# Project Report: Exoplanet Habitability Predictor

---

# 1. Project Overview

The **Exoplanet Habitability Predictor** aims to use machine learning models to predict the potential habitability of exoplanets based on planetary and stellar data. By leveraging data from NASA's Exoplanet Archive, the project focuses on identifying exoplanets that could potentially harbor life. The project is structured as an interactive web application, where users can input parameters of exoplanets and get predictions in real-time.

---

# 2. Project Objectives

1. **Primary Objective:** Develop a machine learning-based tool that can predict the habitability of exoplanets based on key planetary characteristics like radius, orbital distance, and temperature.
2. **Secondary Objectives:**
    - Provide an interactive web interface for users to explore and predict exoplanet habitability.
    - Visualize important planetary features through data visualizations.
    - Educate users about what makes a planet potentially habitable.

---

# 3. Data Sources

The datasets for this project were obtained from NASA’s Exoplanet Archive, which includes detailed information about known exoplanets and their host stars. The key datasets used in the project are:

1. **PlanetarySystems_data.csv**: Contains detailed data about exoplanetary systems.
2. **PlanetarySystemsComposite_data.csv**: Combines various features of planets, such as mass, radius, and distance from their star.
3. **StellarHosts_data.csv**: Contains information about the stars hosting these exoplanets.
4. **TransitingPlanetsTable_data.csv**: Provides data on transiting planets, including their orbital characteristics.

These datasets were cleaned, combined, and preprocessed to create a final dataset for training machine learning models.

---

# 4. Project Functionality

The **Exoplanet Habitability Predictor** app offers several key functionalities:

## 4.1 Real-Time Habitability Prediction

The core functionality of the app is predicting exoplanet habitability based on user-input planetary features such as:

- **Planetary radius**
- **Orbital distance from the host star**
- **Surface temperature**
- **Mass of the exoplanet**
- **Stellar characteristics of the host star**

Users can input these features into the app, and it returns a habitability score along with a classification (habitable or non-habitable).

## 4.2 User-Friendly Web Interface

The app is built using **Streamlit**, allowing users to interact easily with the machine learning model via a simple and intuitive web interface.

## 4.3 Data Visualizations

Key data visualizations are embedded into the app to help users understand the distribution of planetary features and their correlation with habitability. These include:

- Temperature vs. habitability plots
- Radius vs. orbital distance scatter plots
- Host star temperature distributions

## 4.4 Machine Learning Integration

The machine learning model, built using **Scikit-learn**, integrates a **Random Forest Classifier**. The model provides the following functionalities:

- **Training** on the NASA exoplanet dataset with features like radius, orbital distance, and surface temperature.
- **Real-time predictions** based on user inputs, enabling users to explore various planetary combinations.
- **Evaluation Metrics** including accuracy, precision, recall, and F1-score, ensuring reliable predictions.

---

# 5. Technologies Used

The project utilizes a wide range of technologies for data processing, machine learning, and web development:

1. **Python**: The primary programming language for the entire project.
2. **Pandas** and **Numpy**: Used for data cleaning, manipulation, and analysis.
3. **Scikit-learn**: Employed for machine learning tasks like model training, prediction, and evaluation.
4. **Matplotlib** and **Seaborn**: Libraries used for generating data visualizations.
5. **Streamlit**: Framework used to build the interactive web interface.
6. **Git and GitHub**: For version control and collaboration.

---

# 6. Methodology

## 6.1 Data Cleaning and Preprocessing

- **Data Cleaning**: Handle missing values, remove irrelevant columns, and correct inconsistencies in the dataset.
- **Feature Selection**: Select key planetary and stellar features such as mass, radius, and distance to predict habitability.
- **Data Normalization**: Normalize the features for better performance of machine learning models.

## 6.2 Machine Learning Model Development

- **Model Selection**: Several models were tested, including Logistic Regression, Support Vector Machines (SVM), and Random Forest. The **Random Forest Classifier** was selected for its high accuracy and robustness.
- **Model Training**: The model was trained using 70% of the dataset and validated using the remaining 30%.
- **Performance Evaluation**: Accuracy, precision, recall, and F1-score were used to assess the performance.

## 6.3 Web Application Development

- **Frontend**: The user interface was developed using Streamlit, which allows for real-time interaction with the machine learning model.
- **Backend**: The backend handles model inference and data processing, providing real-time responses to the user’s input.

---

# 7. Results and Performance

## 7.1 Model Performance

The Random Forest Classifier achieved the following results on the test dataset:

- **Accuracy**: 85%
- **Precision**: 0.83
- **Recall**: 0.86
- **F1-Score**: 0.84

## 7.2 App Performance

- **Real-Time Predictions**: The app responds within seconds to user inputs, providing habitability predictions.
- **Visualization**: The app's visualizations help users understand planetary features and their role in habitability.

---

# 8. Challenges Faced

1. **Handling Missing Data**: Some datasets had missing values, which required imputation techniques and careful feature selection.
2. **Model Selection**: Choosing the right machine learning model involved testing multiple algorithms and tuning hyperparameters.
3. **Real-Time Response Optimization**: Optimizing the app to provide quick and accurate predictions without long loading times.

---

# 9. Future Enhancements

## 9.1 Improved Prediction Models

- Future iterations of the project could implement more advanced machine learning techniques, such as neural networks, to improve the prediction accuracy.

## 9.2 Expansion of Features

- Adding more planetary and stellar features, such as atmospheric composition or gravitational pull, could enhance the prediction accuracy.

### 9.3 Collaboration with NASA

- The project could be extended to integrate more datasets from ongoing NASA missions like TESS or the James Webb Space Telescope, providing richer data for habitability analysis.

---

# 10. Conclusion

The **Exoplanet Habitability Predictor** successfully integrates data science, machine learning, and web technologies to offer real-time habitability predictions for exoplanets. With a user-friendly interface and reliable model, it serves as an educational tool as well as a research aid for studying exoplanetary systems. The project demonstrates how machine learning can be applied to astronomical data to draw meaningful insights and contribute to the broader search for life beyond Earth.

---

## Appendices

### A. Machine Learning Model Details

- **Random Forest Classifier**:
  - Hyper parameters: Number of estimators = 100, max depth = 10
  - Performance metrics: Accuracy = 85%, Precision = 0.83, Recall = 0.86, F1-Score = 0.84

### B. Datasets Used

1. **PlanetarySystems_data.csv**
2. **PlanetarySystemsComposite_data.csv**
3. **StellarHosts_data.csv**
4. **TransitingPlanetsTable_data.csv**