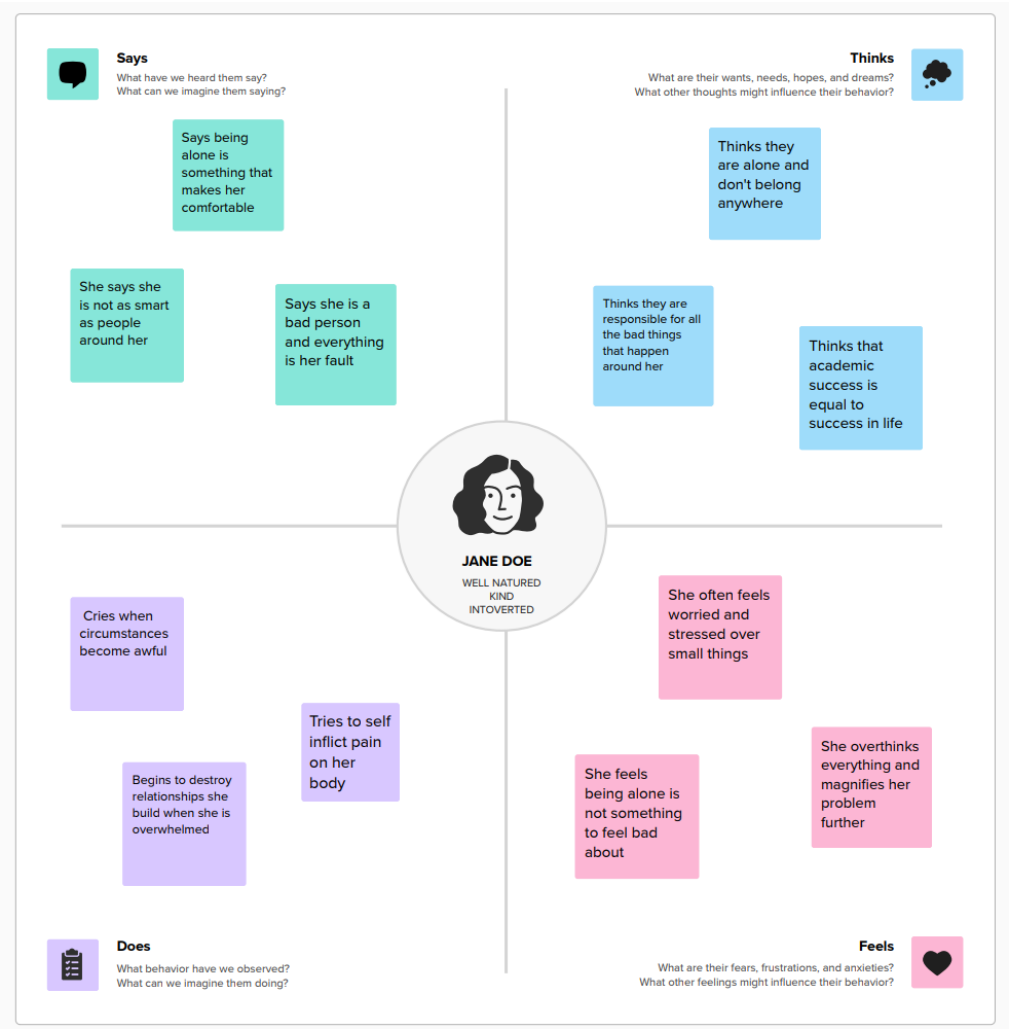
Ideation Phase
Empathize & Discover

Date	5 November 2023
Team ID	Team-591674
Project Name	Project - Predicting Mental health of working professionals
Maximum Marks	4 Marks

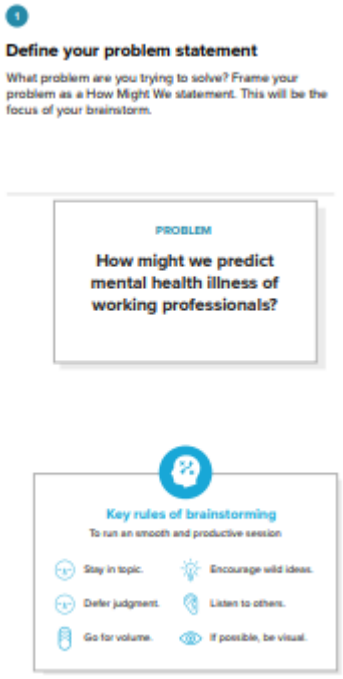
Empathy Map Canvas:



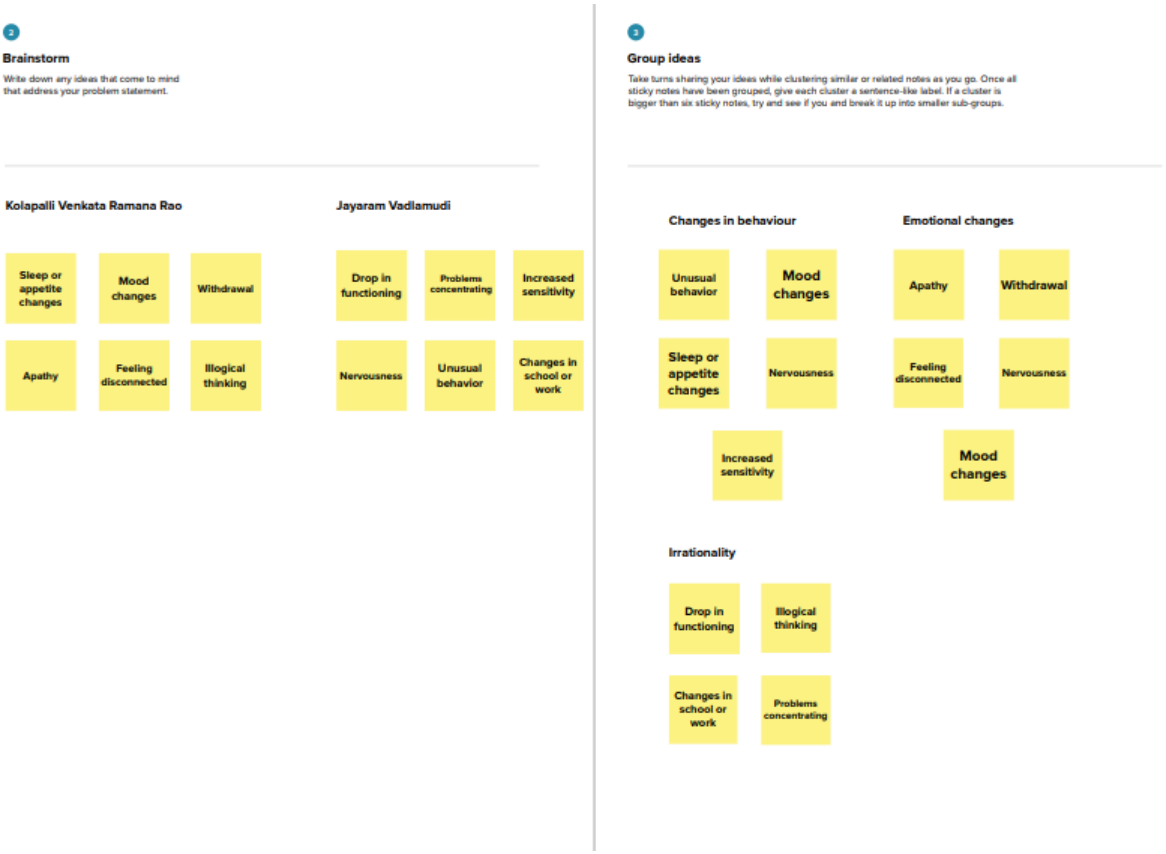
Ideation Phase
Brainstorm & Idea Prioritization Template

Date	6 November 2023
Team ID	Team-591674
Project Name	Project - Predicting Mental health of working professionals
Maximum Marks	4 Marks

