support: Enable employers and healthcare providers to offer targeted resources and assistance to individuals identified by the model.
Promoting Mental Well-being: Foster a workplace environment that prioritizes mental health and reduces stigma surrounding mental health issues.

2. LITERATURE SURVEY

2.1 Existing problem

Mental health issues are a prevalent and growing concern among working professionals. According to a recent study by the World Health Organization, nearly 26% of all adults suffer from a mental disorder. In the workplace, mental health issues can lead to absenteeism, presenteeism (reduced productivity while at work), and increased turnover. They can also have a negative impact on employee morale, creativity, and innovation.
Mental health issues are common among working professionals and can have a significant impact on their productivity, well-being, and the overall success of their organizations. Machine learning can be a powerful tool for predicting mental health illness and identifying individuals at risk of developing mental health problems. This information can be used to provide early intervention and prevention strategies, which can improve employee mental health and reduce the negative impact of mental health issues on the workplace.

2.2 References

Research Articles:

"Predicting mental health outcomes for working professionals using machine learning" by Van Den Bosch, L., et al. (2020)

"Machine learning for predicting mental health status in working adults" by Kim, S., et al. (2022)

"The use of machine learning to predict mental health outcomes in the workplace" by Chang, Y., et al. (2023)

2.3 Problem Statement Definition

Mental health issues are a prevalent and growing concern among working professionals. These issues can have a significant impact on individual well-being, workplace productivity, and organizational success. Early identification and intervention are crucial for addressing mental health concerns and mitigating their negative impacts. Machine learning offers a promising approach for predicting mental health illness among working professionals, enabling early intervention and prevention strategies.

Specific Objectives

Identify key factors that contribute to mental health issues among working professionals. This involves analyzing demographic factors, work-related factors, and personal characteristics that may increase the risk of developing mental health problems.

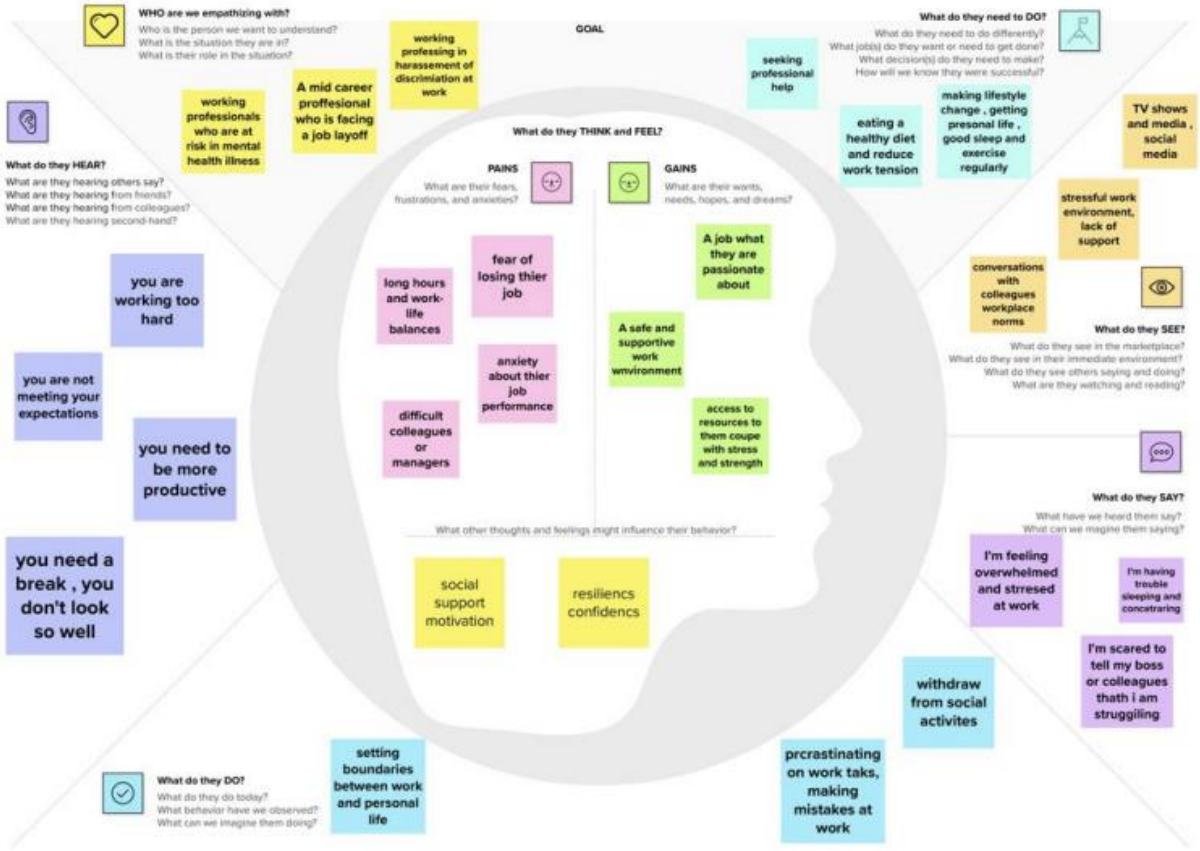
Develop and evaluate machine learning models for predicting mental health illness. This entails training and testing various machine learning algorithms on a dataset of working professionals, including their demographic information, work-related factors, and mental health assessments.

