1. Problem Definition (6 points)
Hypothetical AI Problem: Detecting biased language in online job postings.
Objectives:
1. Identify and flag gender-coded or discriminatory phrases.
2. Suggest neutral language alternatives in real time.
3. Improve diversity in applicant pools over time.
Stakeholders:
• Recruitment teams
• Job seekers from underrepresented groups
KPI: Reduction in flagged biased phrases per 100 job postings over time.
2. Data Collection & Preprocessing (8 points)
Data Sources:
1. Public job listing datasets (e.g., Kaggle's job descriptions dataset)
2. Web-scraped job ads from company websites or job boards (e.g., Indeed, LinkedIn)
Potential Bias: Training data may already reflect historical biases (e.g., overuse of "aggressive,"
"dominant," "rockstar"), which could reinforce the status quo if not corrected.
Preprocessing Steps:
• Text cleaning (punctuation removal, lowercasing)
• Tokenization and lemmatization
• Handling class imbalance (e.g., SMOTE or stratified sampling)
3. Model Development (8 points)
Chosen Model: Fine-tuned BERT (Bidirectional Encoder Representations from Transformers)
Justification: BERT is highly effective for NLP tasks and captures contextual meaning in
language—essential for identifying subtle bias.
Data Split Strategy:
• 70% training
• 15% validation
• 15% testing Stratified sampling ensures representation of different bias types.
Hyperparameters to Tune:
1. Learning rate – to optimize training stability and convergence.
2. Maximum sequence length – to control context depth for longer job descriptions.
4. Evaluation & Deployment (8 points)
Evaluation Metrics:
• F1-Score: Balances precision and recall—critical in detecting rare biased phrases.
• False Positive Rate: Ensures model doesn't over-flag neutral language.
Concept Drift: Occurs when the statistical properties of the data change over time (e.g., new
slang, shifting definitions of inclusivity). Monitoring Solution: Use active learning or continuous evaluation pipelines with periodic re-training using recent data.
Technical Deployment Challenge: Scalability – processing large volumes of job ads in real
time may require distributed computing or efficient model serving via APIs like FastAPI +
TorchServe.

Hospital AI System for Patient Readmission Risk Prediction

1. Problem Scope (5 points)

Problem Definition:Develop an AI system to predict the likelihood of patient readmission within 30 days of discharge to enable proactive interventions and improve patient outcomes while reducing healthcare costs.

Objectives:
- Primary: Achieve 85% accuracy in predicting 30-day readmission risk
- Secondary: Reduce overall readmission rates by 15% through early intervention
- Tertiary: Optimize resource allocation for discharge planning and follow-up care

Stakeholders:
- Primary: Patients (beneficiaries of improved care), physicians and nurses (end users), hospital administrators (decision makers)
- Secondary: Insurance companies (cost implications), regulatory bodies (compliance oversight), IT department (system integration)
- Tertiary: Family members (care coordination), community healthcare providers (continuity of care)

2. Data Strategy (10 points)

Proposed Data Sources:
- Electronic Health Records (EHRs): Diagnosis codes, medication history, lab results, vital signs, procedure codes
- Demographics: Age, gender, socioeconomic status, insurance type, geographic location
- Clinical Data: Length of stay, discharge disposition, comorbidities, severity scores
- Administrative Data: Previous admissions, emergency department visits, outpatient visits
- Social Determinants: Housing stability, transportation access, social support systems

Two Ethical Concerns:

1. Patient Privacy and Data Security: Risk of unauthorized access to sensitive medical information during data collection, storage, and model training. Personal health information could be exposed through data breaches or inadequate anonymization techniques.

2. Algorithmic Bias and Health Equity: The potential for the model to perpetuate existing healthcare disparities by unfairly targeting certain demographic groups (such

as race, socioeconomic status, or insurance type) with higher predicted readmission risks, leading to differential treatment or resource allocation.

Preprocessing Pipeline:
- Data Cleaning: Remove duplicates, handle missing values using median imputation for numerical and mode imputation for categorical variables
- Feature Engineering: Create derived features such as comorbidity scores, medication complexity index, days since last admission, and interaction terms between age and chronic conditions
- Normalization: Apply standard scaling to numerical features and one-hot encoding to categorical variables
- Feature Selection: Use correlation analysis and recursive feature elimination to identify the most predictive variables
- Data Splitting: Implement stratified sampling to ensure balanced representation across risk categories in training, validation, and test sets

3. Model Development (10 points)

Selected Model: Random Forest Classifier

Justification:
Random Forest is optimal for this healthcare application because it handles mixed data types effectively, provides feature importance rankings for clinical interpretation, reduces overfitting through ensemble methods, and offers robust performance with missing values. It also provides probability estimates for risk stratification and maintains good interpretability for healthcare professionals.