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping

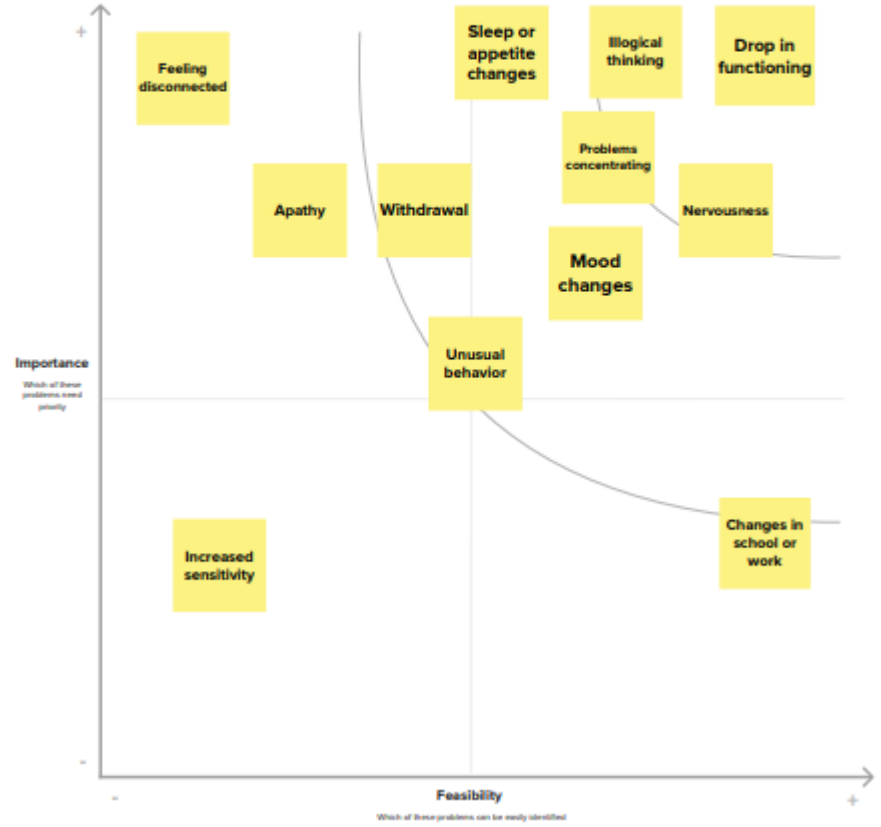


Step-3: Idea Prioritization

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.



Project Design Phase-I
Proposed Solution Template

Date	10 November 2023
Team ID	Team-591674
Project Name	Project - Predicting Mental health of working professionals
Maximum Marks	2 Marks

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	How to identify a mental illness of a working professional.
2.	Idea / Solution description	**Solution:** Developing a system to identify mental illnesses in working professionals using machine learning involves several components. The solution should encompass data collection, feature extraction, model training, and deployment. Here's a step-by-step breakdown: 1. **Data Collection:** ○ Gather anonymized data from various sources such as employee surveys,

		communication logs, and physiological data (if available). ○ Include diverse demographics to ensure the model is inclusive and unbiased. 2. **Feature Extraction:** ○ Extract relevant features from the collected data, including language patterns, sentiment analysis from written communication, and physiological indicators if accessible (heart rate variability, sleep patterns, etc.). ○ Consider external factors such as workload, deadlines, and working hours. 3. **Model Development:** ○ Use machine learning algorithms to train a model on the extracted features. ○ Explore various algorithms such as decision trees, random forests, or deep learning models depending on the complexity of the data. 4. **User Interface:** ○ Develop a user-friendly interface for professionals to interact with the system.

		○ The interface can include a dashboard displaying mental health trends, suggestions for improvement, and resources for seeking help. 5. **Alerts and Intervention:** ○ Implement an alert system to notify both employees and employers when the system detects potential mental health concerns. ○ Provide suggested interventions, which could include stress management resources, counseling services, or wellness programs.
3.	Novelty / Uniqueness	● **Personalized Approach:** Tailor the system to individual users, recognizing that mental health is a highly personal and subjective matter. ● **Continuous Monitoring:** Implement continuous monitoring to detect early signs of mental health issues, allowing for timely intervention. ● **Multimodal Data Integration:** Integrate multiple data sources for a holistic view, considering both subjective (surveys)

		and objective (physiological) data
4.	Social Impact / Customer Satisfaction	● **Reducing Stigma:** By providing a tool that is proactive in identifying mental health concerns, the system contributes to reducing the stigma associated with mental health issues. ● **Promoting Well-being:** Fostering a workplace that actively monitors and supports mental health contributes to a positive working environment and employee well-being.
5.	Business Model (Revenue Model)	● **Subscription-Based Model:** Charge companies a subscription fee based on the number of employees for access to the mental health monitoring system. ● **Consultation Services:** Offer consultation services to companies for interpreting and addressing mental health data. ● **Customization Packages:** Provide customization options for companies that want additional features or integration with existing HR systems
6.	Scalability of the Solution	● **Cloud-Based Infrastructure:** Implement the solution on a scalable cloud

		infrastructure, allowing it to handle an increasing number of users and data points efficiently. ● **API Integration:** Provide APIs for seamless integration with existing HR and communication platforms, facilitating widespread adoption without significant disruptions. ● **Continuous Improvement:** Incorporate feedback loops and periodic model updates to ensure the system adapts to evolving workplace dynamics and incorporates the latest advancements in mental health research.

Project Design Phase-I
Solution Architecture

Date	10 November 2023
Team ID	Team-591674
Project Name	Project - Predicting Mental health of working professionals
Maximum Marks	4 Marks

Solution Architecture:

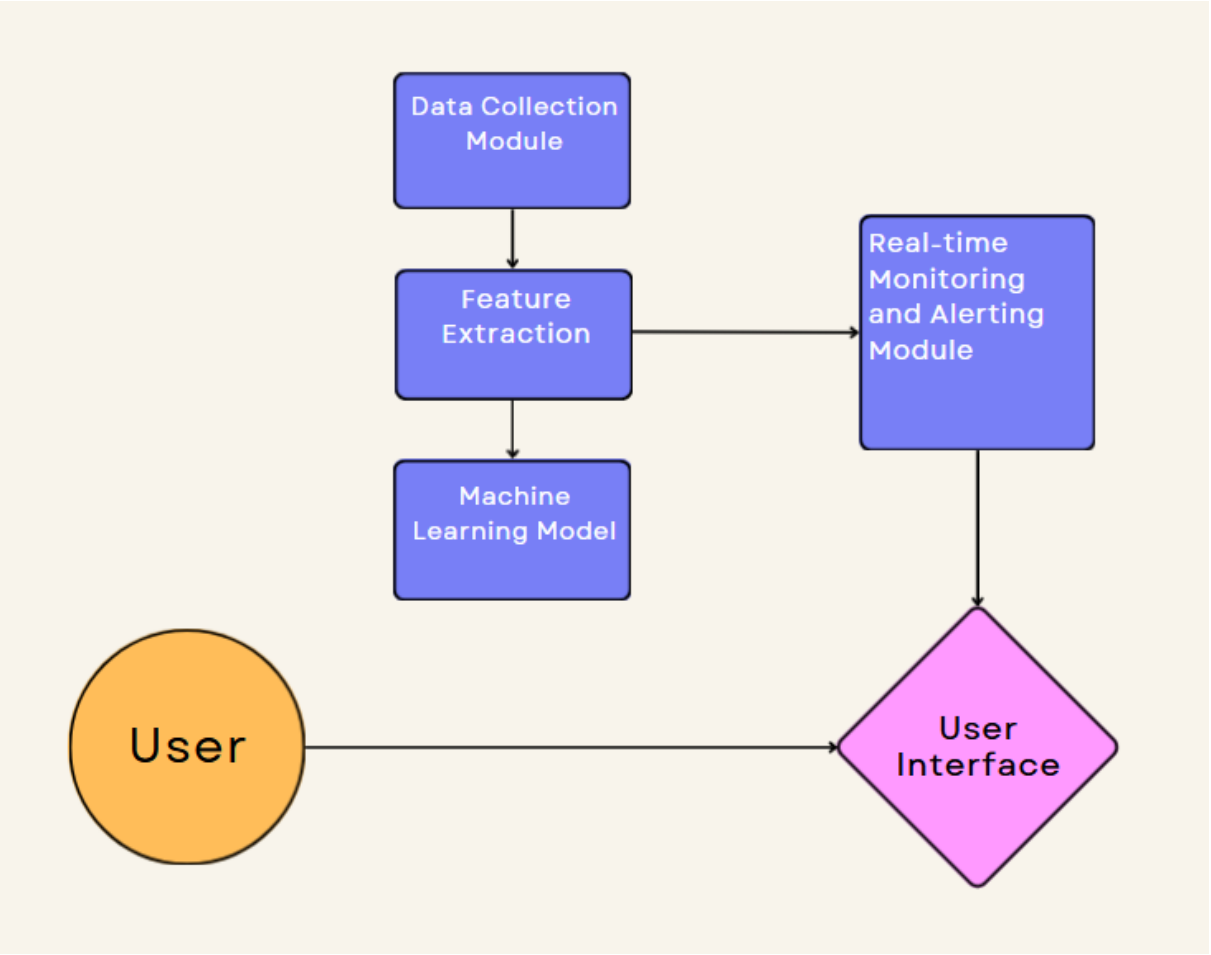
Solution Architecture for Identifying Mental Illness in Working Professionals using Machine Learning