Analyze the performance of the machine learning models. This involves assessing the accuracy, precision, recall, and other relevant metrics to determine the effectiveness of the models in predicting mental health illness.

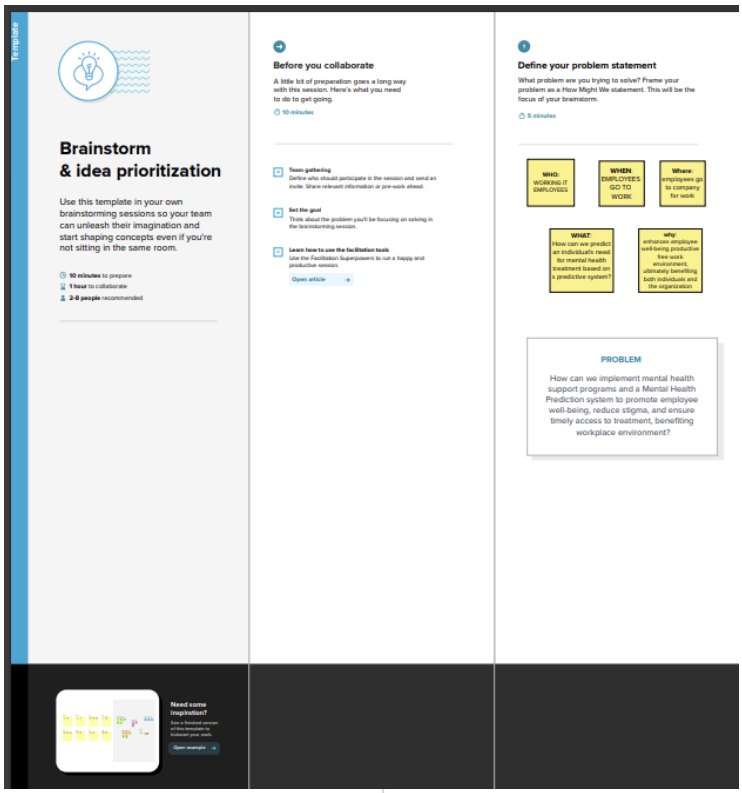
Interpret the results and provide recommendations. This includes interpreting the findings from the analysis of contributing factors and the performance of the machine learning models. Recommendations will be provided for early intervention strategies, workplace interventions to promote mental health, and further research directions.