Performance Calculations:
- Precision: 25/ (25+50) = 0.33 or 33%
- Recall (Sensitivity): 25/ (25+75) = 0.25 or 25%
- Specificity: 850/ (850+50) = 0.94 or 94%
- Overall Accuracy: (850+25)/ (850+50+75+25) = 0.875 or 87.5%

4. Deployment (10 points)

Integration Steps:
1. System Architecture Design: Develop RESTful API endpoints for model inference integrated with existing hospital information systems
2. Real-time Data Pipeline: Establish automated data extraction from EHR systems with real-time preprocessing capabilities
3. User Interface Development: Create a dashboard for clinicians showing risk scores, contributing factors, and recommended interventions
4. Pilot Testing: Implement controlled rollout in one department before hospital-wide deployment
5. Staff Training: Conduct comprehensive training sessions for healthcare providers on system interpretation and workflow integration

6. Monitoring Framework: Establish continuous performance monitoring with automated alerts for model drift or system failures

HIPAA Compliance Measures:
- Data Encryption: Implement end-to-end encryption for all data transmission and storage using AES-256 standards
- Access Controls: Deploy role-based authentication with multi-factor verification and audit logging of all system access
- De-identification: Apply safe harbor method for removing direct identifiers and statistical disclosure control techniques
- Business Associate Agreements: Establish formal agreements with all third-party vendors handling protected health information
- Regular Audits: Conduct quarterly security assessments and annual compliance reviews with documentation of all procedures
- Incident Response: Maintain comprehensive breach notification procedures meeting 60-day reporting requirements

5. Optimization (5 points)

Method to Address Overfitting: Cross-Validation with Regularization

Implement k-fold cross-validation (k=5) combined with hyperparameter tuning to optimize model complexity. This approach involves systematically varying the number of trees, maximum depth, and minimum samples per leaf while monitoring performance across multiple validation folds. Additionally, incorporate early stopping mechanisms that halt training when validation performance plateaus, preventing the model from memorizing training data patterns that don't generalize to new patients. This method ensures robust performance across diverse patient populations while maintaining clinical relevance and interpretability.

Part 3 – Critical Thinking

A. Ethical Considerations and Bias in Healthcare AI (10 marks)

Artificial Intelligence (AI) in healthcare brings massive opportunities for improving diagnosis, treatment, and patient outcomes. However, with these advancements comes a great responsibility to ensure that AI systems are ethical, fair, and safe. In this section, I reflect on some of the major ethical challenges and how they are addressed in the context of building a hospital readmission prediction system.

1.Bias in Training Data

Bias in AI often begins with biased data. In healthcare, this could mean underrepresentation of certain groups—such as rural patients, women, elderly individuals, or low-income populations. If the model is trained primarily on data from urban hospitals or private facilities, it may perform poorly for patients from public or rural hospitals. This creates inequality in predictions and outcomes.

For example, if the dataset used for training consists mainly of patients between 25–45 years old, the model may fail to accurately predict readmission risks for older patients or those with chronic conditions. Such bias can lead to harmful consequences, including misdiagnosis, inappropriate discharge timing, or neglect of high-risk patients.

2.Explainability and Trust

Healthcare professionals must trust AI tools to use them in real practice. This trust cannot be built if the model behaves like a "black box." Clinicians need to understand why the AI made a certain prediction, especially when it influences critical decisions like discharge timing or resource allocation.

To ensure explainability, we applied tools such as:
- SHAP (Shapley, Additive Explanations): to visualize how individual features contribute to predictions.
- LIME (Local Interpretable Model-Agnostic Explanations): to explain single-instance predictions in an interpretable way.

These tools help translate complex model outputs into actionable insights for medical staff.

3. Privacy, Consent, and Legal Compliance

Handling patient data demands strict privacy controls. Any AI model trained on Electronic Health Records (EHRs) or patient demographics must adhere to national and international privacy regulations.

In Kenya, the Data Protection Act (2019 governs how personal data must be collected, processed, stored, and shared. Additionally, globally recognized frameworks such as HIPAA (Health Insurance Portability and Accountability Act) provide guidance on protecting health data.

To ensure compliance:
- All datasets used in this project were anonymized.
- Personally identifiable information (PII) was removed during preprocessing.
- The data pipeline was designed to prevent leakage or exposure of sensitive information.

4. Fairness and Responsible AI Practices

To build a fair AI system:
- We used a balanced dataset with representation across gender, age, and geographical location.

- We monitored fairness metrics during evaluation to check for potential disparities.
- We documented model behavior and evaluation results to enable accountability and transparency.

Summary

Ethics is not a side note in AI—it is core to deploying responsible healthcare systems. The success of AI in hospitals depends not only on accuracy but also on fairness, trust, explainability, and compliance with law and human values.

B. Trade-offs in AI Model Development and Deployment (10 marks)

Every AI project requires making decisions between competing priorities. In a healthcare application such as hospital readmission prediction, these trade-offs must be made carefully to avoid compromising patient safety, model performance, or usability in the clinical environment.