1. **System Architecture Overview:**
    - **Components:**
        - **Data Collection Module**
        - **Feature Extraction Module**
        - **Machine Learning Model**
        - **Real-time Monitoring Module**
        - **Alert and Intervention Module**
        - **User Interface**
    - **Integration Points:**
        - **HR Systems**
        - **Communication Platforms (Email, Chat)**
        - **Performance Metrics**
    - **Deployment Environment:**
        - **Cloud-based infrastructure for scalability and accessibility.**
        - **Microservices architecture for modularity and maintainability.**
2. **Data Flow:**
    - **Data Collection:**
        - **Collect data from various sources (communication logs, performance metrics, physiological data if available).**
        - **Preprocess and anonymize data to ensure privacy.**
    - **Feature Extraction:**
        - **Extract relevant features from the collected data.**
        - **Utilize natural language processing for communication patterns.**
        - **Incorporate performance metrics and physiological data.**
    - **Machine Learning Model:**

        - **Train a personalized machine learning model for each user.**
        - **Utilize a combination of supervised and unsupervised learning techniques.**
        - **Continuously update models to adapt to changes in behavior.**
    - **Real-time Monitoring:**
        - **Implement a real-time monitoring system to analyze ongoing data.**
        - **Utilize anomaly detection algorithms to identify deviations from baseline behavior.**
        - **Trigger alerts for potential mental health concerns.**
    - **Alert and Intervention:**
        - **Send alerts to designated personnel or managers in case of identified concerns.**
        - **Integrate with existing employee assistance programs (EAPs) for intervention.**
        - **Provide resources and guidance for supportive measures.**
    - **User Interface:**
        - **Develop a user-friendly dashboard for both employees and administrators.**
        - **Display personalized insights for individual users.**
        - **Enable administrators to view aggregated analytics and trends across the organization.**
3. **Features and Development Phases:**
    - **Phase 1: Data Collection and Integration**
        - **Implement connectors for HR systems, communication platforms, and performance metrics.**
        - **Set up data collection pipeline and preprocessing.**
    - **Phase 2: Feature Extraction and Model Training**
        - **Develop algorithms for feature extraction.**
        - **Train initial machine learning models with historical data.**
    - **Phase 3: Real-time Monitoring and Alerting**
        - **Implement real-time monitoring capabilities.**
        - **Integrate alerting mechanisms based on anomaly detection.**
    - **Phase 4: User Interface Development**
        - **Design and implement user interfaces for employees and administrators.**
        - **Ensure accessibility and responsiveness.**
    - **Phase 5: Continuous Improvement**
        - **Implement feedback loops for continuous model improvement.**
        - **Regularly update features and user interfaces based on user feedback.**
4. **Solution Requirements and Specifications:**

- **Data Security and Privacy:**
  - **Ensure compliance with data protection regulations.**
  - **Implement encryption and access controls.**
- **Scalability:**
  - **Design for scalability to accommodate a growing number of users.**
  - **Utilize cloud-based services for elastic scaling.**
- **Interoperability:**
  - **Provide APIs for easy integration with existing systems.**
  - **Ensure compatibility with a variety of data sources.**
- **User Training and Support:**
  - **Develop user manuals and training materials.**
  - **Provide ongoing support and training for administrators.**

Solution Architecture Diagram:

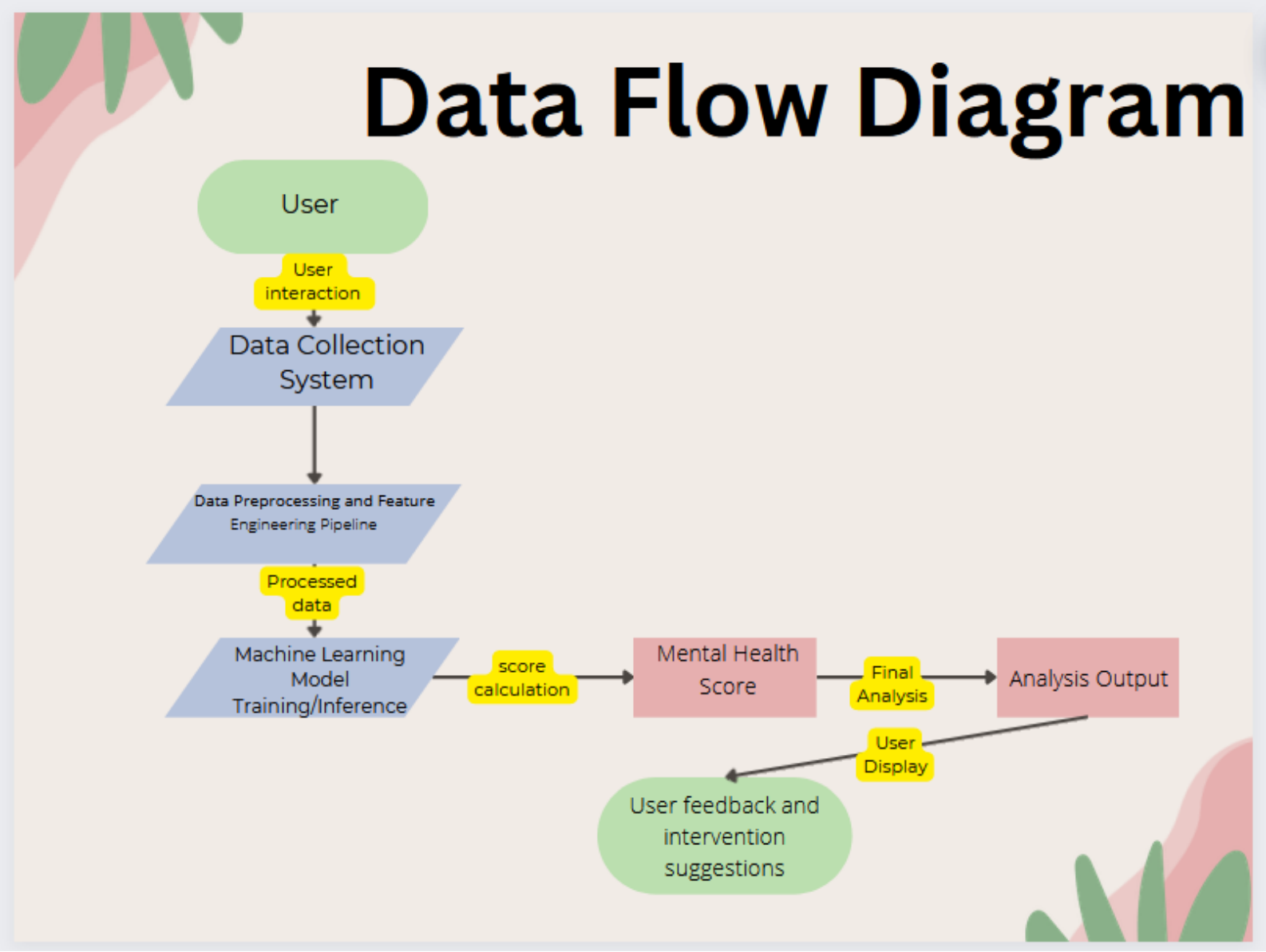


Project Design Phase-II
Data Flow Diagram & User Stories

Date	03 October 2022
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx
Maximum Marks	4 Marks

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.


Data Flow Diagram
User
User interaction
Data Collection System
Data Preprocessing and Feature Engineering Pipeline
Processed data
Machine Learning Model Training/Inference
score calculation
Mental Health Score
Final Analysis
Analysis Output
User Display
User feedback and intervention suggestions

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	User Initiated Assessment	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	The user can initiate a mental health assessment from their account settings.	High	1.0
	Data Collection	USN-2	As a user, I will receive confirmation email once I have registered for the application	The system collects relevant work performance metrics, such as productivity, task completion rates, and deadlines met.	High	1.0
		USN-3	As a user, I can register for the application through Facebook	Privacy is maintained, and only work-related communication data is accessed.	High	1.0
	Machine Learning Model	USN-4	As a user, I can register for the application through Gmail	The model adapts to individual differences in behavior and work patterns.	High	1.1

		USN-5	As a user, I can log into the application by entering email & password	Users receive a detailed and easily understandable explanation of their mental health score.	Medium	1.1
	Continuous Monitoring and Feedback	USN-6	As a user, I can login into the application by entering phone number & OTP	The system continuously monitors user behavior for changes from established baselines.	High	1.2

Team: 591674
Venkata Ramana Rao - 21BCE8249
JayaRam – 21BCE8201

Project Design Phase 1
Technology Stack (Architecture & Stack)

Date	1 December 2023
Team ID	591674
Project Name	**Predicting Mental Health Illness Of Working Professionals Using Machine Learning**
Maximum Marks	4 Marks

1 Introduction :

Project Overview: Predicting Mental Illness in Working Professionals using Machine Learning

Problem: Mental illness among working professionals is a significant concern with detrimental impacts on individual wellbeing and organizational productivity. Early identification and intervention are crucial for improved outcomes.

Solution: This project aims to develop a web-based platform that utilizes machine learning algorithms to predict mental illness risk in working professionals based on their online activity and self-reported information.

