1. **Introduction**
   * **Purpose:** The purpose of this document is to provide a comprehensive overview of the requirements for the development of the "Habitable Exoplanet Explorer" software system.
   * **Scope:** The scope of this project includes the development of a software application that enables users to explore exoplanets within the habitable zone and analyze their characteristics. The application will gather data from sources such as NASA's Kepler and TESS missions, preprocess the data, perform analysis, and visualize the results.
2. **Functional Requirements**
   * **Data Acquisition**
     + **Description:** Gather exoplanet data from sources like NASA's Kepler, TESS, or other relevant space missions.
     + **Acceptance Criteria:** Data is successfully retrieved and stored in a structured format.
   * **Data Preprocessing**
     + **Description:** Clean and preprocess the collected data, handling missing values, outliers, and ensuring consistency.
     + **Acceptance Criteria:** A clean dataset ready for analysis is generated.
   * **Data Analysis**
     + **Description:** Analyze and visualize the processed data, discovering patterns and relationships.
     + **Acceptance Criteria:** Visual analysis of data reveals relevant insights.
3. **Non-Functional Requirements**
   * **Performance:** The system should be able to handle large volumes of data efficiently, providing timely responses to user queries.
   * **Usability:** The user interface should be intuitive and user-friendly, with clear navigation and informative visualizations.
   * **Reliability**:The system should be reliable and available for use at all times, with minimal downtime for maintenance or updates.
4. **External Interfaces**
   * **User Interfaces:** The system will have a web-based user interface accessible through standard web browsers.
   * **Software Interfaces:**  The system may interact with external APIs provided by space agencies for data retrieval and analysis.
5. **System Features**
   * **Feature 1: Stellar Classification**
     + **Description:** The system may classify host stars based on their spectral type, luminosity, and other parameters.
   * **Feature 2: Habitable Zone Calculation**
     + **Description:** The system should calculate the habitable zone for each star in the dataset based on stellar properties.
   * **Feature 3: Exoplanet Characterization**
     + **Description:** The system should enhance exoplanet characterization algorithms to estimate key parameters accurately.
   * **Feature 3: Potential Habitable Exoplanet Candidates**
     + **Description:** The system will yield the best candidates for habitability out of all the exoplanets. Allowing further research and use available resources to focus on the candidates.
6. **Database Requirements**
   * **Logical Database Requirements:**
     + The system will require a relational database to store exoplanet data, including properties such as planetary mass, radius, and orbital period.
     + Database tables may include "Exoplanets," "Stellar Data," "Atmospheric Conditions," etc.
7. **Constraints**
   * **Technical Constraints:**
     + The system must adhere to technical constraints such as compatibility with web standards, scalability, and security requirements.
   * **Regulatory Constraints:**
     + The system must comply with relevant laws, regulations, and ethical guidelines regarding the collection and use of exoplanet data.
   * **Budgetary Constraints:**
     + The project budget will be limited, requiring efficient resource allocation and cost-effective solutions.
8. **Assumptions and Dependencies**
   * **Assumptions:**
     + The availability of reliable exoplanet data from sources such as NASA's Kepler and TESS missions.
   * **Dependencies:**
     + The project may depend on access to external APIs and services provided by space agencies for data retrieval and analysis.
9. **Glossary**

* **Exoplanet:** A planet that orbits a star outside the solar system.
* **Spectral Type:** Classification of stars based on their spectral characteristics.
* **Luminosity:** The total amount of energy emitted by a star per unit time.
* **Orbital Period:** The time it takes for a planet to complete one orbit around its star.
* **Atmospheric Conditions:** The composition and properties of a planet's atmosphere, including temperature, pressure, and chemical composition.
* **Exoplanet Catalog:** A database or collection of exoplanet data, including properties such as mass, radius, orbital period, and habitability indices.
* **Spectral Analysis:** The study of electromagnetic spectra emitted by stars and exoplanets to determine their composition, temperature, and other physical characteristics.
* **Habitable Zone (Goldilocks Zone):** The region around a star where conditions are suitable for the existence of liquid water on the surface of a planet, potentially supporting life as we know it.
* **Bayesian Inference:** A statistical method for updating beliefs or probabilities based on new evidence or data, commonly used in parameter estimation and hypothesis testing.
* **MCMC (Markov Chain Monte Carlo) Methods:** Computational algorithms for sampling from probability distributions, often used in Bayesian inference and parameter estimation tasks.
* **Atmospheric Modeling:** The process of simulating and analyzing the atmospheric conditions of exoplanets, including temperature, pressure, and composition, to assess their potential habitability.
* **Climate Modeling:** The use of mathematical and computational models to simulate and predict climate patterns and changes on exoplanets, taking into account factors such as atmospheric dynamics, greenhouse gases, and solar radiation.
* **Habitability Index:** A metric or scale used to assess the potential habitability of exoplanets based on various factors such as temperature, surface conditions, and atmospheric composition.
* **Stellar Properties:** Characteristics of stars such as spectral type, luminosity, temperature, and age, which influence the habitable zones and potential habitability of their orbiting exoplanets.

