🎯 Phase 1: Problem Definition & Planning

This document serves as the foundational agreement for the **Student Dropout Risk Prediction** machine learning project. It clearly defines the business objective, the technical ML approach, and the criteria for success.

1. Business Understanding

The core objective of this project is to create an early warning system that allows academic advisors and student support staff to proactively engage with students at high risk of dropping out.

Element	Description
Primary Business Goal	**Reduce the student dropout rate by 20%** within the next academic year.
Rationale	Dropout leads to significant revenue loss, administrative overhead, and reduced institutional reputation. Early, targeted intervention is the most cost-effective solution.
Intervention Trigger	A student flagged with a high risk score (above a dynamic threshold, TBD during evaluation) will trigger a personalized outreach campaign.

Project Assumptions

This project operates under the following key assumptions regarding data and model influence:

- **Predictive Features:** We assume that dropout risk is directly and significantly influenced by institutional factors captured in the data (past grades, attendance, engagement metrics).
- **Non-Factors:** We assume that non-academic personal factors outside institutional data (e.g., family illness, personal mental health status) are **not** required as direct input features, though they may indirectly influence the data we observe (e.g., lower attendance).

- **Data Quality:** We assume that historical institutional data is recorded accurately and consistently across cohorts used for training.

2. Machine Learning Problem Definition

The problem is framed as a supervised learning task designed to predict a binary outcome based on a student's profile and historical engagement data.

Element	Specification
Problem Type	**Binary Classification**
Target Variable (Y)	Dropout (1) or Continue (0).
Input Features (X)	A mixture of categorical and continuous variables including Demographics (Age, Income), Academics (Grades, Attendance), and Engagement metrics.
Key Output	A **Probability Score** (ranging from 0.0 to 1.0) representing the likelihood of dropout, which will be converted to a binary risk flag.
Model Choice (Initial)	LightGBM Classifier (selected for its speed, handling of diverse feature types, and built-in feature importance for interpretability).

3. Success Metrics and Thresholds

Given the need to identify the *minority* (at-risk students) and the high cost of a false negative (failing to identify an at-risk student), we prioritize metrics that handle class imbalance effectively.

Metric	Rationale	Target Threshold	Business Translation
Primary Metric: ROC-AUC	Measures the model's ability to discriminate between positive	**> 0.85**	Ensures that **>85%** of high-risk students can be correctly prioritized

	(dropout) and negative (continue) classes across all thresholds.		for intervention.
Secondary Metric: F1-Score	Represents the harmonic mean of precision and recall. Essential for balancing false positives (unnecessary intervention) and false negatives (missed risk).	**Optimized based on business need** (e.g., maximizing F1 for a specific recall level).	Provides a single score balancing the efficiency and effectiveness of the intervention program.
Cost Matrix Consideration	The cost of a **False Negative** (missing an at-risk student) is considered significantly higher than a **False Positive** (intervening unnecessarily).	The threshold will be tuned to achieve high Recall, subject to acceptable Precision limits.	Ensures resources are concentrated on reducing missed cases, aligning with the primary goal of retention.

4. Constraints and Requirements

Constraint Category	Detail	Tracking & Documentation
Interpretability	**High Requirement:** Predictions must be traceable back to specific features (e.g., low attendance, poor grade in X course) to enable meaningful intervention strategies.	Documented in docs/constraints.md. Will use SHAP/LIME for explanation.
Data Privacy	All personally identifiable	Legal review documented

	information (PII) must be anonymized or pseudonymized during preprocessing and storage. Adherence to institutional data governance policies is mandatory.	in a project log.
Model Latency	The prediction API must provide a response within **100ms** to support real-time application usage (e.g., student support dashboard lookups).	Monitored post-deployment via **AWS CloudWatch**.
Model Throughput	The API must be capable of handling an **expected throughput of 1000 API calls/minute** for batch processing and dashboard updates.	Monitored post-deployment via **AWS CloudWatch/Prometheus**.
Data Availability	**Initial training** relies on data derived from the UCI Student Performance dataset. **Scaling to production** requires integration with the institutional data warehouse. This requires validation of schema differences, handling increased feature volume, and managing potential missing values in the institutional data source.	Data source and schema tracked in data/schema.json.

5. Risk Assessment

Key risks that could impact project success and planned mitigations.

Risk	Description	Impact	Mitigation Plan
Data Drift	Student behavior or curriculum changes (e.g., a rapid shift to fully online classes) cause training data distributions to become obsolete.	Reduced model accuracy and intervention effectiveness over time.	Implement automated monitoring (src/monitoring/) to detect distribution shifts and trigger alerts for mandatory model retraining.
Model Bias	The model relies too heavily on proxy features correlated with socioeconomic status, leading to disproportionate flagging of certain demographic groups.	Ethical and legal risk; reduced trust from advisors and students.	Conduct fairness audits during evaluation (comparing metrics across demographic groups) and use feature importance analysis to identify and mitigate over-weighted features.
Deployment Complexity	Integrating the containerized API with institutional systems (e.g., authentication, data pipelines) proves more complex than anticipated.	Delayed launch and increased operational costs.	Prioritize deployment to a simple, isolated staging environment (e.g., AWS Elastic Beanstalk) first before integrating with core institutional services.

Tracking Tools Summary for Phase 1:

- **Project Goals:** Confluence/Notion (Business tracking).
- **ML Specs:** GitHub Wiki (Technical tracking).
- **Configuration:** configs/training_config.yaml (Stores target metrics and initial

hyperparameter ranges).

- **Documentation:** docs/constraints.md (Formalizes non-functional requirements).