

# **Milestone 3 : ExoHabitAI – Backend & Frontent Documentation**

## **1. Introduction**

**The backend of ExoHabitAI is responsible for exposing the trained Machine Learning model through RESTful APIs. It allows users or frontend applications to send exoplanet parameters and receive:**

- **Habitability prediction (Habitable / Non-Habitable)**
- **Confidence score (probability)**
- **Ranking of multiple exoplanets based on habitability**

**The backend is implemented using Python Flask and follows a clean, modular structure.**

**ExoHabitAI is a Machine Learning–based system designed to analyze exoplanet data and predict their potential habitability.**

## **2. Technologies Used**

- **Python 3.x** – Core programming language
- **Flask** – Web framework for building REST APIs
- **NumPy** – Numerical computations and array handling
- **Joblib** – Loading trained ML model
- **Scikit-learn (Pipeline)** – Preprocessing and prediction (used inside the model)

## **3. Requirements.txt**

**The requirements.txt file lists all dependencies required to run the backend.**

**File: requirements.txt**

**flask  
numpy  
joblib  
scikit-learn  
pandas**

### Purpose:

- Ensures consistent environment setup
- Simplifies installation using:

```
python -m pip install -r requirements.txt
```

## 4. utils.py

### Purpose:

**utils.py contains reusable helper functions that:**

- Validate incoming API data
- Perform feature engineering
- Encode categorical features (star types)
- Prepare model-ready input
- Format prediction output

This separation improves readability, maintainability, and follows industry best practices.

### 4.1 Feature Order

The model was trained on **23 features**, and the same order must be preserved during prediction.

```
FEATURE_ORDER = [  
    'pl_rade', 'pl_bmasse', 'pl_orbper', 'pl_orbsmax', 'pl_eqt',  
    'st_teff', 'st_lum', 'st_met', 'pl_insol',  
    'Habitability_Score', 'Steller_Compatibility', 'Orbital_Stability_Score',  
    'Star_Type_A', 'Star_Type_B', 'Star_Type_D', 'Star_Type_F',  
    'Star_Type_G', 'Star_Type_K', 'Star_Type_L', 'Star_Type_M',  
    'Star_Type_T', 'Star_Type_W', 'Stellar_Compatibility'  
]
```

### 4.2 validate\_input()

**Validates input JSON and converts it into a NumPy array suitable for the ML pipeline.**

**Functions performed:**

- Checks for missing required fields
- Converts values to numeric form
- Computes engineered features
- Performs one-hot encoding for star types

## 4.3 Feature Engineering

Derived features such as:

- Habitability Score
- Orbital Stability Score
- Stellar Compatibility

are recomputed in the backend to match the training pipeline.

## 4.4 format\_prediction()

Formats the model output into a clean JSON response.

**Output:** Prediction label & Confidence score

# 5. app.py

**Purpose:**

app.py is the main entry point of the backend application. It:

- Initializes the Flask app
- Loads the trained ML model
- Defines REST API endpoints
- Handles request-response lifecycle

# 6. API Endpoints

## 6.1 Home Endpoint

**URL:** /

**Method:** GET

**Purpose:** Check if backend is running

**Response:**

```
{  
  "message": "ExoHabitAI Backend API is running",  
  "status": "success"  
}
```

## 6.2 Predict Endpoint

**URL:** /predict

**Method:** POST

**Input (JSON):**

```
{  
  "pl_rade": 1.1,  
  "pl_bmasse": 1.2,  
  "pl_orbper": 365,  
  "pl_orbsmax": 1.0,  
  "pl_eqt": 288,  
  "st_teff": 5778,  
  "st_lum": 1.0,  
  "st_met": 0.0,  
  "pl_insol": 1.0,  
  "star_type": "G"}
```

**Output:**{

```
  "status": "success",  
  "result": {  
    "prediction": "Habitable",  
    "confidence_score": 0.82}}
```