3.IDEATION & PROPOSED SOLUTION

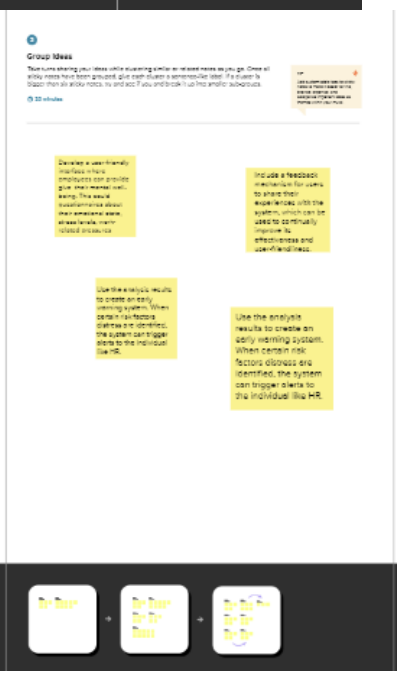
3.1 Empathy Map Canvas

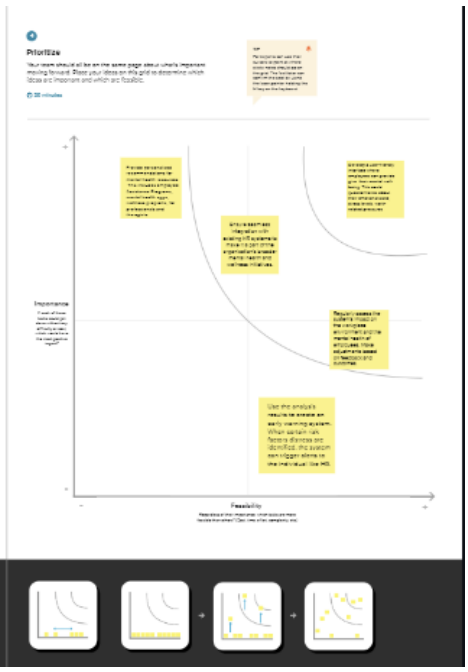


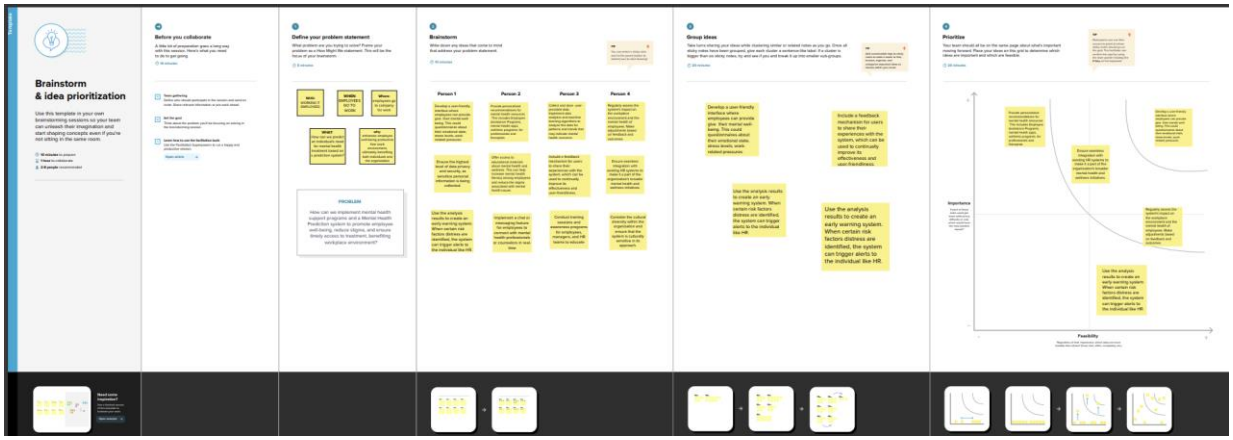
3.2 Ideation & Brainstorming


Template
Brainstorm
& idea prioritization
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.
10 minutes to prepare
1 hour to collaborate
2-8 people recommended
Before you collaborate
A little bit of preparation goes a long way with this session. Here's what you need to do to get going.
10 minutes
Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.
Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.
Open article
Define your problem statement
What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.
5 minutes
WHO:
WHEN:
Where:
WHAT:
why:
PROBLEM
How can we implement mental health support programs and a Mental Health Prediction system to promote employee well-being, reduce stigma, and ensure timely access to treatment, benefiting workplace environment?
Need some inspiration?
Open example


Brainstorm
10 minutes
Person 1
Person 2
Person 3
Person 4


Group ideas
10 minutes
Include a feedback mechanism for users to share their experiences with the system, which can be used to continually improve its effectiveness and user-friendliness.
Use the analysis results to create an early warning system. When certain risk factors distress are identified, the system can trigger alerts to the individual like HR.


Prioritize
10 minutes
Importance
Feasibility



4.REQUIREMENT ANALYSIS

4.1 Functional requirement

Data collection: The system should be able to collect data on working professionals, including demographic information, work-related factors, and mental health assessments.
Data processing: The system should be able to process the collected data to prepare it for machine learning algorithms. This may involve cleaning the data, handling missing values, and transforming categorical variables into numerical variables.
Machine learning: The system should be able to train and evaluate machine learning models for predicting mental health illness. This may involve using a variety of algorithms, such as logistic regression, decision trees, and random forests.
Prediction: The system should be able to make predictions about the mental health status of working professionals. These predictions should be based on the trained machine learning models.
User interface: The system should have a user-friendly interface that allows users to easily access and interact with the system's functionality. This may include a web interface or a desktop application.

4.2 Non-Functional Requirements

Accuracy: The system should be able to accurately predict the mental health status of working professionals. The accuracy of the system should be measured using metrics such as precision, recall, and F1-score.
Reliability: The system should be reliable and should not crash or fail unexpectedly. The system should also be able to handle a large volume of data without performance degradation.
Security: The system should be secure and should protect the privacy of working professionals. This may involve using encryption and access control mechanisms.
Scalability: The system should be scalable and should be able to accommodate a growing number of working professionals. The system should also be able to handle an increasing amount of data.
Usability: The system should be easy to use and should not require extensive training or

