

# Project Architecture Document

---

## Retinal disease detection using OCT Images.

---

### Table of Contents

- [1. Introduction](#)
- [2. System Architecture](#)
- [3. Communication Protocols](#)
- [4. Technologies and Frameworks](#)
- [5. Project Deployment](#)
- [6. Conclusion](#)

## 1. Introduction

The Project Architecture Document provides an overview of the overall architecture and design considerations for the Backorder Prediction System. It outlines the key components, their interactions, and the technologies used in the system.

## 2. System Architecture

The project follows a modular and layered architecture to ensure modifying-ability, scalability, and maintainability. The major components of the system architecture include:

1. **User Interface (UI):** The UI component handles user interactions and provides a user-friendly interface for inputting required information and getting predictions.
2. **Application Layer:** The Application Layer serves as the intermediary between the UI and the backend components. It processes user inputs, performs data validation, and orchestrates the flow of information.
3. **Back-End Services:** The Back-End Services encompass various functionalities required for the prediction process, such as data ingestion, data transformation, data validation, model training, and model prediction.
4. **Machine Learning Model:** The Machine Learning Model component is responsible for predicting the backorder status based on the input data. It utilizes trained models, algorithms, and statistical techniques to make accurate predictions.
5. **Data Storage:** The Data Storage component stores the input data, predictions, and other

relevant information. It can include databases, file systems storage solutions.

### 3. Communication Protocols

To enable smooth communication and data flow between the system components, the following protocols can be employed:

- **HTTP/HTTPS:** Used for communication between the UI, Application Layer, and Back-End Services.
- **RESTful APIs:** Allow for standardized and stateless communication between components.
- **Database Protocols:** Facilitate data storage and retrieval operations between the Back-End Services and the Data Storage component.

### 4. Technologies and Frameworks

This project utilizes the following technologies and frameworks:

#### Front-End:[ Gradio ]

- A tool for quickly building and sharing interactive UIs for machine learning models. It's used here to create a user-friendly interface for interacting with the retinal issue detection model.

#### Back-End:[ Python ]

- Core language for server-side logic and interaction with the AI model.

#### AI/ML

- PyTorch: Framework for running inference with deep learning models.
- Hugging Face Transformers: Library for deploying pre-trained models, such as the one used for retinal issue detection.

### 5. Project Deployment

This project is deployed on Huggingface cloud. [See Here](#)

### 6. Conclusion

The Project Architecture Document outlines the architecture and design considerations for the project. It highlights the key components, their interactions, and the technologies used in the system. This document serves as a reference for the development team, guiding the implementation and ensuring the system's modifying-ability, scalability, and maintainability.