AI-Powered Application for Early Detection of Heart Disease Risk

Milestone 1: Data Foundation & OCR Implementation

Objective: Establish data pipeline, implement OCR functionality, and prepare datasets

Tasks:

1. **Environment Setup & Data Collection**
    - Install Python, scikit-learn, pandas, numpy, and medical ML libraries
    - Set up development environment (IDE, Git repository)
    - Research and collect heart disease datasets (UCI Heart Disease, Framingham, Cleveland)
    - Download and organize datasets with proper structure
2. **Data Analysis & Preprocessing**
    - Analyze dataset features (age, sex, chest pain, blood pressure, cholesterol, etc.)
    - Handle missing values using appropriate imputation techniques
    - Identify and remove outliers and noisy data
    - Perform exploratory data analysis (EDA) with visualizations
3. **OCR Implementation for Medical Documents**
    - Install and configure OCR libraries (Tesseract, pytesseract, OpenCV)
    - Implement document preprocessing (image enhancement, noise reduction)
    - Create OCR pipeline to extract text from medical reports (PDF, images)
    - Develop text parsing logic to identify and extract relevant medical values
    - Handle different document formats and layouts
4. **Data Standardization & Feature Engineering**
    - Normalize and standardize numerical features
    - Encode categorical variables (one-hot encoding, label encoding)
    - Create derived features (BMI calculation, risk factors combination)

- Split datasets into training (70%), validation (15%), and testing (15%) sets

Deliverables:

- Clean, preprocessed heart disease dataset
- Working OCR system for medical document processing
- Feature engineering pipeline with documented transformations
- Data analysis reports with insights

Milestone 2: Model Development & Training

Objective: Design, implement, and train machine learning models for heart disease prediction

Tasks:

1. **Model Architecture Design**
    - Research suitable ML algorithms (Random Forest, SVM, Logistic Regression, Neural Networks)
    - Design model architecture considering medical data characteristics
    - Implement baseline models for comparison
    - Create model evaluation framework
2. **Model Implementation & Training**
    - Implement multiple ML algorithms using scikit-learn/TensorFlow
    - Train models with different hyperparameter configurations
    - Apply cross-validation techniques for robust evaluation
    - Implement ensemble methods for improved accuracy
3. **Model Optimization & Hyperparameter Tuning**
    - Use Grid Search/Random Search for hyperparameter optimization
    - Implement feature selection techniques (LASSO, RFE, correlation analysis)
    - Apply regularization techniques to prevent overfitting
    - Optimize model performance metrics (accuracy, precision, recall, F1-score)

4. **Model Evaluation & Validation**
    - Test models on reserved test dataset
    - Generate comprehensive evaluation metrics and confusion matrices
    - Implement ROC-AUC analysis for risk assessment
    - Create model comparison reports with statistical significance tests
    - Validate model interpretability for medical applications
5. **Risk Categorization System**
    - Implement probability threshold system for Low/Moderate/High risk categories
    - Calibrate model outputs for reliable probability estimates
    - Create risk score calculation logic
    - Validate risk categories against medical guidelines

Deliverables:

- Trained ML models with >85% accuracy
- Model evaluation reports with comprehensive metrics
- Risk categorization system (Low/Moderate/High)
- Saved model files and preprocessing pipelines

Milestone 3: UI Development & System Integration

Objective: Create complete web application with user authentication and real-time prediction

Tasks:

1. **Backend Development**
    - Set up Flask/Django web framework
    - Create API endpoints for model prediction
    - Implement file upload handling for medical documents
    - Integrate OCR processing with web backend
    - Create database schema for user data and predictions
2. **User Authentication System**

- Implement user registration and login functionality
- Create secure password hashing and session management
- Design user profile management system
- Implement role-based access control (patient/doctor views)
- Add password recovery and email verification

3. **Frontend UI Development**
    - Design responsive web interface using HTML/CSS/JavaScript
    - Create user-friendly forms for manual data input
    - Implement file upload interface for medical documents
    - Design results dashboard with clear risk visualization
    - Add interactive charts and health indicator displays
4. **System Integration & Real-time Functionality**
    - Connect frontend with backend APIs
    - Implement real-time prediction processing
    - Create progress indicators for document processing
    - Add result storage and history functionality
    - Implement export functionality for reports
5. **Data Input Methods Integration**
    - Manual form input with validation and error handling
    - OCR document upload with preview and confirmation
    - Data extraction verification interface
    - Hybrid input method (manual + OCR) support

Deliverables:

- Complete web application with authentication
- Functional UI for data input and result display
- Integrated OCR and prediction system
- User dashboard with prediction history

Milestone 4: Documentation & Infosys Presentation Preparation

Objective: Prepare comprehensive documentation and professional presentation materials for Infosys

Tasks:

1. **Technical Documentation Creation**
    - System architecture documentation with diagrams
    - API documentation and database schema
    - Model development methodology and performance metrics
    - Code documentation with comments and explanations
    - Installation and setup guide for deployment
2. **User Documentation & Guides**
    - Comprehensive user manual with screenshots
    - Step-by-step user guide for both input methods
    - Medical professional guide for result interpretation
    - Troubleshooting guide and FAQ section
    - System requirements and compatibility documentation
3. **Project Analysis & Performance Reports**
    - Final model evaluation and benchmarking results
    - OCR accuracy analysis with different document types
    - System performance analysis and metrics
    - Comparison with existing heart disease prediction solutions
    - Limitations, challenges, and future improvement recommendations
4. **Infosys Presentation Materials**
    - Professional PowerPoint presentation (20-25 slides)
    - Executive summary highlighting key achievements
    - Live demo script with key talking points

    - Case studies demonstrating successful predictions
    - Problem statement, solution approach, and outcomes summary
5. **Demo Preparation & Practice**
    - Create compelling demo scenarios with sample data
    - Prepare backup demo materials (screenshots, videos)
    - Practice presentation delivery and timing
    - Prepare answers for potential questions
    - Create handout materials for audience
6. **Final Submission Package**
    - Complete source code with proper organization
    - All documentation in professional format
    - Presentation materials and demo resources
    - Project timeline and milestone achievement summary
    - Individual contribution reports and learning outcomes

Deliverables:

- Complete technical and user documentation
- Professional presentation deck for Infosys
- Demo-ready materials and backup resources
- Final project submission package
- Individual learning and contribution reports