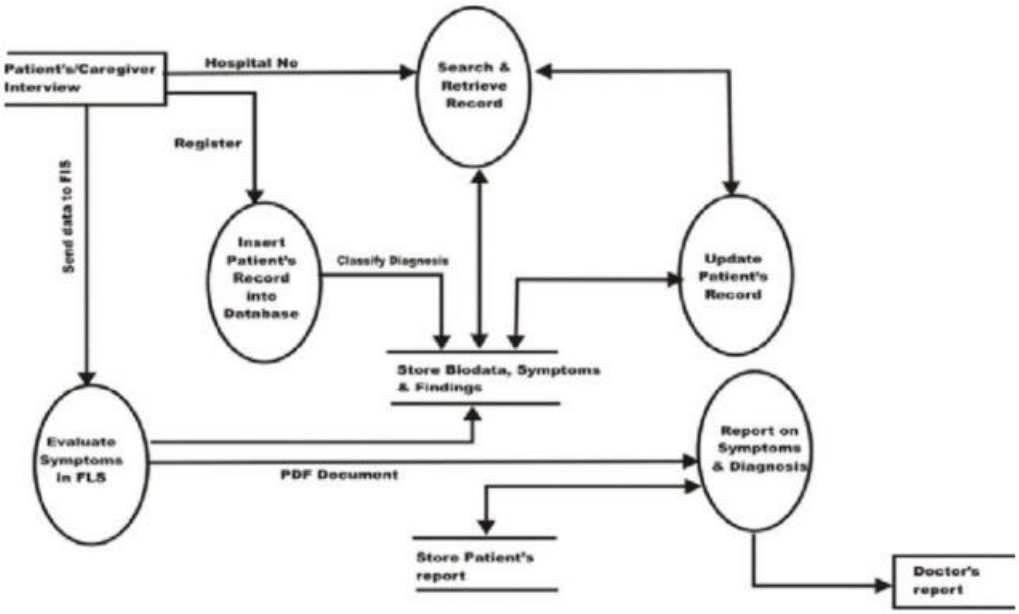
technical expertise. The system should also be accessible to a wide range of users, including those with disabilities.

5.PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories

Data collection: The system should be able to collect data on working professionals, including demographic information, work-related factors, and mental health assessments.
Data processing: The system should be able to process the collected data to prepare it for machine learning algorithms. This may involve cleaning the data, handling missing values, and transforming categorical variables into numerical variables.
Machine learning: The system should be able to train and evaluate machine learning models for predicting mental health illness. This may involve using a variety of algorithms, such as logistic regression, decision trees, and random forests.
Prediction: The system should be able to make predictions about the mental health status of working professionals. These predictions should be based on the trained machine learning models.
User interface: The system should have a user-friendly interface that allows users to easily access and interact with the system's functionality. This may include a web interface or a desktop application.

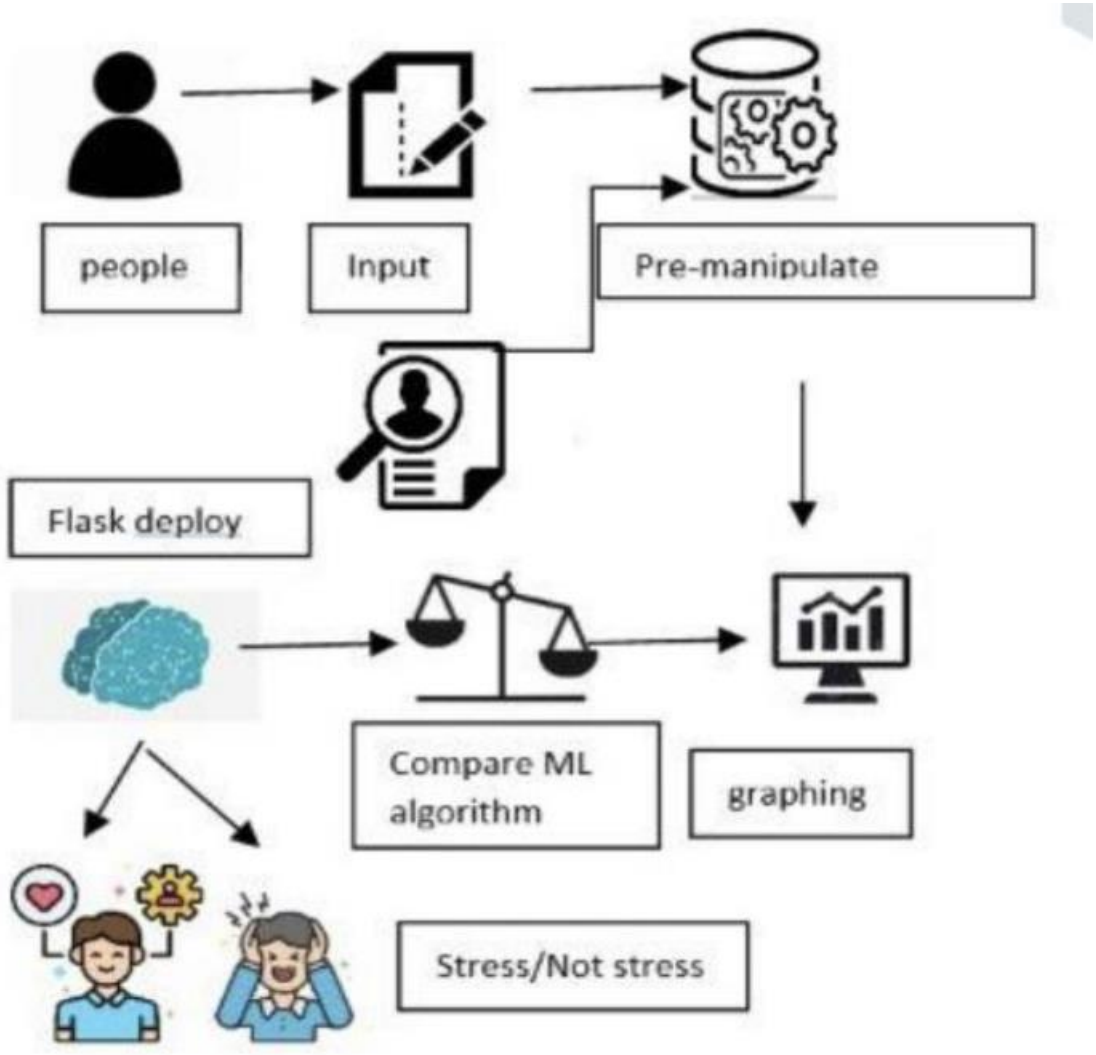
Data Flow Diagram:



