1**. Problem Definition** (6 points)

Hypothetical AI Problem

Predicting Patient Readmission Risk within 30 Days of Hospital Discharge

Objectives

Reduce readmission rates by identifying high-risk patients early.

Improve patient care plans through personalized post-discharge recommendations.

Optimize hospital resources by allocating attention to at-risk individuals.

Stakeholders:

Hospital Administrators – interested in cost control and operational efficiency.

Medical Practitioners – responsible for patient treatment and follow-up care.

Key Performance Indicator (KPI):

Readmission Rate Reduction (%) – Measures the percentage decrease in patient readmissions within 30 days compared to the historical baseline.

2. **Data Collection & Preprocessing** (8 points)

a) Two Data Sources

Electronic Health Records (EHRs) – Includes patient demographics, diagnosis codes, lab results, medications, and previous admissions.

Patient Discharge Summaries & Follow-up Logs – Contains details about post-discharge instructions, follow-up appointments, and outcomes.

b) Potential Bias:

Healthcare Access Bias: Patients from underprivileged backgrounds may have fewer follow-ups documented, leading the model to underpredict their readmission risk.

c)Three Preprocessing Steps

Handling Missing Data-Impute missing values using median/mode for numerical/categorical variables.

Normalization/Standardization-Scale numerical features (e.g., lab values) using Min-Max normalization or z-score standardization to ensure consistency.

Encoding Categorical Variables-Apply one-hot encoding for categorical features like diagnosis codes or insurance types.

3. Model Development (8 points)

Chosen Model- Random Forest Classifier

Justification:

Random Forest handles nonlinear relationships, mixed data types, and missing values well.

It's also robust to overfitting, especially with a diverse dataset like medical records.

Data Splitting Strategy:

Training Set: 70% – Used to train the model.

Validation Set: 15% – Used to fine-tune hyperparameters and evaluate performance.

Test Set: 15% – Used for final performance evaluation on unseen data.

Two Hyperparameters to Tune:

Number of Trees (estimators)-A higher number can improve accuracy but increases training time. Helps balance bias and variance.

Maximum Depth (max_depth)-Controls the complexity of each tree. Prevents overfitting by limiting how deep each tree can grow.