**Title:**

**AI-Powered Mental Health Monitoring and Support System Using Django and Machine Learning**

**Introduction:**

Mental health has become one of the most critical aspects of overall well-being, yet many individuals struggle to seek help at the right time.  
This project introduces an AI-driven platform that helps users monitor their mental health through simple mood assessments and offers personalized recommendations.  
By leveraging machine learning models and a role-based Django backend, the system bridges the gap between patients, guardians, and healthcare providers, facilitating early intervention and continuous support.

# Abstract:

### Information about the area:

Mental health care is increasingly recognized as a vital part of public health. With the rise of mental disorders globally, early diagnosis and support are essential.

### Current trend:

Technology-driven mental health solutions are gaining popularity, including chatbots, AI-based diagnosis tools, mood trackers, and online counseling platforms.

### Issue or gap identified:

Many existing mental health platforms either lack real-time predictions, integration with human caregivers (guardians/providers), or personalization based on mood states.  
Additionally, access to immediate support after identifying mental health risks is often limited.

### What you are going to propose:

This project proposes a web-based platform where users can regularly submit their mood feedback (via emojis), receive real-time mental health assessments, and get personalized recommendations, including emergency support if needed.  
Guardians and healthcare providers can monitor patients, assign activities or medications, and stay informed through a messaging system.

### Technology being used in this project:

* **Backend**: Python, Django, SQLite
* **Machine Learning**: Random Forest, Decision Tree, XGBoost
* **Frontend**: Bootstrap CSS, HTML, JavaScript
* **Authentication**: Django’s Custom User Model
* **Storage**: Django ORM, SQLite

### Output:

* Predict user's mental health condition based on a simple emoji input.
* Store and track mood assessments over time.
* Allow guardians and healthcare providers to intervene early.
* Suggest wellness activities, medication, or emergency help.

# Literature Survey:

| **Title** | **Author** | **Description** | **Advantages** | **Disadvantages** | **Future Scope** | **URL** | **Reference** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Machine Learning for Mental Health | H. Tran et al. | Discusses ML approaches for predicting depression and anxiety. | Real-time prediction | Needs large datasets | Combining multiple data sources | <https://arxiv.org/abs/1806.11322> | [1] |
| AI in Mental Health Diagnosis | P. Thangaraj et al. | Application of AI tools in clinical psychology. | Better diagnosis | Ethical concerns | More personalized models | <https://ieeexplore.ieee.org/document/9186170> | [2] |
| Mental Health Apps and Effectiveness | Nicholas et al. | Review on mental health apps' impact on user well-being. | Easily accessible | Lack of clinical validation | Integration with therapy | <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6520077/> | [3] |
| Predictive Models for Stress Detection | B. Ghosh et al. | Focuses on stress detection using data mining. | Accurate models | Data privacy issues | Real-time monitoring | <https://ieeexplore.ieee.org/document/8692504> | [4] |
| Mobile Apps for Mental Health | F. Firth et al. | Review of mobile interventions for mental disorders. | Easy deployment | Engagement drop-off | Better engagement strategies | https://journals.sagepub.com/doi/10.1177/0706743717745949 | [5] |
| Deep Learning for Depression Analysis | Y. Liu et al. | Deep models for detecting depression from social media. | High accuracy | Requires lots of labeled data | More explainable models | <https://arxiv.org/abs/1904.03275> | [6] |
| Chatbots for Mental Health | R. Inkster et al. | Effectiveness of mental health chatbots. | 24/7 availability | Lack of empathy | Hybrid human-bot models | https://journals.sagepub.com/doi/10.1177/1359105319878267 | [7] |
| Data-Driven Mental Health Support | D. McKinnon et al. | Using data science to aid mental health services. | Evidence-based insights | Needs robust data pipelines | Real-time adaptation | https://www.frontiersin.org/articles/10.3389/fpsyt.2021.567427/full | [8] |
| Mental Health and AI Ethics | M. Mittelstadt | Examines ethical risks of AI in healthcare. | Highlights important risks | Regulatory challenges | Fair AI frameworks | https://academic.oup.com/jlb/article/7/1/lsaa056/6016396 | [9] |
| ML Algorithms for Mental State Prediction | L. Stojanovic et al. | Surveys ML models for predicting emotional states. | Comparative analysis | Performance varies by dataset | Multi-modal prediction | <https://ieeexplore.ieee.org/document/8803313> | [10] |
| Using Mobile Sensing for Mental Health | T. Wang et al. | Mobile sensor data used for mental health prediction. | Passive data collection | Battery drain concerns | Efficient sensing techniques | <https://www.nature.com/articles/s41591-018-0227-8> | [11] |
| Internet Interventions for Depression | Andersson et al. | Clinical trials of online therapy systems. | Clinical validation | Low adoption | Combine with offline support | https://journals.sagepub.com/doi/full/10.1177/0706743714528835 | [12] |
| Machine Learning Predicts PTSD | Karstoft et al. | ML techniques to predict PTSD risk factors. | High predictive accuracy | Ethical challenges | Risk-aware intervention tools | https://academic.oup.com/sleep/article/39/12/2231/2662788 | [13] |
| Online Screening for Mental Health | Ebert et al. | Study on online mental health screening platforms. | Wider outreach | Digital divide | Gamification to improve participation | https://academic.oup.com/jamia/article/25/10/1357/5098525 | [14] |
| Future of AI in Mental Health | M. Fiske et al. | The future landscape of AI-based mental healthcare. | Strong research potential | Systematic biases | Bias mitigation strategies | https://www.frontiersin.org/articles/10.3389/fpsyg.2019.01206/full | [15] |