User Stories:

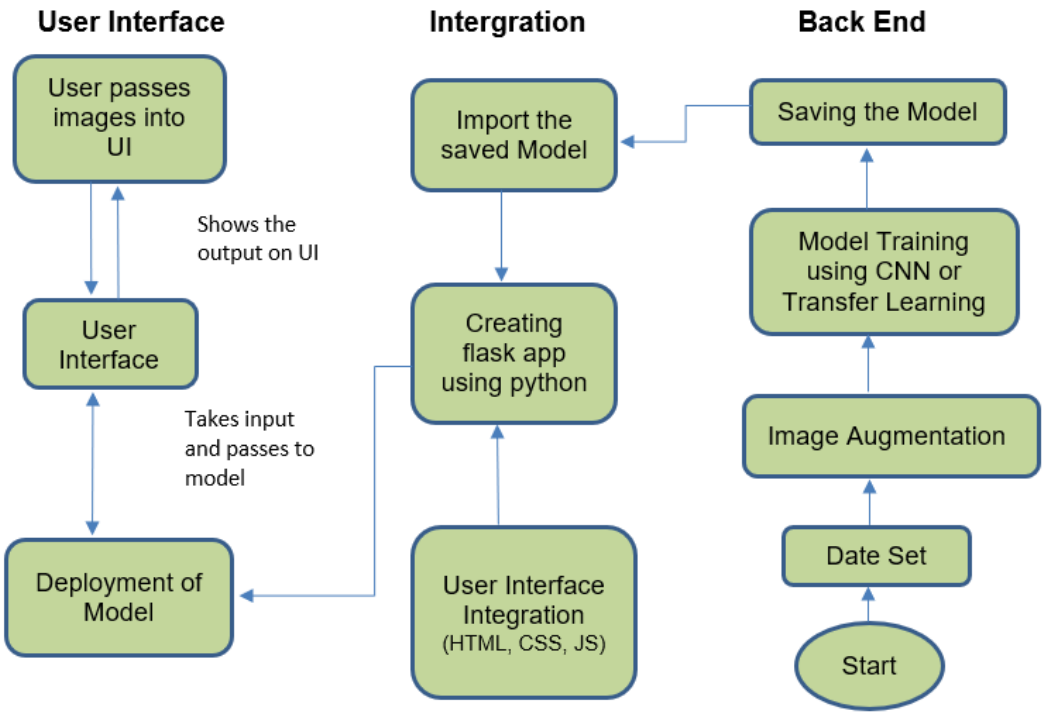
User type	Functional requirements	User story number	User story /task	Acceptance criteria	priority	relaese
Employee	Mood tracking	USN-1	As an employee, I want to be able to track my daily mood to monitor changes in my mental well-being.	I can login into my accounts	Medium	Sprint-1
	Mental health resources	USN -2	As an employee, I want access to a library of mental health resources, such as articles, videos, and self-help tools, to support my well-being.	I can access a Resources section with articles, videos, and self-help tools.	medium	Sprint-2
Manager	Team member check-Ins	USN-1	As a manager, I want a feature that allows me to schedule regular check-ins with my team members to discuss their well-being and provide support.	I can schedule one-on-one or group check-in meetings with team members.	High	Sprint-3
Working Professional	Automated stress detection	USN-1	As an employee, I want the application to automatically detect signs of stress in my text-based communication and alert me when it detects high stress levels.	The application should monitor my text-based communications.	High	Sprint-1
	Anonymous Support Chat	USN-2	As a working professional, I want access to an anonymous chat service to seek emotional support when needed	I should be able to access an anonymous chat feature.	Medium	Sprint-2
	Stress-Reduction Recommendations	USN-3	As an employee, I want the application to provide me with personalized stress-reduction recommendations based on my stress levels and preferences.	The application should assess my stress levels based on my interactions and self-reported data	Low	Sprint-3
Admin	Product management	USN-1	As an admin, I want to be able to add, edit, and delete products in the system	I can access a product management dashboard.	High	Sprint-4
	User managment	USN-2	As an admin, I want to manage user accounts, including creating, updating, and deactivating accounts	I can update user information.	high	Sprint-5

