

# Predicting the Habitability of Exoplanets Using Machine Learning

## Project Statement:

The discovery of exoplanets has accelerated in recent years, but identifying which planets could potentially support life remains a challenge. Current astronomical methods are limited by observational constraints, massive datasets, and complex planetary parameters.

This project, ExoHabitAI, aims to use machine learning to predict the habitability of exoplanets based on physical, orbital, and stellar features. The system evaluates exoplanetary characteristics such as radius, mass, orbital period, equilibrium temperature, and host star parameters to classify planets as potentially habitable or not. By leveraging data-driven models, astronomers and researchers can prioritize candidate exoplanets for further study, enhancing the efficiency of observational campaigns.

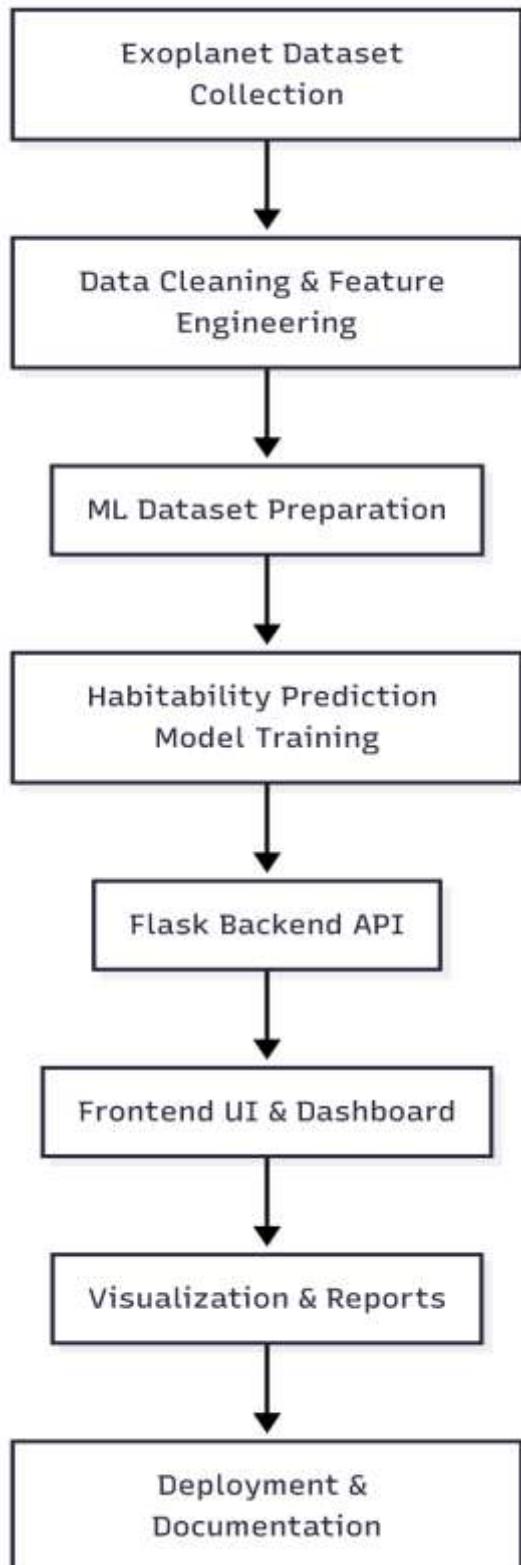
## Outcomes:

- Develop a deployable machine learning model to classify exoplanet habitability.
- Analyze exoplanet features to identify key factors influencing habitability.
- Predict habitability scores and rank planets for further observational study.
- Enable visual insights and exploratory data analysis through plots and dashboards.
- Provide a modular architecture and documentation for future model expansion.

