1. Problem Definition (6 Points)

**Hypothetical AI Problem:**   
Predicting dropout rates among university students using a combination of academic, behavioural, and socioeconomic data.

**Problem Background:**   
Student retention is a key challenge for educational institutions. Dropouts negatively impact institutional rankings and student futures. Many universities lack predictive systems that can flag at-risk students early enough to intervene.

**Objectives:**

1. **Early Identification:** Detect potential dropouts before critical decision points.
2. **Causal Analysis:** Uncover the most impactful features contributing to attrition.
3. **Improve Retention Strategy:** Use data insights to shape personalized interventions and support systems.

**Stakeholders:**

* **University Administrators:** Responsible for student welfare, retention policies, and institutional success.
* **Students:** Directly impacted by targeted support, guidance, and institutional planning.

**KPI – Key Performance Indicator:**  
✅ **F1 Score**: An ideal metric for dropout prediction since it balances the risk of false positives (students incorrectly flagged as at-risk) and false negatives (missed predictions). Precision and recall together offer a nuanced performance measure.

2. Data Collection & Pre-processing (8 Points)

**Data Sources and Types:**

1. 📋 **Student Information System (SIS):** Contains historical data such as grades, attendance, participation in extracurricular activities, and disciplinary records.
2. 💬 **Student Surveys and Psychometric Assessments:** Self-reported engagement levels, satisfaction, mental health indicators, and financial pressures.

**Potential Data Bias:**  
 **Socioeconomic Bias:** Historical records might show students from low-income families as disproportionately prone to dropping out. If not properly mitigated, this can lead to unethical profiling or unfair prioritization.

**Data Pre-processing Steps:**

1. **Missing Data Handling:**
   * Use mean or median imputation for numerical data.
   * Apply domain-specific rules or predictive models for categorical data.
2. **Normalization / Scaling:**
   * Standardize numerical features (e.g., GPA, attendance percentage) using Min-Max or Z-score normalization for better model convergence.
3. **Categorical Encoding:**

* Use one-hot encoding for nominal categorical features (e.g., “Program of Study”).
* Apply ordinal encoding where a meaningful order exists (e.g., satisfaction levels: low → high).

3. Model Development (8 Points)

**Model Selected:**  
 **Random Forest Classifier**

**Justification for Model Choice:**

* Can handle large feature spaces and mixed data types (numerical + categorical).
* Robust to outliers and noisy data.
* Ensemble method reduces risk of over fitting.
* Offers feature importance scores, aiding explain ability.

**Data Splitting Strategy:**

* **Training Set (70%)** – Used for initial learning of patterns.
* **Validation Set (15%)** – Used for hyper parameter tuning and model selection.
* **Test Set (15%)** – Used for evaluating generalization performance and final metrics.

**Hyper parameters to Tune:**

1. n\_estimators:
   * Number of trees.
   * More trees usually improve accuracy but increase computational cost.
2. max\_depth:

* Limits how deep each tree can grow.
* Prevents over fitting and helps balance precision/recall.

4. Evaluation & Deployment (8 Points)

**Evaluation Metrics:**

1. **F1 Score:** Essential for balancing sensitivity (recall) and precision in predicting dropout risk. Reduces overreaction to false positives.
2. **ROC-AUC Score:** Assesses the model’s ability to separate dropout vs. non-dropout cases regardless of threshold. AUC above 0.80 suggests strong classification performance.

**Concept Drift:**   
**Definition:** Changes in the statistical properties of input data over time which reduce model accuracy. For example, new programs or shifting student demographics may change dropout patterns.

**Monitoring Concept Drift:**

* Deploy real-time evaluation dashboards.
* Periodically retrain using newer data.
* Use online drift detection tools (e.g., ADWIN, Page-Hinkley) to trigger automatic alerts for retraining.

**Deployment Challenge – Scalability:**  
**Definition:** The system's ability to scale and maintain performance under increased load, such as being used across multiple departments or campuses.

**Solutions to Scalability:**

* **Containerization:** Use Docker to package the model with all dependencies.
* **Orchestration:** Use Kubernetes to scale deployment across cloud servers.
* **APIs:** Build RESTful APIs for easy integration with SIS dashboards.
* **Cloud Hosting:** Utilize scalable platforms like AWS Lambda, Azure ML, or Google Cloud Run for cost-effective infrastructure.

Ethical Considerations

* **Bias Correction:** Apply re-weighting techniques, fairness-aware algorithms, or data audits.
* **Interpretability:** Provide transparency using feature importance plots or model explanations (e.g., SHAP or LIME).
* **Student Privacy:** Protect sensitive data through encryption, anonymization, and robust access controls.
* **Consent & Transparency:** Inform students how their data is used and give them the option to opt-in/out.