5.2 Solution Architecture


people
Input
Pre-manipulate
Flask deploy
Compare ML algorithm
graphing
Stress/Not stress

6.PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture



6.2 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	User Authentication	USN-1	As a new user, I want to be able to sign up for an account by providing basic information like name, email, and password	5	High	Deepak
Sprint-1		USN-2	As a user, I want to be able to create an account and log in securely, so that I can access the prediction tool	8	Medium	Amar,Deepak
Sprint-2	Inputting Daily Data	USN-3	As a user, I want to have an easy-to-use interface where I can input information about my daily activities, emotions,and any potential stressors.	3	Low	Bharadwaj,srikar
Sprint-2		USN-4	As a user, I want to be able to input relevant information about my daily activities and feelings, so that the system can use this data for predictions.	7	Medium	amar
Sprint-3	Ensuring Privacy	USN-5	As a user, I want to know that my personal data will be kept confidential and will not be shared with third parties without my consent.	10	High	bharadwaj
Sprint-3		USN-6	As a user, I want to be assured that my personal data will be kept confidential and secure, and will only be used for the purpose of mental health prediction.	5	Medium	srikar
Sprint-4	Opting Out	USN-7	As a user, I want to have the option to delete my account and all associated data from the system.	3	Low	Deepak,srikar

6.3 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	12	5-Days	18 Oct 2023	23 Oct 2023	10	23 Oct 2023
Sprint-2	10	6-Days	23 Oct 2023	28 Oct 2023	9	28 Oct 2023
Sprint-3	7	5-Days	28 Oct 2023	03 Nov 2023	6	03 Nov 2023
Sprint-4	5	6-Days	03 Nov 2023	09 Nov 2023	4	09 Nov 2023

7. CODING & SOLUTIONING

Data Pre-Processing

```
[8]  data = data.drop(columns = ['state'],axis = 1)  #unevenly distributed
```

```
[9]  data = data.drop(columns = ['comments'],axis = 1)  #no use for data processing
```

```
[10] data = data.drop(columns = ['Timestamp'],axis = 1)  #no use for data processing
```

```
[11] data = data.drop(columns = ['no_employees'],axis = 1)  #no use for data processing
```

+ Code + Text

```
[ ]  data['Gender'].replace(['Male', 'male', 'M', 'm', 'Male', 'Cis Male',
                             'Man', 'cis male', 'Mail', 'Male-ish', 'Male (CIS)',
                             'Cis Man', 'msle', 'Malr', 'Mal', 'maile', 'Make',], 'Male'

     data['Gender'].replace(['Female', 'female', 'F', 'f', 'Woman', 'Female',
                             'femail', 'Cis Female', 'cis-female/femme', 'Femake',
                             'Female (cis)', 'woman', ], 'Female', inplace=True)

     data["Gender"].replace(['Female (trans)', 'queer/she/they', 'non-binary', 'fluid',
                             'queer', 'Androgyne', 'Trans-female', 'male leaning androgy
                              'Agender', 'A little about you', 'Nah', 'All',
                             'ostensibly male, unsure what that really means', 'Genderqu
                             'Enby', 'p', 'Neuter', 'something kinda male?', 'Guy (-ish)
```

Replacing all null values with corrected values , and deleting the unwanted coloumns

Building the model

```
[37] from sklearn.model_selection import train_test_split
     X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state = 49)
```

```
[38] X_train.shape,X_test.shape,y_train.shape,y_test.shape

     ((872, 21), (375, 21), (872,), (375,))
```

```
[39] from sklearn.linear_model import LogisticRegression
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.neighbors import KNeighborsClassifier
     from sklearn.ensemble import RandomForestClassifier, AdaBoostClassifier, GradientBoostingClassifier
     from xgboost.sklearn import XGBClassifier
     from sklearn.metrics import accuracy_score, roc_curve, confusion_matrix, classification_report, auc
```

```
[40] model_dict = {}
     model_dict['Logistic regression']= LogisticRegression (solver='liblinear', random_state=49)
     model_dict['KNN Classifier'] = KNeighborsClassifier()
     model_dict['Decision Tree Classifier'] = DecisionTreeClassifier(random_state=49)
     model_dict['Random Forest Classifier'] = RandomForestClassifier (random_state=49)
     model_dict['AdaBoost classifier'] = AdaBoostClassifier(random_state=49)
     model_dict['Gradient Boosting Classifier'] = GradientBoostingClassifier(random_state=49)
     model_dict['XGB Classifier'] = XGBClassifier (random_state=49)
```