## 6.3 Rank Endpoint

**URL:** /rank

**Method:** POST

**Purpose:** Rank multiple exoplanets based on habitability score

**Input:** List of exoplanet objects

**Output:**{

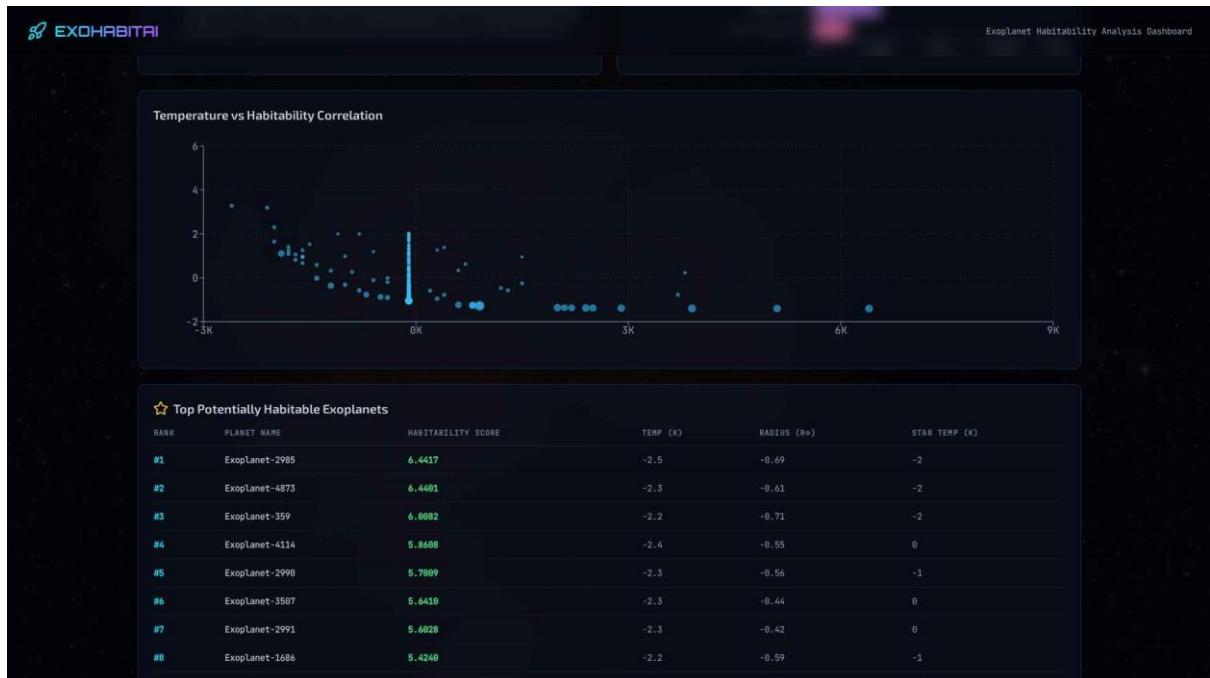
```
"status": "success",
"ranked_exoplanets": [
{
  "planet_name": "Planet-A",
  "prediction": "Habitable",
  "habitability_score": 0.82,
  "rank": 1},
{
  "planet_name": "Planet-B",
  "prediction": "Non-Habitable",
  "habitability_score": 0.54,
  "rank": 2
}]}{
```

## 7. Frontend

### 7.1 Frontend Files

- index.html
- index.css
- GlassCars.js
- MetricCard.js
- HabitabilityDistribution.js
- Featureimportance.js
- Scatterplot.js
- TopOPlanetstable.js
- PredictionPanel.js
- App.js
- App.css

