**PFL Project One**

**High Level Design Document**

Version: 1.0

Date: April 12, 2025

Author: Architecture Team

Table of Contents

Right-click to update table of contents.

# Introduction

## Purpose

This High Level Design Document (HLDD) provides an overview of the system architecture.

## Scope

This document describes the high-level design of the system including its components, interfaces, data structures, and technologies.

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Term/Acronym** | **Definition** |
| HLDD | High Level Design Document |
| API | Application Programming Interface |
| DB | Database |

# System Overview

This section provides an overview of the system, its context, and its primary functions.

## System Context

This section describes the system context and its interactions with external systems and users.

## System Functions

* Function 1
* Function 2
* Function 3

## User Roles

|  |  |
| --- | --- |
| **Role** | **Description** |
| Administrator | System administrator with full access |
| Application Owner | Owner responsible for the application |
| Developer | Software engineer working on the application |
| End User | User who interacts with the application |

# Architecture Overview

Scalable and maintainable architecture design

## Architecture Principles

* Scalability: The system should scale horizontally to handle increasing load
* Reliability: The system should be resilient to failures and maintain high availability
* Security: The system should implement defense in depth and follow security best practices
* Maintainability: The system should be easy to maintain and update

## Architecture Diagram

The following diagram illustrates the high-level architecture of the system:

[Architecture Diagram Placeholder]

## Component Descriptions

### Frontend

User interface layer

### Backend

Business logic and data processing

### Database

Data storage and retrieval

# Technology Stack

Modern, scalable technology stack

## Frontend

|  |  |  |
| --- | --- | --- |
| **Technology** | **Version** | **Purpose** |
| React | Latest | User interface framework |

## Backend

|  |  |  |
| --- | --- | --- |
| **Technology** | **Version** | **Purpose** |
| Python (Flask/Django) | Latest | Server-side logic and API |

# Data Architecture

This section describes the data architecture of the system including data models, storage, and flows.

## Data Stores

### Primary Database

**Type:** Relational **Purpose:** Main application data

### Cache

**Type:** In-memory **Purpose:** Temporary data caching

## Data Models

The following data models are used in the system:

[Data Model Diagram Placeholder]

## Data Flows

The following diagram illustrates the data flows in the system:

[Data Flow Diagram Placeholder]

# Security Architecture

Enhanced security for moderately sensitive applications

## Authentication and Authorization

Multi-factor authentication

## Data Protection

Advanced encryption (AES-256)

## Network Security

Network segmentation, advanced firewall rules

## Compliance

The system is designed to comply with relevant regulatory requirements.

# Deployment Architecture

This section describes the deployment architecture of the system.

## Deployment Diagram

The following diagram illustrates the deployment architecture of the system:

[Deployment Diagram Placeholder]

## Environments

### Development

Used for development and testing

### Staging

Used for integration testing and UAT

### Production

Used for the live system

## CI/CD Pipeline

The system uses a CI/CD pipeline for automated testing and deployment.

# Operational Considerations

This section describes the operational considerations for the system.

## Monitoring and Logging

The system implements monitoring and logging mechanisms to ensure operational visibility.

## Backup and Recovery

The system implements backup and recovery mechanisms to ensure data durability.

## Scaling and Performance

The system is designed to scale to handle increasing load and maintain performance.

## Disaster Recovery

The system implements disaster recovery measures to ensure business continuity.

# Appendices

## References

* Reference 1
* Reference 2
* Reference 3

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 1.0 | 2025-04-12 | Architecture Team | Initial version |