Start coding or generate with AI.

Part 1: Short Answer Questions (30 points)

1. Problem Definition (6 points)

Define a hypothetical AI problem (e.g., "Predicting student dropout rates").

List 3 objectives and 2 stakeholders.

Propose 1 Key Performance Indicator (KPI) to measure success.

2. Data Collection & Preprocessing (8 points)

Identify 2 data sources for your problem.

Explain 1 potential bias in the data.

Outline 3 preprocessing steps (e.g., handling missing data, normalization).

3. Model Development (8 points)

Choose a model (e.g., Random Forest, Neural Network) and justify your choice.

Describe how you would split data into training/validation/test sets.

Name 2 hyperparameters you would tune and why.

4. Evaluation & Deployment (8 points)

Select 2 evaluation metrics and explain their relevance.

What is concept drift? How would you monitor it post-deployment?

Describe 1 technical challenge during deployment (e.g., scalability).

Theme: Using AI to Support SDG 8 – Decent Work and Economic Growth in Zimbabwe AI Problem Focus: Predicting Youth Unemployment Risk (Ages 15–24) in Zimbabwe

1. Problem Definition (6 points) Hypothetical AI Problem Develop a predictive model to identify Zimbabwean youth (ages 15–24) who are at high risk of becoming unemployed or staying unemployed for extended periods.

Objectives:

Detect early indicators of youth unemployment to guide timely intervention.

Assist the Ministry of Youth and partner organizations in targeting job creation programs.

Improve national employment metrics aligned with SDG 8.

Stakeholders:

Ministry of Youth, Sports, Arts and Recreation

NGOs and development agencies (e.g., ILO, UNDP Zimbabwe)

KPI:

Year-over-year reduction in youth unemployment rate in targeted regions (e.g., Mashonaland Central, Matabeleland South).

2. Data Collection & Preprocessing (8 points) Data Sources:

Quarterly Labour Force Survey (QLFS) – Offers data on employment status, job sectors, hours worked, and income.

Demographic and Health Survey (DHS) – Includes household characteristics, education level, access to services.

Potential Bias:

Urban bias: Labor surveys often have higher response rates in urban areas, which may skew insights and model performance in underrepresented rural areas.

Preprocessing Steps:

Impute missing values – Fill gaps in household income or education data using median or KNN-based imputation.

Normalize continuous variables – Scale variables like age, hours worked, and income to ensure model balance.

Encode categorical features – Use one-hot encoding for region, education level, and gender.

3. Model Development (8 points) Model Choice: Random Forest Classifier Reason: Handles mixed data well, is less sensitive to outliers, and gives interpretable feature importance—critical for policy use.

Data Splitting Strategy:

70% for training

15% for validation

15% for testing Stratified sampling based on employment status and geographic region.

Hyperparameters to Tune:

n_estimators – Number of trees; controls overall prediction strength.

max_depth – Maximum depth of trees; prevents overfitting and improves generalization.

4. Evaluation & Deployment (8 points) Evaluation Metrics:

Recall – Critical to correctly identify youth at risk of unemployment.

F1 Score – Balanced measure when class imbalance is present (e.g., fewer unemployed youth in the sample).

Concept Drift: This occurs when patterns in the data change over time due to shifts in the economy, education access, or policy reforms. Monitoring Strategy: Re-train or re-evaluate the model quarterly with new QLFS data. Set alerts for sharp drops in performance.

Deployment Challenge:

Infrastructure limitations – Rural areas may lack reliable internet or digital tools, making it difficult to deploy AI-based screening solutions. A mobile-friendly, offline-first app could help mitigate this.

Part 2: Case Study Application (40 points)

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

Tasks:

Problem Scope (5 points): Define the problem, objectives, and stakeholders.

Data Strategy (10 points):

Propose data sources (e.g., EHRs, demographics).

Identify 2 ethical concerns (e.g., patient privacy).

Design a preprocessing pipeline (include feature engineering steps).

Model Development (10 points):

Select a model and justify it.

Create a confusion matrix and calculate precision/recall (hypothetical data).

Deployment (10 points):

Outline steps to integrate the model into the hospital's system.

How would you ensure compliance with healthcare regulations (e.g., HIPAA)?

Optimization (5 points): Propose 1 method to address overfitting.

Part 2: Case Study Application – Hospital Readmission Prediction

Problem Scope (5 points)

Problem Definition:

Hospital readmissions within 30 days of discharge place a heavy burden on Zimbabwe's already strained healthcare system. These readmissions are often avoidable and can signal gaps in post-discharge care, misdiagnoses, or premature discharges.

An AI system is being developed to predict which patients are at high risk of being readmitted within 30 days after discharge. The goal is to assist medical teams in making more informed decisions regarding discharge plans, follow-ups, and resource allocation.

