High Level Design Document

Introduction

This High Level Design (HLD) document outlines the architecture and core components for **FitArena - Sports Performance Analytics Platform**. FitArena is a web application enabling sports organizations to analyze athlete performance using big data and AI, featuring real-time dashboards, predictive analytics, and team collaboration tools.

1. System Architecture Overview

Architecture Summary:
FitArena is a modular, cloud-hosted web application with a microservices-inspired backend, scalable data pipelines, and a responsive frontend. The system is composed of the following main modules:

Module	Role
Frontend (Web App)	User interface for data upload, visualization, collaboration
API Gateway	Entry point for all client requests, authentication, routing
Backend Services	Business logic, data processing, analytics, and ML recommendations
Data Processing Engine	ETL, data validation, transformation, aggregation
Machine Learning Engine	Predictive analytics, personalized recommendations
Database (MongoDB)	Scalable storage for user, team, and performance data
File Storage	Stores uploaded CSV/Excel files
Notification Service	Sends alerts and recommendations to users

2. Component Interactions

Sequence Step	Interaction Description
1. User logs in via frontend	Auth request sent to API Gateway (JWT/OAuth2)
2. Data upload (CSV/Excel/API)	Frontend → API Gateway → Data Processing Engine
3. Data validation & processing	Data Processing Engine cleans/transforms, stores in DB
4. Analytics/dashboard request	Frontend → API Gateway → Backend Services → Database
5. ML recommendations generation	Backend triggers ML Engine; results stored/retrieved
6. Visualization & collaboration	Frontend fetches analytics, visuals, and team data
7. Notifications/alerts	Backend → Notification Service → User

3. Data Flow Overview

Data Flow	Source	Destination	Purpose
User Data Upload	Frontend	Data Processing	Ingest and validate performance data
Processed Data Storage	Data Processing	MongoDB	Store cleaned/aggregated data
Analytics Query	Frontend	Backend Services	Fetch metrics for dashboards/reports
ML Model Input/Output	Backend Services	ML Engine	Generate predictions/recommendations
Collaboration Data	Frontend	Backend Services	Share insights, notes, discussions
Notification Events	Backend Services	Notification Service	Alert users of anomalies/milestones

4. Technology Stack

Layer/Component	Technology/Frameworks
Frontend	Vue.js, Responsive Design, Chart Libraries
Backend/API	Python (FastAPI/Flask), RESTful APIs
Data Processing	Python, Pandas
Machine Learning	Scikit-learn, Custom ML Models
Database	MongoDB (Cloud-hosted, Sharded)
File Storage	Cloud Object Storage (AWS S3/GCP/Azure Blob)
Authentication	JWT, OAuth2
Deployment	Docker, CI/CD, Cloud Hosting (AWS/GCP/Azure)
Security	Data Encryption, Role-Based Access, Audit Logs

5. Scalability, Reliability & Security

- **Scalability:**
  - Cloud-native deployment with auto-scaling for backend and database
  - Sharded MongoDB for large datasets and high user concurrency
  - Asynchronous data processing pipelines
- **Reliability:**
  - 99.5% uptime target, automated backups, health checks
  - CI/CD for rapid, safe deployments
- **Security:**
  - JWT/OAuth2 authentication, role-based access control
  - Data encryption in transit and at rest
  - GDPR compliance, audit logging, user data export/delete

End of High Level Design Document