Technical Approach:

1. Data Collection: User data will be collected through the platform's interface, including self-reported questionnaires, social media activity, and wearables data (optional).
2. Data Pre-processing and Feature Engineering: User data will be pre-processed and transformed into features suitable for machine learning analysis. Feature engineering techniques like sentiment analysis and linguistic analysis will be employed.
3. Model Training and Deployment: Ensemble machine learning models, like Logistic Regression and Random Forest, will be trained on the extracted features to predict the risk of mental illness.
4. User Interface: A user-friendly web interface will be developed for professionals to access the platform, complete assessments, and receive personalized feedback on their mental health risk.

Expected Outcomes:

- Early Identification: The project aims to identify individuals at risk of mental illness early, allowing for timely intervention and support.
- Improved Awareness: The platform will raise awareness about mental health issues among working professionals and encourage individuals to seek help.
- Enhanced Productivity: Early detection and intervention can help improve individual and organizational productivity by reducing absenteeism and presenteeism.

Next Steps:

- Prototype Development: Develop a functional prototype of the platform with core functionalities.
- Evaluation and Testing: Conduct pilot studies with working professionals to evaluate the accuracy and effectiveness of the model.

Refinement and Deployment: Refine the platform and model based on feedback and pilot testing results. Deploy the platform for broader use by working professionals.

2 LITERATURE SURVEY

Objectives:

- Analyze existing research on mental illness prediction using machine learning.
- Identify relevant features and machine learning techniques used for prediction.
- Assess the accuracy and limitations of existing solutions.

Scope:

- Research papers published in the last 5 years focusing on predicting mental health in working professionals using machine learning.
- Focus on features derived from online activity, self-reported information, and wearable data.
- Include studies utilizing various machine learning algorithms and architectures.

Methodology:

1. Conduct a comprehensive literature search using academic databases (e.g., PubMed, Google Scholar) with relevant keywords.
2. Screen and select relevant research papers based on inclusion/exclusion criteria.
3. Analyze the selected papers to extract key information, including:
    - Data sources and features used
    - Machine learning algorithms employed
    - Prediction accuracy and performance metrics
    - Limitations and challenges identified
4. Synthesize the findings and identify common trends, gaps, and opportunities in existing research.

Expected Outcomes:

- A comprehensive understanding of existing research on mental illness prediction using machine learning.
- Identification of promising features and machine learning techniques for predicting mental health risk in working professionals.
- Critical insights into the strengths, weaknesses, and limitations of current approaches.

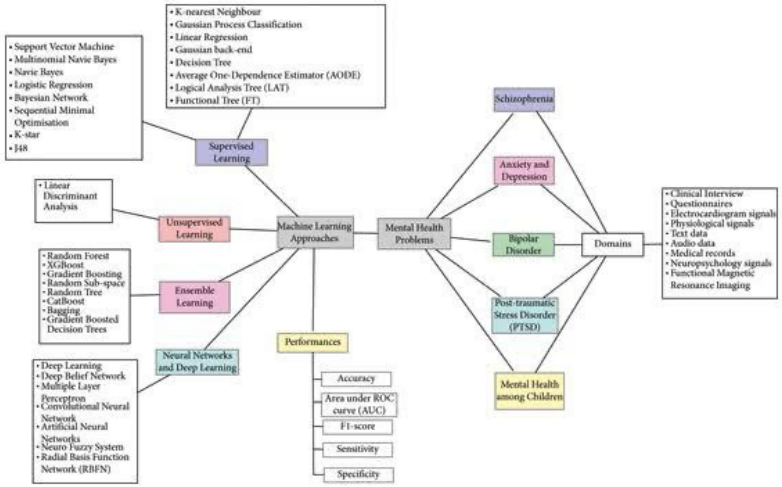


Diagram: Table-1 :

Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Mobile App, Chatbot	HTML, CSS, JavaScript (React or Angular), React Native
2.	Application Logic-1	Pre-processing and cleaning of user data	Python libraries (e.g., pandas, scikitlearn)

3.	Application Logic-2	Feature engineering and extraction (e.g., sentiment analysis, linguistic features)	NLP libraries (e.g., spaCy, NLTK)
4.	Application Logic-3	Training and deployment of machine learning models for prediction	TensorFlow, PyTorch
5.	Database	Store user data, preferences, and historical information	MySQL or NoSQL (e.g., MongoDB)
6.	Cloud Database	Store and retrieve data from the cloud	AWS DynamoDB, Google Cloud Firestore, Azure
7.	File Storage	Store necessary files or documents securely	Cloud-based storage (e.g., AWS S3, Google Cloud
8.	External API-1	Integrate external services for additional data	Relevant APIs (e.g., mental health survey APIs)
9.	External API-2	Integrate external services for additional data	Aadhar API, etc.
10.	Machine Learning Model	Train a model on relevant features to predict mental health	Python, scikit-learn, TensorFlow, PyTorch
11.	Infrastructure (Server / Cloud)	Deploy on local servers or cloud	Local servers, AWS, Google Cloud, Azure, Docker, Kubernetes

Table-2: Application Characteristics:

S.No	**Characteristics**	**Description**	**Technology**
1.	Open-Source Frameworks	Utilize open-source frameworks for development	Django, Flask (backend), React, Angular (frontend)
2.	Security Implementations	Implement strong security measures.	Encryption (SSL/TLS), RBAC, Compliance standards.
3.	Scalable Architecture	Design for scalability (Microservices, Load balancing)	Microservices, Load balancing (Nginx)
S.No	**Characteristics**	**Description**	**Technology**

4.	Availability	Ensure high availability through redundancy and failover	Load balancers, Distributed server architecture
5.	Performance	Optimize for quick and reliable predictions	Caching mechanisms, CDN (Content Delivery Network)

3 IDEATION AND PROPOSED SOLUTION

Ideation Phase:

1. Brainstorming and Problem Analysis:
    - Identify the key challenges and pain points associated with predicting mental illness in working professionals.
    - Conduct brainstorming sessions to generate innovative ideas for solutions.
    - Evaluate potential solutions based on feasibility, effectiveness, and impact.
2. User Research and Requirements Gathering:
    - Conduct surveys and interviews with working professionals to understand their needs and expectations for a mental health prediction system.
    - Identify user personas and user stories to define the system's functionalities and features.
    - Prioritize features based on user needs and feasibility constraints.
3. Concept Development and Iteration:
    - Develop initial prototypes and mockups of the proposed solution.
    - Gather feedback from stakeholders and users to refine and iterate on the concept. ○ Define the overall system architecture and identify key technologies required.

Proposed Solution:

- Machine Learning-based Mental Health Prediction Platform:
    - Develop a web-based platform where working professionals can access mental health assessments and receive personalized feedback on their mental health risk.
    - Utilize machine learning models trained on data from online activity, self-reported information, and potentially wearable data to predict the risk of mental illness.
    - Integrate with existing mental health resources and provide users with personalized recommendations for further support.

Key Features:

- Self-assessment tools and questionnaires
- Secure data storage and privacy protection
- Personalized risk assessment reports
- Educational resources and coping mechanisms
- Integration with external mental health services

Differentiation from Existing Solutions:

- Focus on working professionals and their specific needs
- Utilize multiple data sources for comprehensive risk assessment
- Employ advanced machine learning algorithms for high accuracy
- Offer personalized recommendations and support resources
- Prioritize user privacy and data security

Expected Benefits:

- Early identification of individuals at risk of mental illness
- Increased awareness and understanding of mental health
- Improved access to appropriate mental health services
- Enhanced well-being and productivity for working professionals

Project Planning Phase 2

5 Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	1 December 2023
Team ID	591674
Project Name	**Predicting Mental Health Illness Of Working Professionals Using Machine Learning**
Maximum Marks	8 Marks

4 REQUIREMENT ANALYSIS

4.1 Functional Requirements:

- User Management: Create profiles, manage account settings, and access personalized information.
- Assessment Tools: Complete self-assessment questionnaires and surveys related to mental health.
- Data Collection: Securely upload and integrate data from various sources (social media, wearables, etc.).
- Prediction Engine: Utilize machine learning models to predict the risk of mental illness.
- Risk Assessment Report: Generate personalized reports with risk scores, insights, and potential explanations.
- Resource Recommendations: Provide access to personalized resources and support options based on individual needs.
- Data Security and Privacy: Ensure secure data storage, encryption, and user privacy compliance.

4.2 Non-Functional Requirements:

- Performance: The platform should respond quickly and efficiently to user actions.
- Scalability: The platform should be able to handle a large number of users and data points.
- Availability: The platform should be available 24/7 with minimal downtime.
- Security: The platform should be secure against unauthorized access and data breaches.
- Usability: The platform should be user-friendly and easy to navigate.
- Accessibility: The platform should be accessible to users with disabilities.