Objectives:

Predict 30-day readmission risk using patient health records and discharge data.

Enable timely interventions like follow-up visits, home care, or patient education.

Reduce overall readmission rates and improve health outcomes.

Stakeholders:

Primary: Hospital administrators, clinicians, discharge planners

Secondary: Ministry of Health and Child Care, insurance providers, patients

Data Strategy (10 points)

Proposed Data Sources: Electronic Health Records (EHRs)

Includes admission and discharge summaries, diagnosis codes (e.g., ICD-10), lab test results, comorbidities, medications prescribed, procedures performed, and vital signs.

Demographic and Social Data

Includes patient age, gender, location, insurance status, housing situation, and social support indicators (e.g., if they live alone or have caregivers).

Two Ethical Concerns: Patient Privacy and Confidentiality

Health data is extremely sensitive. If improperly handled, it may lead to breaches of patient confidentiality. Any data processing must ensure de-identification, encryption, and compliance with local and international privacy laws (e.g., Zimbabwe Health Information Policy, or HIPAA if referencing best practice).

Algorithmic Bias and Health Inequities

If the training data is skewed toward certain demographics (e.g., urban hospitals, privately insured patients), the model might unfairly perform worse on vulnerable groups (e.g., rural, uninsured, elderly patients). This could lead to inequitable care.

Preprocessing Pipeline (including feature engineering): Missing Data Handling

Impute missing lab results or vitals using population medians or clinician-approved domain values.

For missing demographic fields (e.g., insurance type), use "Unknown" category or conditional imputation.

Feature Engineering

Create new features like:

“Number of previous admissions in last 6 months”

“Average length of stay”

“Discharge type” (e.g., to home, rehabilitation, hospice)

“Risk score of comorbidities” using Charlson Comorbidity Index

Normalisation and Encoding

Normalise continuous values (e.g., blood pressure, glucose levels).

One-hot encode categorical variables (e.g., gender, discharge disposition, diagnosis category).

Model Development (10 points) Model Selected: Gradient Boosting Machine (e.g., XGBoost or LightGBM)

Justification: Gradient boosting models work well on structured, tabular data like EHRs.

They handle missing values, capture non-linear relationships, and rank feature importance, which helps in explaining model decisions to clinicians.

These models often outperform simpler models in clinical prediction tasks when tuned correctly.

Hypothetical Confusion Matrix (on test set of 100 patients)

	Predicted: Readmit	**Predicted: No Readmit**
Actual: Readmit	18 (True Positive)	7 (False Negative)
Actual: No Readmit	5 (False Positive)	70 (True Negative)

Metrics Calculated: Precision =

$T P T P + F P$ = 18 18 + 5 = 18 23 ≈ 0.783 TP+FP TP= 18+5 18= 23 18≈0.783 Recall =

$T P T P + F N$ = 18 18 + 7 = 18 25 = 0.72 TP+FN TP= 18+7 18= 25 18=0.72

Interpretation: Precision (78.3%) means that when the model predicts a patient will be readmitted, it is right most of the time — this helps reduce unnecessary interventions.

Recall (72%) means the model catches the majority of readmissions — crucial in preventing avoidable patient returns.

Deployment (10 points) Steps to Integrate the Model into the Hospital’s System: Develop an API Service

Wrap the trained model in an API (e.g., using Flask or FastAPI) to make predictions accessible from the hospital’s digital systems.

Integrate with Hospital Information System (HIS)

Connect the model API to the hospital’s EHR or discharge planning system to automatically pull patient data and generate risk scores upon discharge.

Build a Clinician Dashboard

Visualize patient readmission risk alongside key factors (e.g., comorbidities, number of prior visits). Display color-coded risk levels (e.g., Low, Medium, High).

Pilot and Iterate

Launch the system in one hospital unit (e.g., Internal Medicine) and gather feedback from clinical staff to refine the model and interface.

Train Hospital Staff

Conduct workshops to train doctors, nurses, and discharge planners on how to interpret and act on the model's outputs responsibly.

Ensuring Compliance with Healthcare Regulations: Data Privacy and Security

Use encryption for data in transit and at rest.

Store data on secure, access-controlled servers.

Ensure strict user authentication and logging of all access.

De-identification and Anonymization

For model training and validation, remove or mask personally identifiable information (PII) such as names, ID numbers, and addresses.

Follow Best Practices from HIPAA (if local regulations are unavailable)

Apply principles of minimum necessary use and data minimization.

Maintain an audit trail of who accessed or modified data.

Regularly review and update security protocols.

Compliance with National Guidelines

Align with Zimbabwe's Health Information and Surveillance Policy and any applicable frameworks under the Ministry of Health.

