

Part 1: Short Answer Questions (30 points)

1. Problem Definition (6 points)

Hypothetical AI Problem:

Building an AI system to predict which mothers in rural Kilifi are most likely to stop using the *Sauti Ya Mama* USSD/IVR platform early so that the system can send reminders or extra support before they drop out.

Objectives:

1. Identify mothers at high risk of stopping engagement early.
2. Support them with customized voice messages or SMS to encourage continued learning.
3. Improve long-term use of the Sauti Ya Mama platform.

Stakeholders:

- Mothers using the Sauti Ya Mama service.
- Program coordinators (Sauti Ya Mama team).

Key Performance Indicator (KPI):

Retention rate: percentage of active users after 3 months.

Data Collection & Preprocessing (8 points)

Data Sources:

1. USSD/IVR usage logs (frequency, time spent, session completion).
2. User demographic data (age, literacy level, language preference, location).

Potential Bias:

If most early users are from one area (e.g., Bamba), the model may not work well in other regions with different languages or cultures.

Preprocessing Steps:

1. Handle missing usage data (e.g., when a session fails to record).
2. Normalize engagement data (convert usage counts to a common scale).
3. Encode non-numeric data (e.g., “language preference” or “literacy level”) into numeric form.

3. Model Development (8 points)

Model Choice:

Logistic Regression because it works well for binary outcomes (e.g., “will continue using” vs. “will drop out”) and is easy to interpret by project staff.

Data Split:

- 70% Training data
- 15% Validation data
- 15% Test data

Hyperparameters to Tune:

1. Regularization strength (to control overfitting).
2. Learning rate (to improve model stability and convergence).

4. Evaluation & Deployment (8 points)

Evaluation Metrics:

1. Precision — measures how many users predicted as “likely to drop out” actually did.
2. Recall — measures how many actual dropouts the system was able to identify correctly.

Concept Drift:

Over time, user behavior may change (for example, more women may join with better literacy or different phone habits). This can reduce model accuracy.

Monitoring Plan: Recheck model performance monthly and retrain with new data.

Technical Challenge:

Low network coverage in rural areas may affect data updates or real-time AI feedback to users.

Part 2: Case Study Application (40 points)

Scenario: A hospital wants an AI system to predict patient readmission risk within 30 days of discharge.

Problem Scope (5 points)**Problem:**

Many patients return to the hospital soon after being discharged. This increases costs and workload for healthcare workers.

Objectives:

1. Predict which patients are most likely to be readmitted within 30 days.
2. Help doctors plan follow-up visits for high-risk patients.
3. Reduce hospital costs and improve patient recovery rates.

Stakeholders:

- Doctors and nurses
- Hospital management
- Patients

Data Strategy (10 points)**Data Sources:**

- **Electronic Health Records (EHRs):** Include diagnosis, treatment, and discharge notes.
- **Patient Demographics:** Age, gender, medical history, and lifestyle data (e.g., smoking, exercise).

Ethical Concerns:

1. **Privacy:** Patient data must be protected and only used for medical purposes.
2. **Bias:** The model might favor certain age groups or ignore others if training data is unbalanced.

Preprocessing Pipeline:

1. Remove duplicates and incorrect entries.
2. Handle missing values (for example, replace blank lab results with average values).
3. Feature Engineering:
 - Create new features like “number of past admissions” or “days stayed in hospital.”
 - Normalize numeric data (e.g., blood pressure readings).

Model Development (10 points)

Model: Logistic Regression — simple, interpretable, and good for predicting yes/no outcomes (readmitted or not).

Example Confusion Matrix:

	Predicted Yes	Predicted No
Actual Yes	90 (True Positive)	10 (False Negative)
Actual No	30 (False Positive)	170 (True Negative)

Key:

True Positive (TP) = The model correctly said “Yes,” and the patient was indeed readmitted.

False Positive (FP) = The model said “Yes,” but the patient was *not* readmitted.

False Negative (FN) = The model said “No,” but the patient was readmitted.

True Negative (TN) = The model correctly said “No,” and the patient was not readmitted.

Workings:

Precision and Recall (Performance Scores)

Precision answers:

"Of all the patients the model said will be readmitted, how many really were?"

Formula:

$$\text{Precision} = \text{TP} / \text{TP} + \text{FP}$$

$$\Rightarrow : 90 / (90 + 30) = 0.75 \rightarrow 75\%$$

This means **75%** of the time, when the model predicts a readmission, it's correct.