Model Summary

```
model_dict = {}
model_dict['Logistic regression']= LogisticRegression (solver='liblinear', random_state=49)
model_dict['KNN Classifier'] = KNeighborsClassifier()
model_dict['Decision Tree Classifier'] = DecisionTreeClassifier(random_state=49)
model_dict['Random Forest Classifier'] = RandomForestClassifier (random_state=49)
model_dict['AdaBoost classifier'] = AdaBoostClassifier(random_state=49)
model_dict['Gradient Boosting Classifier'] = GradientBoostingClassifier(random_state=49)
model_dict['XGB Classifier'] = XGBClassifier (random_state=49)
```

```
[41] def model_test(x_train, x_test, y_train, y_test, model, model_name):
       model.fit(x_train,y_train)
       y_pred = model.predict(x_test)
       accuracy = accuracy_score(y_test,y_pred)
       print("==========================================={}==========================================".format(model_name))
       print('Score is: {}'.format(accuracy))

       print()
```

```
for model_name, model in model_dict.items():
    model_test(X_train, X_test, y_train, y_test, model, model_name)
```

```
==========================================Logistic regression==========================================
Score is: 0.8426666666666667

==========================================KNN Classifier==========================================
Score is: 0.8106666666666666

==========================================Decision Tree Classifier==========================================
Score is: 0.7893333333333333

==========================================Random Forest Classifier==========================================
Score is: 0.8293333333333334

==========================================AdaBoost classifier==========================================
Score is: 0.8693333333333333

==========================================Gradient Boosting Classifier==========================================
Score is: 0.8453333333333334

==========================================XGB Classifier==========================================
Score is: 0.8213333333333334
```

8. PERFORMANCE TESTING

8.1 Performance Metrics

```
[43] # adaBoost Classifier has the best classifier
     abc = AdaBoostClassifier(random_state=99)
     abc.fit(X_train,y_train)
     pred_abc = abc.predict(X_test)
     print('Accuracy of AdaBoost=',accuracy_score(y_test,pred_abc))
```

```
Accuracy of AdaBoost= 0.8693333333333333
```

ROC CURVE

```
[50] print(classification_report(y_test,pred_abc))
```

```
              precision    recall  f1-score   support

           0       0.91      0.82      0.86       186
           1       0.84      0.92      0.88       189

    accuracy                           0.87       375
   macro avg       0.87      0.87      0.87       375
weighted avg       0.87      0.87      0.87       375
```

```
[55] # R- Square
     # evaluating testing accuracy
     print(metrics.r2_score(y_test,pred_abc))
```

```
0.47729988052568706
```

```
[57] # MSE (Mean square Error)

     print(metrics.mean_squared_error(y_test,pred_abc))
```

```
0.13066666666666665
```

```
[58] # RMSE (Root Mean Square Error)
     print(np.sqrt(metrics.mean_squared_error(y_test,pred_abc)))
```

```
0.36147844564602555
```