## 7.2 Frontend ScreenShots



**EXOHABITAI**

Exoplanet Habitability Analysis Dashboard

RANK	PLANET NAME	HABITABILITY SCORE	TEMP (K)	RADIUS (R <sub>⊕</sub> )	STAR TEMP (K)
#1	Exoplanet-2985	<b>6.4417</b>	-2.5	-0.69	-2
#2	Exoplanet-4873	<b>6.4401</b>	-2.3	-0.61	-2
#3	Exoplanet-359	<b>6.0082</b>	-2.2	-0.71	-2
#4	Exoplanet-4114	<b>5.8688</b>	-2.4	-0.55	0
#5	Exoplanet-2998	<b>5.7089</b>	-2.3	-0.56	-1
#6	Exoplanet-3507	<b>5.6410</b>	-2.3	-0.44	0
#7	Exoplanet-2991	<b>5.6028</b>	-2.3	-0.42	0
#8	Exoplanet-1686	<b>5.4240</b>	-2.2	-0.59	-1
#9	Exoplanet-2984	<b>5.4224</b>	-2.2	-0.7	-2
#10	Exoplanet-862	<b>5.3521</b>	-2.3	-0.36	0
#11	Exoplanet-4856	<b>5.1631</b>	-2.3	-0.36	-1
#12	Exoplanet-5774	<b>5.1360</b>	-2.3	-0.36	-2
#13	Exoplanet-5714	<b>5.1358</b>	-2.5	-0.57	-2
#14	Exoplanet-5366	<b>5.1203</b>	-2.2	-0.62	-1
#15	Exoplanet-1027	<b>4.5563</b>	-2.4	-0.36	0

**⚡ Habitability Prediction Tool**

Exoplanet Habitability Analysis Dashboard

Planet Name Test-Earth-442b	Radius (Earth radii) 1.1	Mass (Earth masses) 1.3
Orbital Period (days) 385	Semi-Major Axis (AU) 1.05	Equilibrium Temp (K) 285
Stellar Temp (K) 5700	Stellar Luminosity (Solar) 0.95	Stellar Metallicity 0.02
Insolation Flux 0.98	Star Type G	

**EXOHABITAI**

Exoplanet Habitability Analysis Dashboard

⭐ Top Potentially Habitable Exoplanets					
RANK	PLANET NAME	HABITABILITY SCORE	TEMP (K)	RADIUS (R <sub>⊕</sub> )	STAR TEMP (K)
#1	Exoplanet-2985	<b>6.4417</b>	-2.5	-0.69	-2
#2	Exoplanet-4873	<b>6.4401</b>	-2.3	-0.61	-2
#3	Exoplanet-359	<b>6.0082</b>	-2.2	-0.71	-2
#4	Exoplanet-4114	<b>5.8688</b>	-2.4	-0.55	0
#5	Exoplanet-2998	<b>5.7089</b>	-2.3	-0.56	-1
#6	Exoplanet-3507	<b>5.6410</b>	-2.3	-0.44	0
#7	Exoplanet-2991	<b>5.6028</b>	-2.3	-0.42	0
#8	Exoplanet-1686	<b>5.4240</b>	-2.2	-0.59	-1
#9	Exoplanet-2984	<b>5.4224</b>	-2.2	-0.7	-2
#10	Exoplanet-862	<b>5.3521</b>	-2.3	-0.36	0
#11	Exoplanet-4856	<b>5.1631</b>	-2.3	-0.36	-1
#12	Exoplanet-5774	<b>5.1360</b>	-2.3	-0.36	-2
#13	Exoplanet-5714	<b>5.1358</b>	-2.5	-0.57	-2
#14	Exoplanet-5366	<b>5.1203</b>	-2.2	-0.62	-1
#15	Exoplanet-1027	<b>4.5563</b>	-2.4	-0.36	0

**⚡ Habitability Prediction Tool**

Planet Name e.g., Kepler-442b	Radius (Earth radii) 1.2	Mass (Earth masses) 1.5
Orbital Period (days)	Semi-Major Axis (AU)	Equilibrium Temp (K)

## 8. Conclusion

The ExoHabitAI Backend and Brontend successfully provides an intuitive, informative, and interactive interface for exoplanet habitability analysis.

It enhances the usability of the machine learning model and enables users to understand both predictions and underlying data patterns effectively.