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	2
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	2
Sprint-2		USN-3	As a user, I can register for the application through Facebook	2	Low	2
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	2
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	2

	Dashboard					

Project Tracker, Velocity & Burndown Chart: (4 Marks)

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022		
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		

Velocity:

Eg: Imagine the team's velocity is 20 points per sprint (story points per day)

$$AV = \frac{Sprint\ Duration}{Velocity} = \frac{6\ Days}{20\ points}$$

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. Unfortunately, specific values for work completed are not available for Sprints 2, 3, and 4, so the chart cannot be generated at this point.

Note: In practice, team members and their availability would need to be considered for more accurate estimations and planning. This table assumes a simple case with predefined story points and priorities and a team of 2 members.

5. Project Design

5.1 Data Flow Diagrams & User Stories:

- Develop data flow diagrams to visualize the flow of data through the system, including data sources, processing steps, and storage locations.
- Create user stories to capture user requirements from different perspectives, including professionals seeking assessments, administrators managing user accounts, and system developers.

5.2 Solution Architecture:

- Define the overall architecture of the system, including the front-end, back-end, database, and other components.
- Identify the technologies and tools that will be used to implement each component.

Design the communication protocols between different components.

6. Project Planning & Scheduling

6.1 Technical Architecture:

- Choose a suitable cloud platform (AWS, Azure, GCP) to host the system.
- Select the appropriate database technology (MySQL, MongoDB) to store data.
- Choose a machine learning framework (TensorFlow, PyTorch) for model development.
- Utilize APIs for data integration and external services.

6.2 Sprint Planning & Estimation:

- Divide the project into manageable sprints, each with specific goals and deliverables.
- Estimate the time and resources required for each task within each sprint.
- Use tools and techniques like agile methodologies and scrum boards for efficient project management.

6.3 Sprint Delivery Schedule:

- Create a detailed timeline for sprint execution, including start and end dates, key milestones, and dependencies.
- Track progress regularly and make adjustments as needed to ensure project completion on time and within budget.
- Conduct sprint reviews and retrospectives to evaluate progress, identify areas for improvement, and plan for future sprints.

7. CODING & SOLUTIONING

Feature 1

```python
# Function to extract work-related stress features from employee survey data
def extract_stress_features(survey_data):
    workload_score = survey_data["workload_rating"]
    job_satisfaction_score = survey_data["job_satisfaction_rating"]
    pressure_score = survey_data["pressure_rating"]
    stress_score = (workload_score + pressure_score) / 2 - job_satisfaction_score
    return stress_score

# Function to obtain stress indicators from work performance data
def get_stress_indicators(performance_data):
    lateness_count = performance_data["lateness_count"]
    absenteeism_rate = performance_data["absenteeism_rate"]
    productivity_change = performance_data["productivity_change"]
    stress_indicators = [lateness_count, absenteeism_rate, productivity_change]
    return stress_indicators

# Function to analyze social media interactions for stress
def analyze_social_media(social_media_data):
    negative_words_count = count_negative_words(social_media_data)
    stress_related_topics = identify_stress_related_topics(social_media_data)
    stress_score = negative_words_count + len(stress_related_topics)
    return stress_score
```

Feature 2

```
# Function to extract sleep features from wearable data
def extract_sleep_features(wearable_data):
    sleep_duration = wearable_data["sleep_duration"]
    deep_sleep_percentage = wearable_data["deep_sleep_percentage"]
    sleep_disruptions = wearable_data["sleep_disruptions_count"]
    sleep_quality_score = (sleep_duration * deep_sleep_percentage) / sleep_disruptions
    return sleep_quality_score

# Function to analyze sleep diary data
def analyze_sleep_diary(diary_data):
    sleep_onset_time = diary_data["sleep_onset_time"]
    wake_up_time = diary_data["wake_up_time"]
    sleep_duration = wake_up_time - sleep_onset_time
    sleep_quality_rating = diary_data["sleep_quality_rating"]
    return sleep_duration, sleep_quality_rating

# Function to infer sleep quality from mobile app data
def infer_sleep_quality(app_data):
    late_night_app_usage = app_data["late_night_app_usage"]
    phone_activity_
```

DataBase Schema:

	A	B	C	D	E
1	Column Name	Data Type	Description		
2	employee_id	integer	Unique identifier for each employee		
3	survey_id	integer	Unique identifier for each survey response		
4	workload_rating	integer	Self-reported rating of workload (1-5)		
5	job_satisfaction_	integer	Self-reported rating of job satisfaction (1-5)		
6	pressure_rating	integer	Self-reported rating of perceived pressure (1-5)		
7	lateness_count	integer	Number of times late for work in a specific period		
8	absenteeism_rat	float	Percentage of days absent from work over a time period		
9	productivity_char	float	Percentage change in productivity compared to a baseline		
10	social_media_da	string	Text data extracted from social media interactions		
11	stress_score	float	Combined score representing work-related stress		

8. Performance Testing

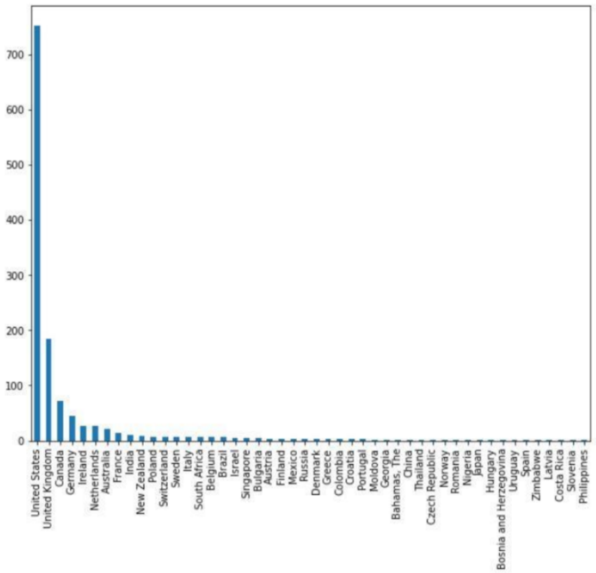
8.1 Performance Metrics:

- Response Time: Measure the time it takes for the platform to respond to user actions.
- Throughput: Determine the number of concurrent users the platform can handle.
- Resource Utilization: Monitor CPU, memory, and network usage to identify bottlenecks.

- Scalability: Test the platform's ability to handle increased load.
- Availability: Measure the uptime and downtime of the system.
- Accuracy: Evaluate the accuracy of the machine learning models in predicting mental illness risk.
- Precision: Determine the proportion of true positives among all positive predictions.
- Recall: Measure the proportion of true positives correctly identified.

F1-score: Calculate a harmonic mean of precision and recall.

9 Results Output Screenshots:




```
data.isnull().sum()
```

```
Age                          0
Gender                       0
self_employed               18
family_history               0
treatment                    0
work_interfere             264
no_employees                 0
remote_work                  0
tech_company                 0
benefits                     0
care_options                 0
wellness_program             0
seek_help                    0
anonymity                    0
leave                        0
mental_health_consequence    0
phys_health_consequence      0
coworkers                    0
supervisor                   0
mental_health_interview      0
phys_health_interview        0
mental_vs_physical           0
obs_consequence              0
dtype: int64
```

```
data['self_employed'].value_counts()
```

```
No      1095
Yes      146
Name: self_employed, dtype: int64
```

```
data['self_employed'].fillna('No', inplace=True)
```

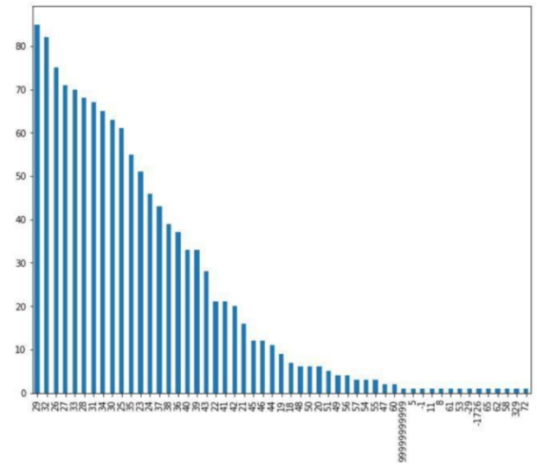
```
data['work_interfere'].value_counts()
```

```
Sometimes    465
Never        213
Rarely       173
Often        144
Name: work_interfere, dtype: int64
```

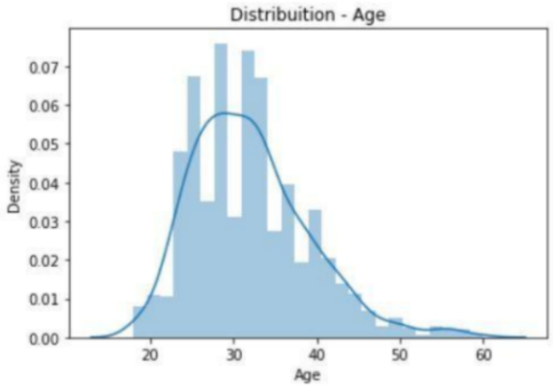
```
data['work_interfere'].fillna('N/A',inplace=True)
```