**User-Side Use Cases:**

1. **Search for Exoplanet Data:**
   * Description: Users can search for exoplanet data based on various criteria such as name, distance from Earth, or habitable zone characteristics.
   * Actors: User
   * Preconditions: User is logged in to the system.
   * Basic Flow:
     1. User enters search criteria.
     2. System retrieves relevant exoplanet data.
     3. System displays search results to the user.
2. **Visualize Exoplanet Properties:**
   * Description: Users can visualize the properties and characteristics of exoplanets through interactive charts, graphs, and maps.
   * Actors: User
   * Preconditions: User is logged in to the system.
   * Basic Flow:
     1. User selects an exoplanet from the search results.
     2. System generates visualizations of the selected exoplanet's properties.
     3. User interacts with the visualizations to explore different aspects of the exoplanet.
3. **Predict Exoplanet Habitability:**
   * Description: Users can see the likelihood of exoplanet habitability based on machine learning models trained on stellar and exoplanet features.
   * Actors: User
   * Preconditions: User is logged in to the system.
   * Basic Flow:
     1. User selects an exoplanet from the search results.
     2. System runs the selected exoplanet through the habitability prediction model.
     3. System provides the user with the predicted likelihood of habitability for the selected exoplanet.

**Server-Side Use Cases:**

1. **Receive Exoplanet Data Request:**
   * Description: Server receives requests for exoplanet data from the user interface.
   * Actors: Server
   * Preconditions: None
   * Basic Flow:
     1. Server receives a request for exoplanet data from the user interface.
     2. Server processes the request and retrieves the requested data from the database or external sources.
     3. Server sends the requested data back to the user interface for display.
2. **Run Exoplanet through ML Model:**
   * Description: Server runs selected exoplanets through the machine learning model to predict habitability.
   * Actors: Server
   * Preconditions: None
   * Basic Flow:
     1. Server receives a request to predict the habitability of selected exoplanets.
     2. Server retrieves the necessary data for the selected exoplanets from the database.
     3. Server applies the machine learning model to the data to predict habitability.
     4. Server sends the predicted habitability back to the user interface for display.
3. **Handle Authentication and Authorization:**
   * Description: Server handles user authentication and authorization for accessing sensitive data or features.
   * Actors: Server
   * Preconditions: None
   * Basic Flow:
     1. Server receives user authentication request from the user interface.
     2. Server verifies the user's credentials.
     3. If authenticated, server checks the user's permissions for accessing the requested data or feature.
     4. Server grants or denies access based on the user's permissions.

**Software Development Plan**

**Project Overview**

The Habitable Exoplanet Explorer project aims to develop a web application that allows users to explore data related to exoplanets and assess their habitability. The project involves data acquisition, preprocessing, analysis, visualization, and prediction using machine learning models.

**Project Objectives**

1. **Data Acquisition and Preparation:**
   * Gather exoplanet data from sources such as NASA's Kepler and TESS missions.
   * Clean and preprocess the collected data for analysis.
2. **Habitable Zone Identification:**
   * Calculate the habitable zone for each star based on stellar properties.
   * Define regions where planets could potentially support liquid water.
3. **Exoplanet Characterization:**
   * Estimate key parameters of exoplanets using advanced algorithms.
   * Incorporate atmospheric data to assess potential habitability.
4. **Machine Learning for Habitability Prediction:**
   * Train machine learning models to predict the likelihood of habitability based on stellar and exoplanet features.
   * Evaluate and optimize model performance for accurate predictions.
5. **Interactive Visualization and Outreach:**
   * Develop an interactive visualization tool to showcase habitable exoplanets and engage the public.

**Development Methodology**

The project will follow an Agile development methodology, with iterative sprints focusing on delivering incremental functionality. The development process will involve regular meetings, continuous collaboration, and feedback from stakeholders to ensure alignment with project goals.

**Development Tools and Technologies**

1. **Frontend Development:**
   * React.js for building the user interface.
   * Various libraries will be used for interfacing as well as visualizing of the data
2. **Backend Development:**
   * Node.js with Express.js for building the server-side logic.
   * SQL database for storing and retrieving exoplanet data.
3. **Machine Learning:**
   * Python with libraries such as TensorFlow and Scikit-learn for developing machine learning models.
   * Jupyter Notebook for prototyping and experimentation.
4. **Version Control:**
   * Git for version control and collaboration.
   * GitHub or GitLab for repository hosting.
5. **Deployment:**
   * Docker for containerization of application components.
   * Continuous Integration/Continuous Deployment (CI/CD) pipelines for automated deployment.

**Quality Assurance**

1. **Testing:**
   * Unit testing, integration testing, and end-to-end testing to ensure functionality and reliability.
   * Performance testing to assess system responsiveness and scalability.
   * Security testing to identify and address potential vulnerabilities.
2. **Documentation:**
   * Comprehensive documentation of code, APIs, and system architecture for future reference and maintenance.