Recall answers:

"Of all the patients who were truly readmitted, how many did the model catch?"

Formula:

$$\text{Recall} = \text{TP} / \text{TP} + \text{FN}$$

$$\Rightarrow 90 / (90 + 10) = 0.90 \rightarrow 90\%$$

This means the model correctly identifies **90%** of all actual readmissions.

Conclusion:

A **high recall (90%)** means the model is good at identifying most patients at risk, which is very important in healthcare, as missing a sick patient can be dangerous.

A **good precision (75%)** means most of the alerts it gives are helpful, though there are still some false alarms (25%).

Logistic Regression is used because it easily predicts two outcomes, in this case, whether a patient will be readmitted or not. It studies patient details like age, past admissions, and diagnosis to calculate the chance of coming back within 30 days.

The confusion matrix helps check how correct the model is by comparing what it predicted and what really happened. Precision shows how often the model's "yes" predictions are right, while recall shows how many real readmissions the model successfully caught. In this example, a precision of 75% and recall of 90% mean the model finds most patients who are at risk and gives useful alerts most of the time. This makes it reliable and easy for doctors to use when planning patient follow-ups.

Deployment (10 points)

Integration Steps:

1. Deploy the model as part of the hospital's internal software.
2. When a patient is discharged, the model predicts their readmission risk.
3. Doctors receive alerts for high-risk patients so they can plan follow-ups.

Compliance with Healthcare Regulations (e.g., HIPAA):

- Encrypt all patient data.
- Give access only to authorized medical staff.
- Store data securely and delete it when no longer needed.

Optimization (5 points)

Method to Reduce Overfitting:

Use cross-validation which means testing the model on different small parts of the data to make sure it works well every time. Regularization (L2 penalty) can also be used, to keep the model from becoming too complex.

Part 3: Critical Thinking (20 points)

Ethics & Bias (10 points)

If the training data mostly includes patients from one region or age group, the model may not work well for others. For example, it might predict higher risks for older patients just because most training examples were older people. This kind of bias can lead to unfair treatment or missed care for younger patients. To reduce bias, hospitals should collect data from different age groups, genders, and backgrounds, and regularly test the model for fairness before and after deployment.

Trade-offs (10 points)

There is often a trade-off between how accurate a model is and how easy it is to understand. Complex models like deep neural networks might give more accurate predictions but are hard for doctors to interpret. Simpler models, like Logistic Regression, may be slightly less accurate but are easier to explain and trust in medical decisions. If the hospital has limited computing power, it is better to use lightweight models like Logistic Regression or Decision Trees, which run faster and still give reliable results.

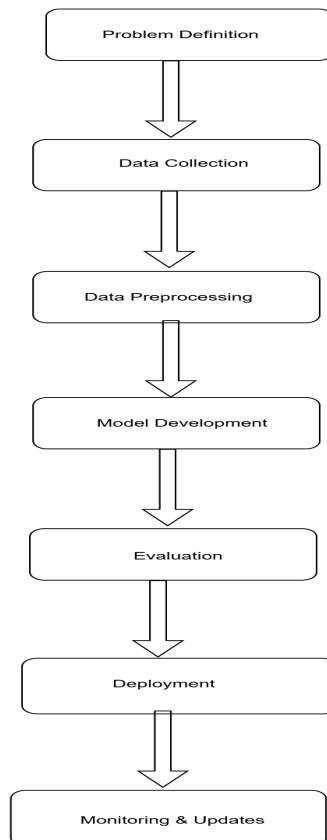
Part 4: Reflection & Workflow Diagram (10 points)

Reflection (5 points)

The most challenging part of the workflow was data preprocessing. Patient data often has missing or inconsistent values, and it takes time to clean and organize it before training the model. I also found it tricky to decide which features were most useful for predicting readmissions. With more time and resources, I would collect more balanced data from different hospitals and test other models to compare their performance. I would also automate the data cleaning steps to save time in future projects.

Diagram (5 points)

AI Development Workflow used in this project



References

1. IBM (1999). *CRISP-DM 1.0: Step-by-Step Data Mining Guide*.
2. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
3. PowerLearn Project Lecture Notes (2025). *AI for Software Engineering – Week 5*.
4. U.S. Department of Health & Human Services (1996). *Health Insurance Portability and Accountability Act (HIPAA)*.
5. Kenya National Bureau of Statistics (2023). *Kenya Demographic and Health Survey Factsheet – Kilifi County*.