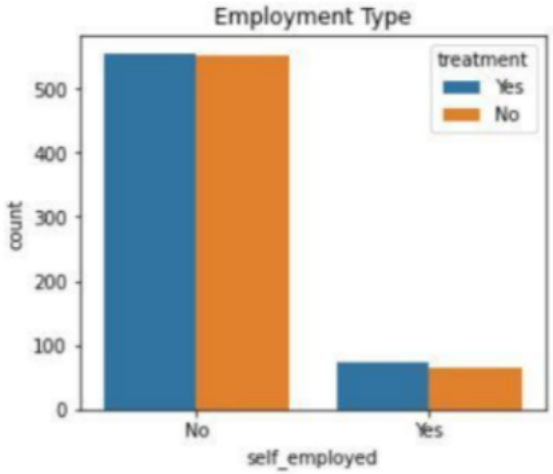
```
data['Age'].value_counts().plot(kind='bar',figsize=(10,8))
```

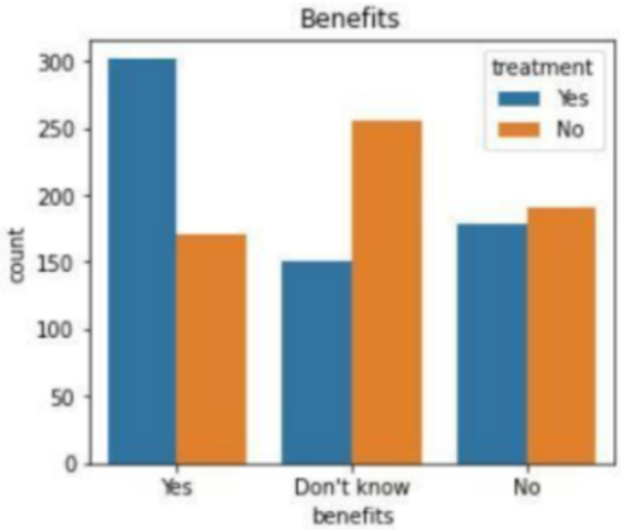
```
<AxesSubplot:>
```

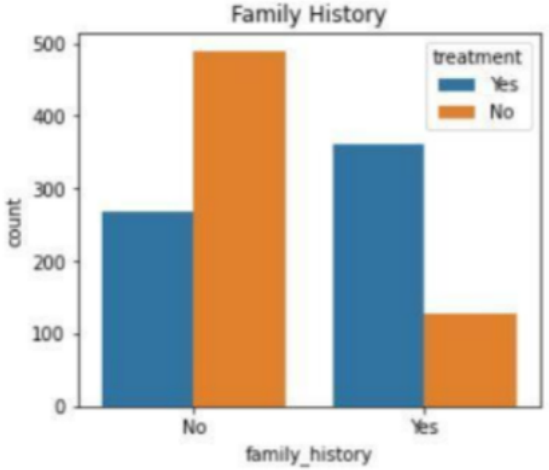


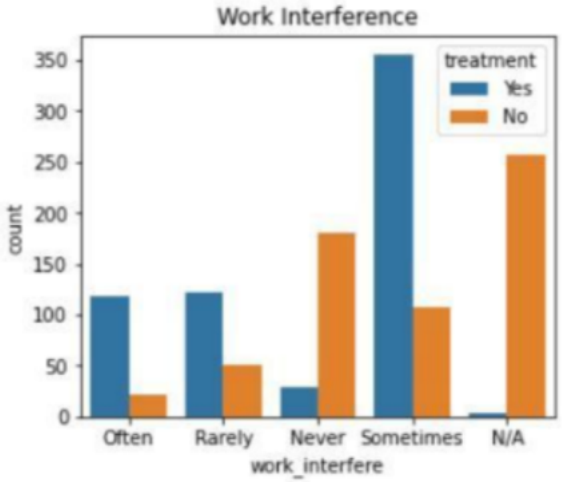
```
sb.distplot(data["Age"])
plt.title("Distribuition - Age")
plt.xlabel("Age")
```

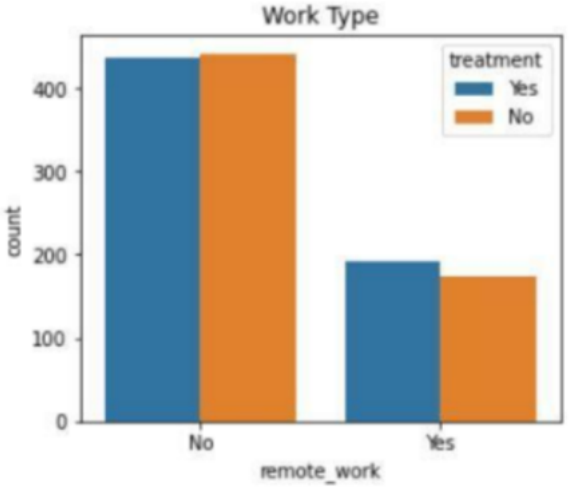


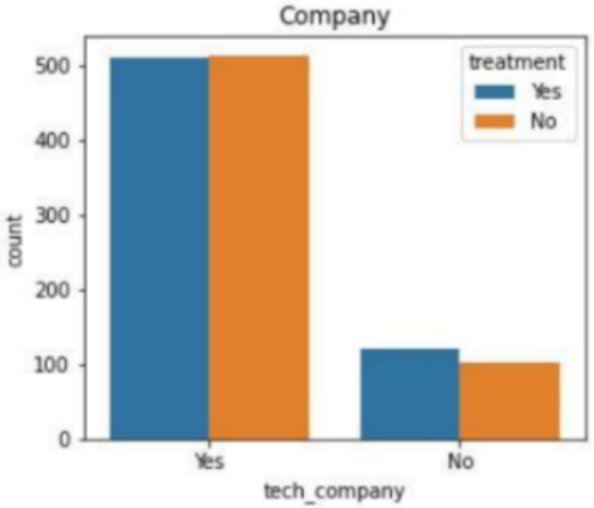


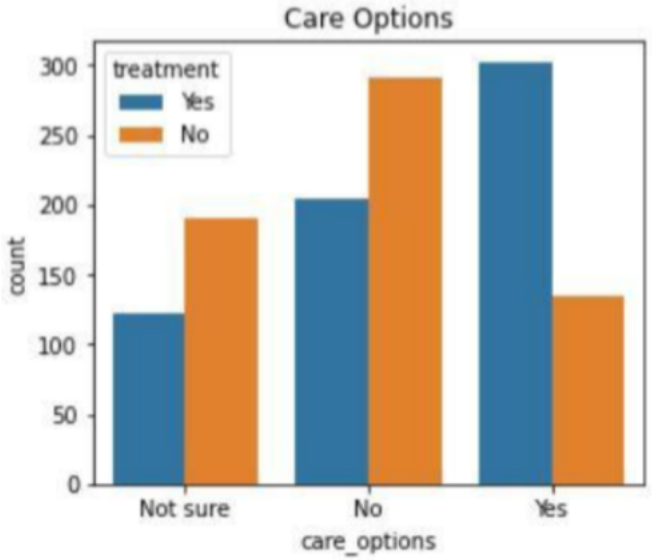

Benefits
treatment
Yes
No
count
300
250
200
150
100
50
0
Yes
Don't know
No
benefits

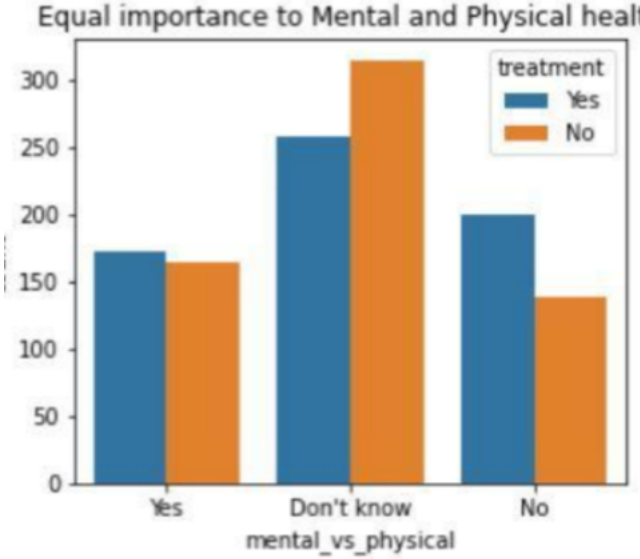

Family History
treatment
Yes
No
count
500
400
300
200
100
0
No
Yes
family_history

