AI Astra Project Management Plan

Cancer Detection using Machine Learning

Duration: 6 Weeks (3 Sprints × 2 Weeks)

Team Size: 6 people

Methodology: Agile with Scrum Framework

Prepared by: Project Team

Project Management Plan

Project Goal

Build a classification model to detect whether a person has cancer or not.

Duration

6 Weeks (3 Sprints × 2 Weeks)

Team Size

6 people

Methodology

Agile with Scrum framework

Sprint 1 (Week 1–2): Data Preparation & Project Setup

Objective: Set up environment, collect/clean data, explore dataset, and prepare for modeling.

Tasks

1. Project Setup
• Develop data preparation script.
• Generate cleaned dataset.
• Prepare Software Requirements Specification (SRS) document.
• Initial Project Management Plan, and assign roles.

2. Data Collection & Understanding
• Gather datasets (medical/public cancer datasets or hospital-provided).
• Perform exploratory data analysis (EDA).
• Handle missing values (continuous & categorical).
• Understand class imbalance (if many 'no cancer' vs few 'cancer').

3. Preprocessing
• Feature engineering (scaling, encoding categorical variables, handling outliers).
• Data split into train/validation/test.

Deliverables

- Cleaned, well-documented dataset
- System Requirement Specification draft
- Preprocessing pipeline ready

Sprint 2 (Week 3–4): Model Development & Training

Objective: Train baseline models, optimize, evaluate, and persist models for deployment.

Tasks

1. Data Preparation and Preprocessing (AI_Astra_DataPrep.py)
• Load cleaned dataset (AI_Astra_CleanedData.csv).
• Perform feature transformation: LabelEncoder for categorical features, StandardScaler for numerical features.
• Save preprocessing artifacts: standard_scaler.pkl, label_encoders.pkl.
• Save processed dataset (AI_Astra_ProcessedDataset.xlsx).

2. Model Training and Persistence (AI_Astra_ModelTrainAndSave.py)
• Load processed dataset.
• Train models: Random Forest, Gradient Boosting, Logistic Regression, SVM, KNN.
• Evaluate models using Accuracy, Precision, Recall, F1.
• Save trained models (.pkl files).

- Save performance metrics report (AI_Astra_ModelMetrics.xlsx).

3. Unified Pipeline (AI_Astra_ModelDevAndDeploy.py)
- Combined script handling preprocessing, training, and persistence.
- Outputs processed dataset, model metrics, and serialized models.

4. Prediction and Documentation
- Deliver PredictionReport.zip containing prediction charts and documentation.
- Update Project Management documentation.

Deliverables

- AI_Astra_ModelDevAndDeploy.py
- AI_Astra_ProcessedDataset.xlsx
- AI_Astra_ModelMetrics.xlsx
- PredictionReport.zip (Prediction Report + Charts + Documentation)
- Documentation.pdf (Updated PM Documentation)

Expanded Methodology (Sprint 2)

- Common Classification Models:

- Logistic Regression: Interpretable baseline classifier.
- Support Vector Machine (SVM): Separates classes using optimal hyperplane.
- K-Nearest Neighbors (KNN): Classifies based on similarity to nearest neighbors.
- Decision Trees: Tree-based interpretable structure.
- Random Forest: Ensemble of decision trees reducing overfitting.

- Classification Evaluation Metrics:

- Precision: Of predicted positives, proportion that are true positives.
- Recall: Of actual positives, proportion correctly identified.
- Accuracy: Proportion of total correct predictions.
- F1-Score: Harmonic mean of precision and recall.
- AUC-ROC: Area under curve for trade-off between recall and false positive rate.

- Imbalanced Data:

Occurs when one class has far fewer samples than the other (e.g., cancer cases vs non-cancer). Accuracy may be misleading. Metrics like Recall and F1 are emphasized.

- Cost of Errors:

False Positives: Predicting cancer when absent → unnecessary stress/tests.
False Negatives: Missing a cancer case → severe consequences.
In cancer detection, minimizing false negatives is prioritized, hence Recall and F1 are critical.

Sprint 3 (Week 5–6): Deployment, Testing & Final Report

Objective: Finalize best model, test robustness, deploy, and prepare project documentation.

Tasks

1. Model Finalization
- Select best-performing model.
- Test on holdout test set.
- Check fairness & bias (avoid false negatives).

2. Deployment
- Build API/Flask/Django app for model inference.
- Containerize with Docker (optional).
- Prepare demo UI (simple web form for input + output).

3. Project Documentation & Reporting
- Prepare final report (dataset description, methods, results, conclusion).
- Create presentation deck.
- Assign team members to present.

Deliverables

- Final cancer detection model (with reproducible code)
- Deployed demo (API/web app)
- Final report & presentation

Project Timeline (Summary)

Sprint	Week	Focus Area	Key Deliverables
Sprint 1	1–2	Data collection, cleaning, preprocessing	Clean dataset, EDA report, preprocessing pipeline
Sprint 2	3–4	Modeling, training, evaluation, persistence	Processed dataset, trained models, model metrics, prediction reports, updated documentation
Sprint 3	5–6	Deployment & final reporting	Final model, deployed app, final report, presentation.

Roles in Project

- Project manager – Rohit Singh
- Data engineer – Dhrithi MV
- Data scientists – Lavanya K Gowda
- Machine learning engineer – Kusumitha K P
- Quality test engineer – Sourabh Rajendragouda Doddagoudar
- ML op's – Keerthana V and Likhitha D

Table of Contents