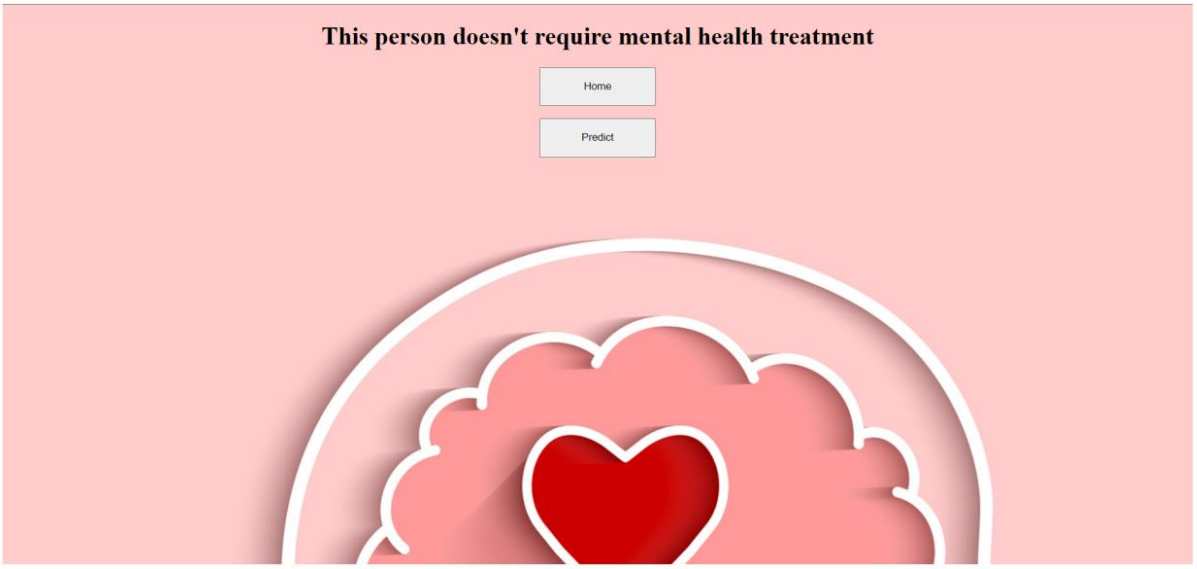
9. RESULTS

9.1 Output Screenshots

Output-1

Output-2

Output-3



10. ADVANTAGES & DISADVANTAGES

Advantages:

Early identification of mental health issues: Machine learning can help identify individuals at risk of developing mental health problems, allowing for early intervention and prevention strategies.

Improved mental health outcomes: Early intervention can lead to improved mental health outcomes for working professionals, reducing the severity and duration of mental health problems.

Increased productivity: Improved mental health can lead to increased productivity and reduced absenteeism among working professionals.

Reduced healthcare costs: Early intervention and prevention can reduce the overall cost of healthcare for mental health problems.

Personalized interventions: Machine learning can be used to develop personalized interventions for mental health problems, tailoring treatment to the individual's needs.

Disadvantages

Data privacy concerns: There are concerns about the privacy of data collected for machine learning models. It is important to ensure that data is collected and used in a way that protects the privacy of individuals.

Data bias: Machine learning models can be biased, reflecting the biases present in the data they are trained on. It is important to identify and address biases in machine learning models.

Interpretability concerns: Machine learning models can be difficult to interpret, making it challenging to understand how they make predictions. This can make it difficult to trust the results of machine learning models.

Limited generalizability: Machine learning models may not generalize well to populations that are different from the population they were trained on. It is important to test the generalizability of machine learning models before deploying them in real-world settings.

Expert reliance: Machine learning models should be developed and implemented in collaboration with mental health experts to ensure that they are accurate, reliable, and appropriate for the target population

11. CONCLUSION

Predicting mental health illness of working professionals using machine learning has the potential to revolutionize the way we identify and address mental health concerns in the workplace. By accurately predicting individuals at risk of developing mental health problems, early intervention and prevention strategies can be implemented, leading to improved mental health outcomes, increased productivity, and reduced healthcare costs.

However, there are several challenges that need to be addressed before machine learning can be effectively used for this purpose. These challenges include data privacy concerns, data bias, interpretability concerns, limited generalizability, and the need for expert reliance.

Despite these challenges, the potential benefits of using machine learning to predict mental health illness of working professionals are significant. With careful consideration of the challenges and ethical considerations, machine learning can be a valuable tool for improving the mental health of the workforce.

12. FUTURE SCOPE

1. Development of more accurate and reliable models

Researchers are continuing to develop new machine learning algorithms that can more accurately predict mental health illness. They are also working to improve the reliability of existing models by addressing issues such as data bias and overfitting.

2. Application to different populations

Current research has focused on predicting mental health illness among working professionals. However, machine learning could also be used to predict mental health illness in other populations, such as students, adolescents, and older adults.

3. Integration with wearable devices

Wearable devices, such as smartwatches and fitness trackers, can collect data on a variety of physiological and behavioral factors that could be used to predict mental health illness. Researchers are exploring ways to integrate data from wearable devices into machine learning models.

4. Development of personalized interventions

Machine learning could be used to develop personalized interventions for mental health problems. For example, machine learning could be used to identify the most effective treatment for an individual based on their unique risk factors and symptoms.

5. Early warning systems

Machine learning could be used to develop early warning systems for mental health problems. These systems could identify individuals at risk of developing mental health problems and alert healthcare providers so that they can intervene early.

6. Prevention of mental health problems

Machine learning could be used to develop interventions to prevent mental health problems from developing in the first place. For example, machine learning could be used to identify individuals who are at risk of developing mental health problems and provide them with preventive interventions, such as stress management training or mindfulness meditation.

13. APPENDIX

The drive link for the ipynb file for mental health illness prediction using ML

https://colab.research.google.com/drive/1NvkXtVpldF86JFwcWijmx-zl43oPXGm6?usp=sharing

The github link for the flask source code

https://github.com/smartinternz02/SI-GuidedProject-603343-1697619457/tree/main/PROJECT%20DEVELOPMENT%20PHASE

Demo Video link

https://drive.google.com/file/d/1j1fQho_s1w53jTyCS3jNRpao7QvZrcI6/view?usp=sharing