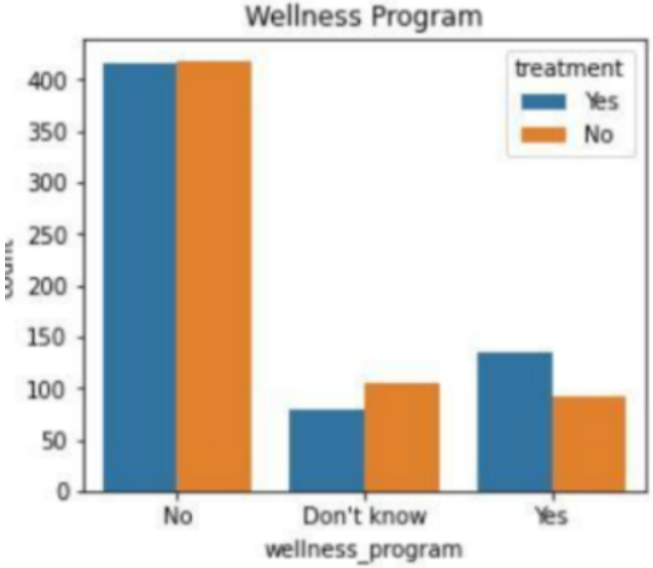

Work Interference
treatment
Yes
No
count
350
300
250
200
150
100
50
0
Often
Rarely
Never
Sometimes
N/A
work_interfere

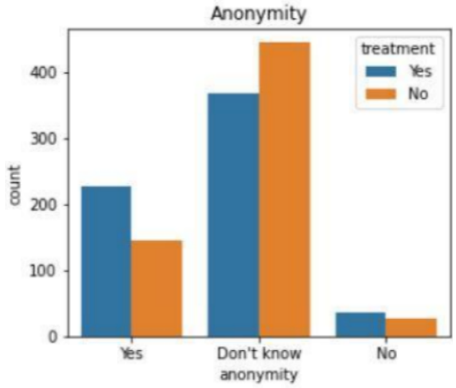

Work Type
treatment
Yes
No
count
400
300
200
100
0
No
Yes
remote_work

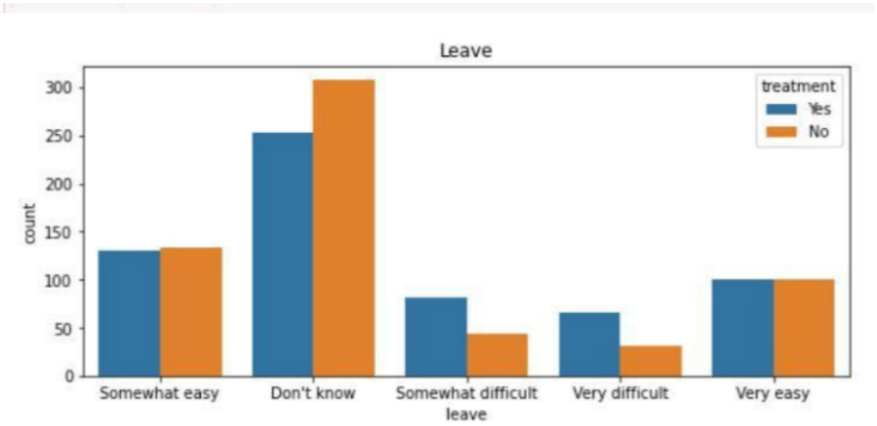

Company
treatment
Yes
No
count
0
100
200
300
400
500
Yes
No
tech_company

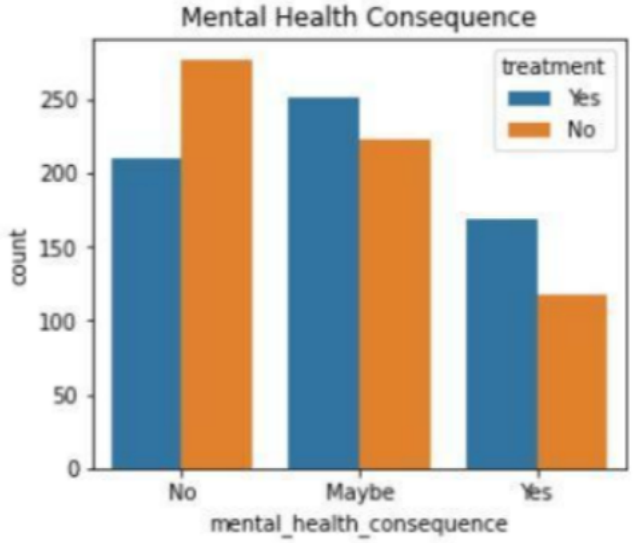

Care Options
treatment
Yes
No
count
0
50
100
150
200
250
300
Not sure
No
Yes
care_options

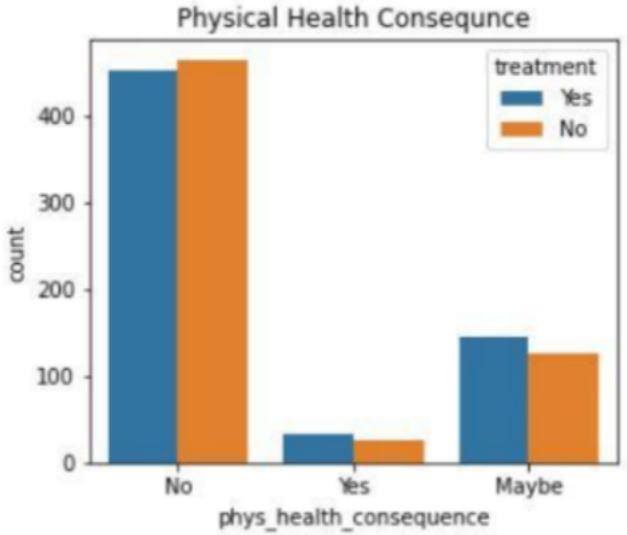

Equal importance to Mental and Physical health
treatment
Yes
No
0
50
100
150
200
250
300
Yes
Don't know
No
mental_vs_physical

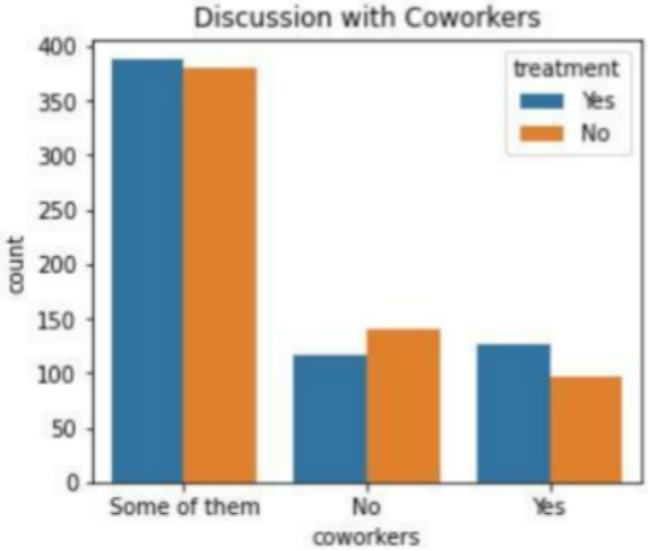

Wellness Program
treatment
Yes
No
0
50
100
150
200
250
300
350
400
No
Don't know
Yes
wellness_program

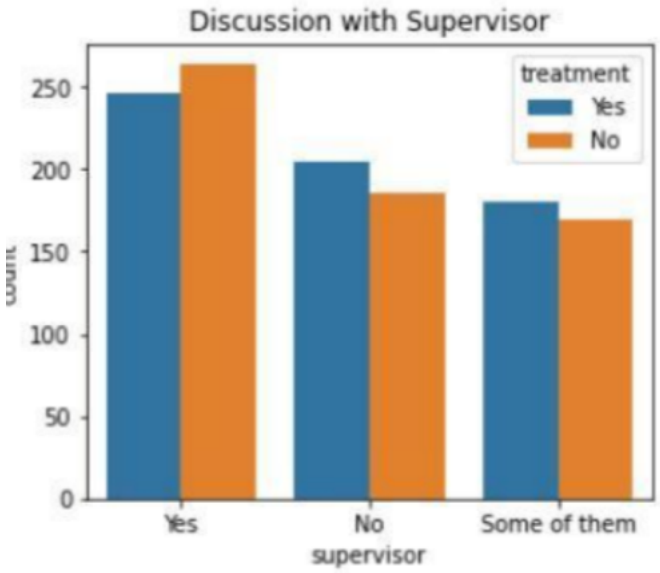

Anonymity
treatment
Yes
No
count
0
100
200
300
400
Yes
Don't know
No
anonymity

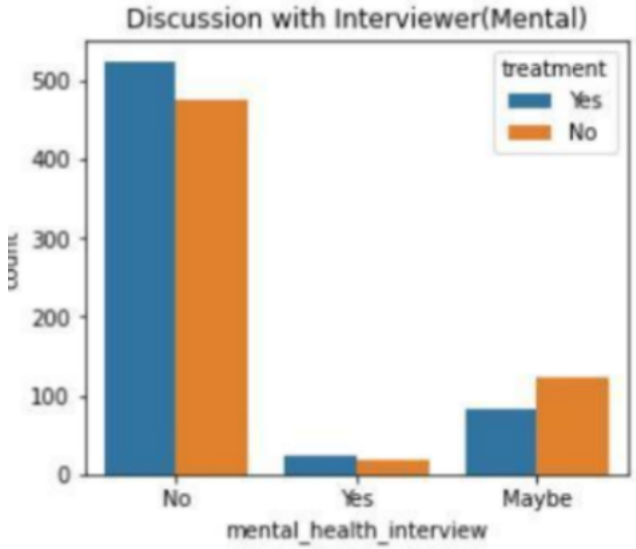

Leave
treatment
Yes
No
count
300
250
200
150
100
50
0
Somewhat easy
Don't know
Somewhat difficult
Very difficult
Very easy
leave