**Gaps Identified:**

* Lack of real-time prediction in many existing systems.
* Weak integration between prediction and immediate caregiver intervention.
* Limited personalization for different user types (patients, guardians, healthcare providers).
* Poor availability of emergency support based on user mood status.

# System Requirement Specification:

## Hardware Requirements:

* **Processor**: Intel i5/i7 or equivalent
* **RAM**: 8 GB minimum
* **Storage**: 256 GB SSD (or 1 TB HDD)
* **Internet Connection**: Stable for web hosting

## Software Requirements:

* **Operating System**: Windows 10/11, Ubuntu 20.04+
* **Backend**: Python 3.10+, Django 4.x
* **Database**: SQLite (development
* **Frontend**: HTML5, Bootstrap 5, JavaScript
* **Other Libraries**: Scikit-learn, XGBoost, Joblib

**Problem Statement:**

* There is a critical need for a technology-driven mental health platform that not only assesses the user's mental state in real-time but also connects them immediately with guardians or healthcare providers for timely support and recommendations.

**Objectives:**

* To develop a web-based platform for mental health monitoring using machine learning.
* To allow different roles (patient, guardian, healthcare provider) for better personalized care.
* To predict mental health condition using simple emoji-based input.
* To provide instant recommendations, suggestions, and emergency contact information.
* To enable guardians to assign activities, medications, and recommendations easily.
* To maintain user records securely and ensure data privacy.

**Key Components:**

* **User Authentication System**: Role-based login (Patient, Guardian, Healthcare Provider).
* **Mood Tracking Module**: Input system via emoji selection.
* **Prediction Engine**: Machine Learning model to predict mental health status.
* **Recommendation System**: Provides mental health advice, activities, or emergency contacts.
* **Guardian/Provider Dashboard**: Monitor patient's condition, assign medications or activities.
* **Messaging/Alert System**: Alert guardians/providers in case of poor mental health prediction.
* **Data Storage Module**: SQLite database to store assessments, activities, medications.

# Functional and Non-Functional Requirements:

### Functional Requirements:

* User registration/login with role specification.
* Mood input by patients.
* Real-time mental health prediction.
* Assignment of tasks/medications by guardians or providers.
* Notifications for critical mental health conditions.
* Historical record tracking for patients.

### Non-Functional Requirements:

* System should respond within 2-3 seconds for prediction.
* Data must be securely stored and encrypted.
* System should be scalable for larger user bases.
* Should be mobile and desktop accessible (responsive design).
* High system availability (uptime > 99%).

**Models:**

* **User Model**: Custom Django User Model with role fields.
* **Mood Assessment Model**: Stores daily mood inputs and prediction results.
* **Recommendation Model**: Stores activity/medication assignments.
* **Machine Learning Models**:
  + Random Forest Classifier
  + Decision Tree Classifier
  + XGBoost Classifier

**Proposed Methodology:**

1. Users submit mood via an emoji-based interface.
2. Mood input is fed into the trained ML model.
3. ML model predicts user's mental health condition (e.g., stable, mild depression, high-risk).
4. Depending on prediction:
   * Provide advice, activities, or urgent alerts.
   * Notify guardian or healthcare provider if needed.
5. Track mood over time for continuous assessment.
6. Allow guardians and providers to log and assign wellness tasks/medications.