1. Accuracy vs Interpretability

Highly accurate models like XGBoost, neural networks, or ensemble methods often offer superior performance on complex data. However, these models tend to be difficult to interpret. In contrast, models such as ogistic regression, decision trees, or Naive Bayes are easier to explain but may underperform on some metrics.

In our case, we prioritized a balance between performance and transparency. Random Forest was chosen for its relatively high accuracy and moderate interpretability. Feature importance plots allowed us to explain model behavior to non-technical users.

2. Precision vs Recall

In hospital readmission prediction, recall is more important than precision. We want the model to capture as many high-risk patients as possible—even if it flags some low-risk patients incorrectly. A false negative (failing to identify a high-risk patient) can lead to a critical patient being discharged without proper follow-up.

Therefore, our evaluation focused on metrics such as:
- Recall (to reduce false negatives)
- F1-score (to balance precision and recall)
- AUC-ROC (to evaluate model performance across all thresholds)

3. Model Complexity vs Practical Deployment

A very complex model may offer small gains in accuracy but be hard to deploy in a real hospital system. Hospitals often have limited infrastructure and expect systems that integrate easily with existing workflows.

That’s why we prioritized a model that:
- Runs quickly
- Can be deployed via a REST API (e.g., Flask or FastAPI)
- Can be updated with new data without full retraining

This allows the hospital IT team to monitor performance, detect concept drift, and update the model in a cost-effective way.

4. Computation Cost vs Sustainability

Training deep learning models can be a resource-intensive process. In many Kenyan or low-resource hospital settings, using GPUs or cloud computing is not always feasible. We considered models that work well on standard hardware (CPU), and we optimized for speed without compromising essential performance.

Summary

Trade-offs are not weaknesses—they are strategic decisions that reflect the context and constraints of the real-world environment. In healthcare AI, we must always ask:
"Does this choice help us save lives, reduce harm, and improve care?"
If not, we must rethink it.

Part 4 – Reflection and Workflow

A. Reflection on the AI Workflow Project (5 marks)

Working on the hospital readmission prediction project has been a powerful learning experience. It allowed me to move beyond theoretical machine learning and apply AI concepts in a real-world, high-impact setting. I realized that successful AI systems are not only defined by their technical accuracy but also by their ability to function ethically, transparently, and sustainably.

One of the key challenges I encountered was understanding how to address fairness and bias practically. Reading about bias in models is one thing, but detecting and mitigating it in actual datasets is much harder. I learned to ask critical questions: Who is represented in the data? Who might be left out? How do model predictions vary across groups?

Another key takeaway was the importance of explainability. I had to think deeply about how to help non-technical healthcare staff understand what the model is doing and why. Tools like SHAP and LIME were extremely useful in this regard, and I now consider them essential components of responsible AI development.

I also gained hands-on experience in model selection, metric interpretation (especially precision vs recall trade-offs), and designing a clean deployment plan using APIs. I now better understand the full AI lifecycle—from data collection and preprocessing, all the way to deployment and monitoring.

If I were to do this project again, I would engage domain experts (such as nurses or doctors) earlier in the process. Their insights could improve the feature selection and interpretation of results. Overall, this project has strengthened my technical skills, broadened my ethical understanding, and prepared me to build human-centered AI systems.

B. AI Workflow Diagram Description (5 marks)

The hospital readmission prediction system was developed using a clear AI workflow that followed best practices in machine learning and software development. Below is a detailed explanation of each stage in the workflow diagram.

1. Problem Definition
We started by clearly defining the problem: *predicting whether a patient will be readmitted within 30 days of discharge*. The objectives were to reduce preventable readmissions, optimize resource use, and improve patient care outcomes.

2. Data Collection
We collected data from electronic health records (EHRs), which included patient demographics, diagnosis codes, hospital stay durations, comorbidities, and historical readmission records.

3. Data Preprocessing
This phase involved:
- Handling missing values and inconsistent records
- Encoding categorical variables
- Normalizing numerical features
- Anonymizing sensitive data
- Checking for bias across demographics

4. Model Training & Selection
We experimented with several models including:
- Logistic Regression (baseline)
- Random Forest (final choice)
- XGBoost (high performance but lower explainability)

We selected Random Forest due to its balance between accuracy and interpretability. Cross-validation and hyperparameter tuning were used to optimize the model.

. Model Evaluation
Key evaluation metrics included:

-Recall: prioritized to catch high-risk patients
- F1-Score: balance between precision and recall
- AUC-ROC: assess model discrimination

We also used SHAP plots to visualize the top contributing features in predictions.

6. Deployment
The model was wrapped into a Flask AP to simulate real-world integration into a hospital's IT system. This allowed healthcare staff to input patient data and receive readmission risk predictions in real-time.

7. Monitoring and Feedback
Once deployed, the system requires regular monitoring:
- Track model performance over time
- Detect concept drift using periodic validation
- Schedule retraining with new patient data every 3–6 months
- Collect feedback from healthcare users to improve usability

The visual version of this workflow is shown in the `workflow_diagram.png` file submitted alongside this document.