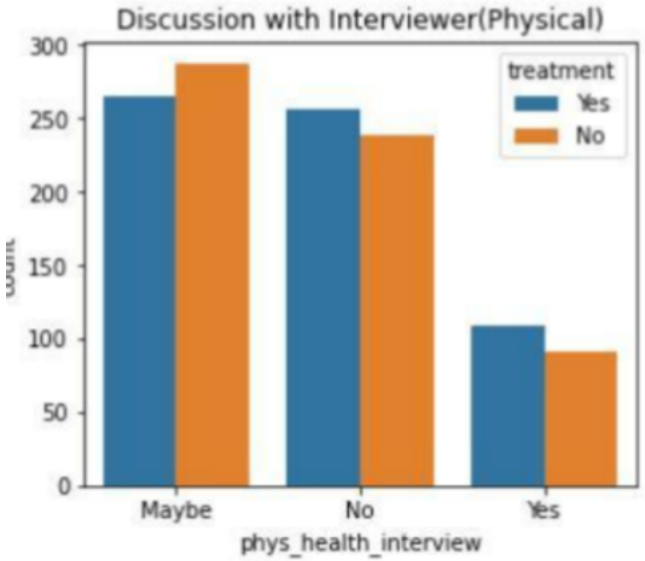

Mental Health Consequence
treatment
Yes
No
count
250
200
150
100
50
0
No
Maybe
Yes
mental_health_consequence

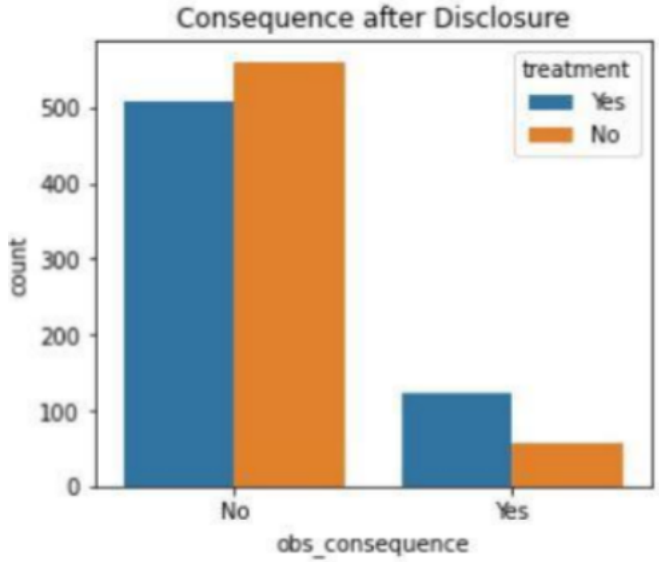

Physical Health Consequnce
treatment
Yes
No
count
400
300
200
100
0
No
Yes
Maybe
phys_health_consequence


Discussion with Coworkers
treatment
Yes
No
count
400
350
300
250
200
150
100
50
0
Some of them
No
Yes
coworkers


Discussion with Supervisor
treatment
Yes
No
250
200
150
100
50
0
Yes
No
Some of them
supervisor

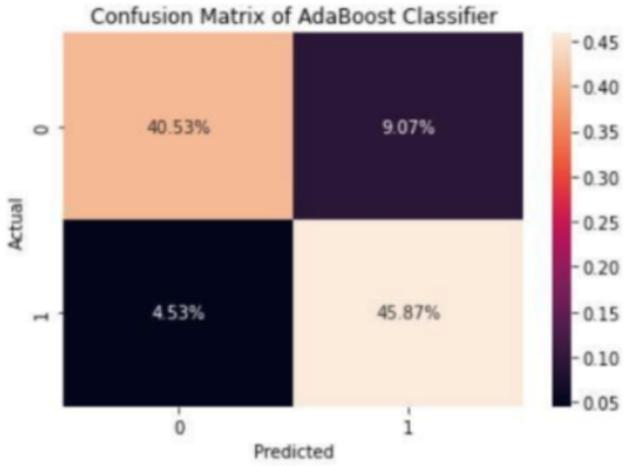

Discussion with Interviewer(Mental)
treatment
Yes
No
500
400
300
200
100
0
No
Yes
Maybe
mental_health_interview


Discussion with Interviewer(Physical)
treatment
Yes
No
300
250
200
150
100
50
0
Maybe
No
Yes
phys_health_interview


Consequence after Disclosure
treatment
Yes
No
count
500
400
300
200
100
0
No
Yes
obs_consequence

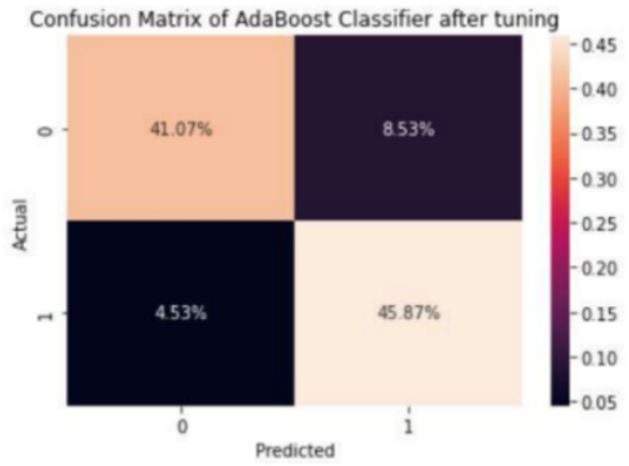
```
cf_matrix = confusion_matrix(y_test, pred_abc)
sb.heatmap(cf_matrix/np.sum(cf_matrix), annot=True, fmt='.2%')
plt.title('Confusion Matrix of AdaBoost Classifier')
plt.xlabel('Predicted')
plt.ylabel('Actual')
```

```
Text(33.0, 0.5, 'Actual')
```




```
cf_matrix = confusion_matrix(y_test, pred_abc_tuned)
sb.heatmap(cf_matrix/np.sum(cf_matrix), annot=True, fmt='.2%')
plt.title('Confusion Matrix of AdaBoost Classifier after tuning')
plt.xlabel('Predicted')
plt.ylabel('Actual')
```

```
Text(33.0, 0.5, 'Actual')
```



Classification report:

```
1 print(classification_report(y_test,pred_abc))
```

	precision	recall	f1-score	support
0	0.90	0.82	0.86	186
1	0.83	0.91	0.87	189
accuracy			0.86	375
macro avg	0.87	0.86	0.86	375
weighted avg	0.87	0.86	0.86	375

```
1 print(classification_report(y_test,pred_abc_tuned))
```

	precision	recall	f1-score	support
0	0.96	0.72	0.82	195
1	0.76	0.97	0.85	180
accuracy			0.84	375
macro avg	0.86	0.84	0.84	375
weighted avg	0.87	0.84	0.84	375

10. Advantages & Disadvantages
Advantages:

- Early identification of individuals at risk of mental illness
- Improved access to appropriate mental health services
- Increased awareness and understanding of mental health
- Enhanced well-being and productivity for working professionals
- Personalized risk assessment and recommendations

Disadvantages:

- Potential for misdiagnosis and false positives
- Reliance on self-reported data, which can be inaccurate
- Privacy concerns regarding data collection and usage

- Ethical considerations surrounding machine learning algorithms
- Limited availability of mental health resources and support

11. Conclusion

- Summarize the key findings of the project, including the performance of the system and its potential impact on mental health well-being.

  Discuss the limitations of the project and areas for future improvement.

12. Future Scope

- Integrate with additional data sources, such as health sensors and wearables.
- Develop more sophisticated machine learning models for improved accuracy.
- Implement personalized interventions and support programs.
- Conduct further research on the ethical implications of using machine learning for mental health prediction.
- Expand the platform to other populations beyond working professionals.