**Existing System:**

* Existing mental health apps (e.g., Woebot, Wysa) mainly rely on chatbot interactions.
* Traditional therapy sessions and mental health tracking apps require manual input without real-time ML-based diagnosis.

**Disadvantages of Existing System:**

* Lack of instant machine-learning-based prediction.
* No real-time caregiver involvement.
* Limited personalization based on the patient's history.
* Lack of emergency escalation support.

# Proposed System and Advantages:

### Proposed System:

An AI-driven, web-based system using Django and ML models for real-time mood analysis, tracking, and guardian/provider involvement.

### Advantages:

* Real-time mental health condition prediction.
* Immediate guardian/provider alerts.
* Personalized health advice and activities.
* Centralized dashboard for easy monitoring.

**Algorithms Used:**

* **Random Forest Classifier**: Ensemble model for robust and accurate prediction.
* **Decision Tree Classifier**: Simple, explainable model for quick testing.
* **XGBoost Classifier**: Boosted trees for high-accuracy performance.
* **Support Libraries**: Scikit-learn, XGBoost Python library, Pandas, Numpy.

**Expected Results:**

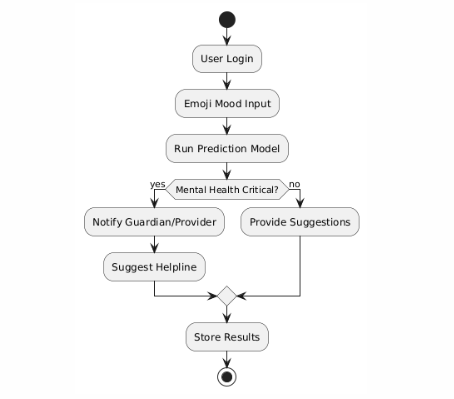
* Users will be able to quickly assess their mental health with minimal input.
* Guardians and healthcare providers will get timely notifications for intervention.
* Overall reduction in the time between mental health issue detection and support.
* The system will store and analyze mood patterns over time for long-term insights.
* Achieve prediction accuracy greater than 85% using ensemble ML models.

**Conclusion:**

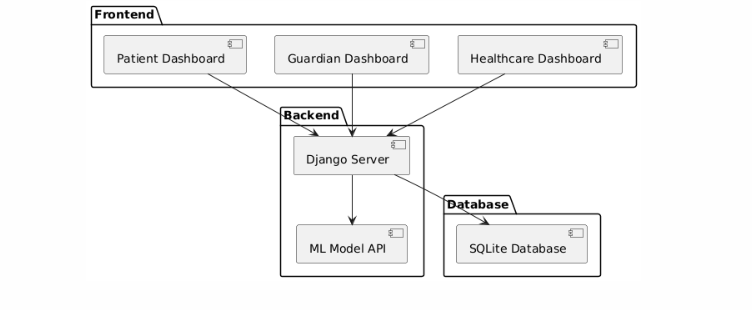
* Mental health issues require immediate and personalized attention, yet many existing platforms fail to provide real-time predictions and interventions.  
  This project offers a novel solution by combining Machine Learning and web technologies to enable patients, guardians, and healthcare providers to work collaboratively.  
  Through early detection and personalized support, the proposed system aims to improve user well-being, lower the burden on healthcare systems, and foster a healthier society.

# DIAGRAMS:

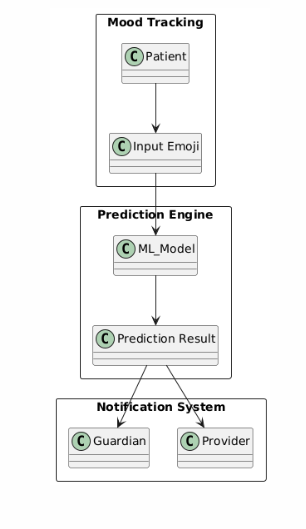
### 1. Flowchart Diagram:



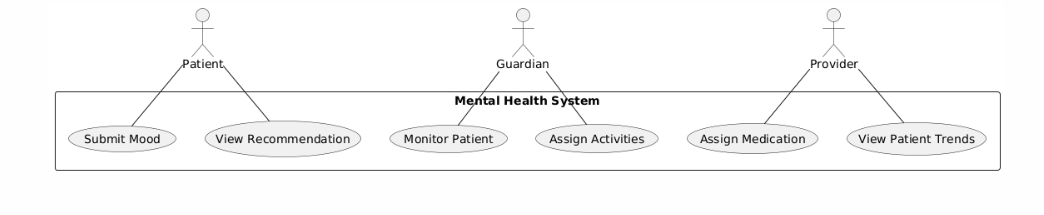
Architecture Diagram



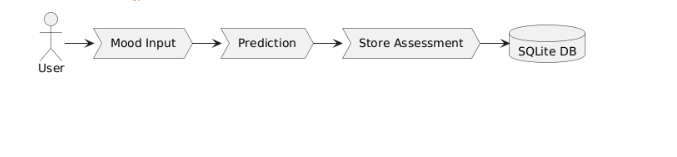
Design Diagram



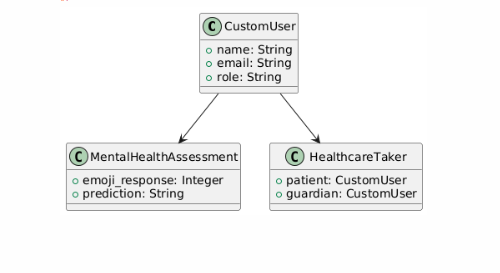
Use Case Diagram



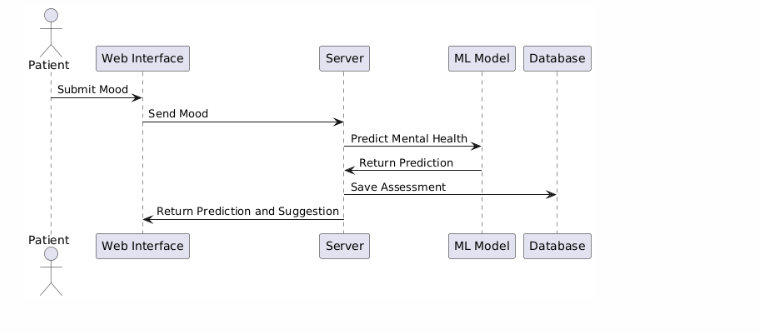
Data Flow Diagram



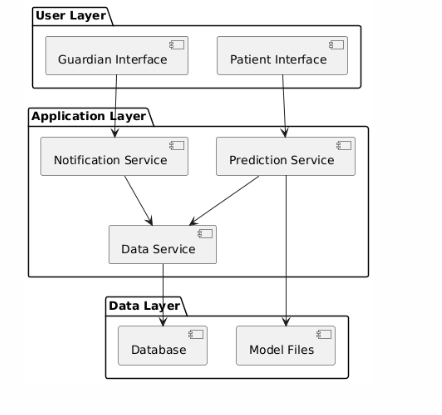
Class Diagram



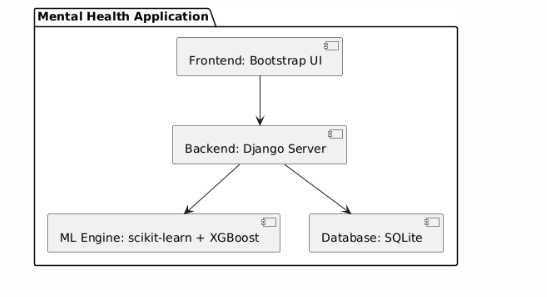
Sequence Diagram



Block Diagram



Component Diagram



| **Diagram Type** | **Explanation** |
| --- | --- |
| Flowchart | Shows the overall system flow from user input to prediction to storage. |
| Architecture Diagram | Depicts the frontend, backend, and database setup of the system. |
| Design Diagram | High-level overview of key modules and their interactions. |
| Use Case Diagram | Shows user interactions with the system (patients, guardians, providers). |
| Data Flow Diagram | Represents how data moves through the system. |
| Class Diagram | Describes major entities (models) and their relationships. |
| Sequence Diagram | Shows the timeline of operations from mood input to result delivery. |
| Block Diagram | Layers the system into User, Application, and Data layers. |
| Component Diagram | Shows the major software components and their connections. |