High Level Design Document

This High Level Design (HLD) document outlines the architecture and core components for **HealthBridge - Integrated Patient Data Exchange**, a secure, open-source platform enabling healthcare providers to exchange, analyze, and collaborate on anonymized patient data across organizations.

Project Name

HealthBridge - Integrated Patient Data Exchange

Purpose:
To provide a secure, interoperable, and extensible web platform for compliant patient data exchange, real-time analytics, AI-driven insights, and collaborative case management among healthcare organizations.

1. System Architecture Overview

Architecture Description:
HealthBridge is a modular, containerized web application built on Django, supporting RESTful APIs, real-time analytics, and machine learning. The system is composed of backend services, a PostgreSQL database, data integration pipelines, analytics/ML modules, and a web frontend. All components are orchestrated via Docker for scalability and portability.

Main System Modules

Module	Role
API Gateway	Handles REST API requests, authentication, and routing
User & Org Management	Manages users, roles, organizations, and permissions
Data Exchange Engine	Secure upload/download, FHIR/CSV parsing, anonymization, validation
Interoperability Layer	FHIR mapping, EHR data transformation
Analytics & Visualization	Real-time dashboards, cohort analysis, report export
AI Insights Engine	ML models for predictions and recommendations
Case Management	Collaborative workspaces, discussions, audit trails
Database	Stores all structured data (PostgreSQL)
Frontend	Web UI for all user roles, dashboards, and collaboration

2. Component Interactions

Interaction Step	Description

1. User Authenticates	API Gateway verifies user via OAuth2, enforces role-based access
2. Data Upload/Download	Providers upload/download records via Data Exchange Engine; data is anonymized and validated
3. Data Integration	Interoperability Layer maps incoming data to FHIR, transforms as needed
4. Analytics Request	User requests analytics; Analytics module queries DB, generates visualizations
5. AI Insights	Provider requests insights; AI Engine processes data, returns recommendations
6. Case Collaboration	Users interact in shared workspaces; actions logged in audit trail

3. Data Flow Overview

Data Flow	Source	Destination	Notes
Patient Data Upload	Provider	Data Exchange	Anonymized, validated, mapped to FHIR
Data Storage	Data Exchange	Database	Secure, encrypted at rest
Analytics Query	Frontend/API	Analytics Module	Real-time, <2s response for up to 10k records
AI Model Input/Output	Database	AI Insights Engine	Feedback loop for model improvement
Case Management Activity	Users	Case Module	Audit trail maintained
Report Export	Analytics Module	User	PDF/CSV formats

4. Technology Stack

Layer/Component	Technology/Frameworks
Backend/API	Django 3.10+, Django REST Framework
Database	PostgreSQL 14+
Data Integration	Custom ETL, HL7 FHIR mapping utilities
Machine Learning	scikit-learn, TensorFlow/PyTorch
Frontend	Django templates, Chart.js, D3.js
Containerization	Docker, docker-compose
API Documentation	OpenAPI/Swagger
CI/CD	GitHub Actions, Docker Hub

5. Scalability, Reliability & Security

- **Scalability:**
  Dockerized microservices enable horizontal scaling; stateless API design supports load balancing.

- **Reliability:**
  99.5% uptime target; robust error handling, audit logging, and automated CI/CD pipelines.
- **Security & Compliance:**
  End-to-end TLS encryption, strict data anonymization, role-based access, audit trails, and HIPAA/GDPR-ready controls.

End of High Level Design Document