High Level Design Document

Introduction

This High Level Design (HLD) document outlines the architecture and core components for **ExplainX - Model Interpretability Dashboard**. ExplainX is a Streamlit-based dashboard enabling users to upload machine learning models and datasets, visualize feature importance, and generate local explanations using SHAP and LIME. The HLD provides an implementation-focused overview for developers and stakeholders.

1. System Architecture Overview

Architecture Description:
ExplainX is a modular, web-based application built with Streamlit. It processes user-uploaded models and datasets, computes interpretability metrics using SHAP and LIME, and presents results via interactive visualizations.

Main System Modules

Module	Role
UI Layer (Streamlit)	User interaction, file uploads, parameter selection, result display
Data Handler	Validates, parses, and preprocesses uploaded datasets
Model Loader	Loads and validates user-uploaded ML models (scikit-learn compatible)
Explanation Engine	Computes global (feature importance) and local (instance-level) explanations using SHAP and LIME
Visualization Engine	Generates and renders plots/visualizations for interpretability

2. Component Interactions

Sequence Step	Interaction Description
1	User uploads model and dataset via UI Layer
2	Data Handler validates and preprocesses dataset
3	Model Loader loads and validates the ML model
4	Explanation Engine computes SHAP/LIME explanations using model and data
5	Visualization Engine generates plots and sends them to UI Layer for display
6	User interacts with visualizations and requests local explanations as needed

3. Data Flow Overview

Data Source	Consumed By	Output/Result
Uploaded Dataset	Data Handler	Cleaned/preprocessed data
Uploaded Model	Model Loader	Loaded/validated model object
Data & Model	Explanation Engine	SHAP/LIME explanation objects
Explanations	Visualization Engine	Plots/visualizations (feature importance, local explanations)
Visualizations	UI Layer	Interactive display to user

4. Technology Stack

Layer/Function	Technology/Framework
Web UI	Streamlit
ML Model Handling	scikit-learn
Data Processing	pandas, numpy
Model Interpretability	SHAP, LIME
Visualization	Streamlit built-in, matplotlib/plotly (as needed)
Language	Python

5. Scalability & Reliability

- **Scalability:** Designed for single-user or small group usage; can be containerized for deployment. For larger datasets/models, consider resource limits and optimize data processing.
- **Reliability:** Input validation for models and datasets; error handling for unsupported formats; clear user feedback on failures.
- **Security:** User uploads are processed in-memory; no persistent storage of sensitive data by default.

End of Document