## Modules to be implemented

- Data Collection & Management (Exoplanet Dataset)
- Data Cleaning & Feature Engineering
- Machine Learning Dataset Preparation
- AI Model for Habitability Prediction (ML-Based)
- Backend API Integration (Flask)
- Frontend UI (Bootstrap + HTML + JavaScript)
- Visualization & Dashboard (Habitability Insights)
- Deployment & Documentation

## Workflow for Predicting Exoplanet Habitability Using Machine Learning



## **Week-wise Module Implementation and high-level requirements with output Screenshots**

### **Milestone 1: Week 1-2**

#### **Module 1: Data Collection and Management**

- Collect exoplanet datasets from sources like NASA Exoplanet Archive and Kaggle.
- Dataset features may include:
  - Planet radius, mass, density, surface temperature, orbital period, distance from star
  - Host star type, luminosity, temperature, metallicity
- Store data in PostgreSQL or CSV format.
- Validate schema and ensure completeness.

#### **Module 2: Data Cleaning and Feature Engineering**

- Handle missing values, outliers, and inconsistent entries.
- Encode categorical features (e.g., star type) using one-hot encoding.
- Feature engineering examples:
  - Habitability Score Index based on key planetary parameters.
  - Stellar Compatibility Index to measure host star influence.
  - Normalize numerical features.
- Validate data quality using descriptive statistics and visualization.

### **Milestone 2: Week 3-4**

#### **Module 3: Machine Learning Dataset Preparation**

- Split data into training and testing sets (e.g., 80:20).
- Select features for model input based on correlation with habitability.
- Define target variable: Habitability class (Habitable/Non-Habitable) or habitability score.
- Create data pipelines with scaling, encoding, and feature selection.

#### **Module 4: AI Model for Habitability Prediction**

- Train models such as:
  - Random Forest Classifier for binary habitability prediction
  - XGBoost Classifier for multi-class habitability levels
  - Optionally, Logistic Regression or SVM for comparison
- Evaluate models using:
  - Accuracy, Precision, Recall, F1-score, ROC-AUC
- Rank exoplanets based on predicted habitability scores.

## Milestone 3 : Week 5-6

### Module 5: Flask Backend API

- Create REST APIs to:
- Accept exoplanet data input from users
- Return habitability prediction and ranking
- Connect Flask backend to the database for dynamic data retrieval
- Secure endpoints and implement JSON response structure

### Module 6: Frontend UI Development

- Build a clean, responsive interface using HTML, CSS, and Bootstrap
- Input forms for planetary parameters
- Display predictions, habitability scores, and ranking tables
- Visualize key features affecting habitability

## Milestone 4: Week 7-8

### Module 7: Visualization & Dashboard

- Build dashboard for visual insights:
- Feature importance plots
- Habitability score distribution
- Star-planet parameter correlations
- Use matplotlib, seaborn, or plotly for interactive charts
- Export reports (PDF/Excel) for top candidate exoplanets

### Module 8: Deployment & Documentation

- Deploy application on Heroku or Render with connected database
- Provide full technical documentation and README
- Include project report and video demo

## **Evaluation Criteria:**

### **Milestone 1 Evaluation (Week 1-2):**

- Successful collection and organization of exoplanet datasets
- Proper setup of database or CSV schema
- Completion of basic data cleaning and feature engineering
- Generation of initial habitability-related metrics
- Initial documentation of data dictionary

### **Milestone 2 Evaluation (Week 3-4):**

- Preparation of training and testing datasets
- Implementation of ML pipelines (scaling, encoding)
- Training and evaluation of ML models (Random Forest/XGBoost)
- Initial habitability prediction accuracy and model comparison
- Ranking of exoplanets based on model outputs

### **Milestone 3 Evaluation (Week 5-6):**

- Backend API development and ML integration
- REST API endpoints working for data input and prediction
- JSON response structure correctly implemented
- Initial frontend integration

### **Milestone 4 Evaluation (Week 7–8):**

- Development of interactive dashboard with visualizations
- Export of habitability insights and reports
- Full integration of frontend, backend, and ML model
- Successful deployment of application with documentation