Optimization (5 points) Method to Address Overfitting: Cross-Validation with Early Stopping

Explanation: Cross-validation:

Split the training data into multiple folds (e.g., 5-fold cross-validation) and train the model on different combinations.

This allows the model to generalize better and ensures it's not just memorizing one dataset slice.

Early stopping (specific to boosting models like XGBoost or LightGBM):

During training, the model's performance on a validation set is monitored.

If performance stops improving after a certain number of rounds (e.g., 50 iterations), training is halted early to avoid overfitting to the training set.

Why this works: It balances learning enough patterns from data without going too deep into the noise.

It's computationally efficient and widely used in clinical machine learning tasks where model trust and generalization are crucial.

Part 3: Critical Thinking (20 points)

Ethics & Bias (10 points):

How might biased training data affect patient outcomes in the case study?

Suggest 1 strategy to mitigate this bias.

Trade-offs (10 points):

Discuss the trade-off between model interpretability and accuracy in healthcare.

If the hospital has limited computational resources, how might this impact model choice?

Ethics & Bias (10 points) How might biased training data affect patient outcomes in the case study? Biased training data can lead to unequal predictions, which may disproportionately affect certain patient groups. For example:

If the dataset contains mostly urban patients, the model might perform poorly on rural patients due to missing health context (e.g., limited follow-up care access).

If elderly or low-income patients are underrepresented, the model may fail to recognize their specific readmission risks, leading to inadequate care or missed interventions.

This could worsen health inequalities and create systemic disparities in treatment and resource allocation.

Strategy to Mitigate Bias: Stratified sampling and fairness-aware evaluation:

Ensure that the training data includes a representative mix of demographics (e.g., age, gender, rural vs. urban).

During evaluation, measure model performance separately for subgroups.

If disparities are found, apply techniques like reweighting, adversarial debiasing, or fairness regularization to correct imbalances.

Trade-offs (10 points) Discuss the trade-off between model interpretability and accuracy in healthcare. High-accuracy models (e.g., deep learning or ensemble methods like XGBoost) often act as “black boxes” — hard for clinicians to understand how predictions are made.

Interpretable models (e.g., logistic regression, decision trees) are easier to explain but may lack predictive power in complex medical data.

In healthcare, where decisions affect lives, interpretability is often prioritized, even if it means sacrificing some accuracy. Doctors must be able to justify decisions to patients and comply with medical ethics and policy.

A balanced approach may involve using a powerful model (like XGBoost) with explainability tools (like SHAP values) to visualize what features drove each prediction.

If the hospital has limited computational resources, how might this impact model choice? Limited resources affect:

Training time – complex models like neural networks may be infeasible.

Real-time predictions – slower models can’t be deployed in clinical workflows.

Implication: The hospital may need to choose lightweight models (e.g., decision trees, logistic regression), or optimized versions of larger models. Alternatively, predictions could be run periodically in batches rather than in real-time.

Part 4: Reflection & Workflow Diagram (10 points)

Reflection (5 points):

What was the most challenging part of the workflow? Why?

How would you improve your approach with more time/resources?

Diagram (5 points):

Sketch a flowchart of the AI Development Workflow, labeling all stages.

Reflection (5 points) What was the most challenging part of the workflow? Why? The most challenging part was the data preprocessing and bias mitigation stage. Real-world healthcare data is often incomplete, messy, and imbalanced across different patient groups. Ensuring fairness while preserving model performance required careful feature engineering, stratification, and ethical consideration — all of which demand both domain knowledge and technical skill.

How would you improve your approach with more time/resources? With more time and resources, I would:

Collect and integrate more diverse data (e.g., social determinants of health like income, housing, and caregiver availability).

Implement explainability tools like SHAP to provide transparent model outputs to clinicians.

Collaborate with clinicians for human-in-the-loop validation to ensure model predictions align with real-world healthcare decisions.

Workflow Diagram (5 points)

```
┌──────────────────────┐
│ 1. Problem Definition│
└──────────┬───────────┘
           ↓
┌──────────────────────┐
│ 2. Data Collection   │
└──────────┬───────────┘
           ↓
┌─────────────────────────┐
│ 3. Data Preprocessing   │
└──────────┬──────────────┘
           ↓
┌──────────────────────┐
│ 4. Model Development│
└──────────┬───────────┘
           ↓
┌──────────────────────┐
│ 5. Model Evaluation  │
└──────────┬───────────┘
           ↓
┌──────────────────────┐
```

```
│ 6. Deployment         │
└───────────┬───────────┘
            ↓
┌───────────────────────┐
│ 7. Monitoring &       │
│    Optimization       │
└───────────────────────┘
```