Project Proposal – AI-Driven Financial Model to Help New Investors

# 1. Project Summary.

This project focuses on designing, training, and evaluating an AI model capable of providing personalized portfolio recommendations and natural-language investment explanations. The model is intended to serve as the core intelligence behind an AI-driven financial advisor system but will be developed and analyzed independently. It will aim at assisting first time investors with low to medium income in investment goals by giving safe, explainable portfolio recommendations.

The model will consist of two key components:

* A reinforcement learning agent that learns portfolio allocation strategies under user-defined goals and risk constraints.
* A large language model-based interface that converts technical outputs into user-friendly explanations.

An AI pipeline will be implemented as a local, containerized infrastructure. It will automate nightly data collection from public APIs, store time-series data in a structured SQL database, run GPU-accelerated training and serve outputs. The project will cover machine learning, distributed systems, software engineering, and ethics. Furthermore, it will address several technical challenges: safe reinforcement learning with financial constraints, prompt engineering for financial goal translation, and explainability in both structured output and natural language. The final deliverable will include a trained model, reproducible training scripts, and evaluation metrics that quantify financial suitability, user constraints, and beginner level understanding.

# 2. Weekly Timeline

Week 1 – Finalize project scope and objectives, set up project repository and development environment (Docker, GPU libraries, API keys). Begin background reading on safe RL and financial goal modeling.

Week 2 – Review RL in finance, LLMs in explainable AI, and user-aligned portfolio recommendation systems.

Week 3 – Design and implement data intake pipeline: pull historical market data, store in a local SQL database, and confirm schema.

Week 4 – Design the RL environment

Week 5 – Implement baseline RL agent, train on historical data, and watch allocation behavior without constraints.

Week 6 – Integrate safety constraints into the RL training loop using constrained RL techniques.

Week 7 – Develop evaluation framework.

Week 8 – Implement the LLM layer

Week 9 – Generate and refine natural-language explanations from model outputs

Week 10 – Compare model behavior with/without constraints and LLM integration. Visualize model decisions and explanations.

Week 11 – Organize training scripts, logging, configs, and model output tracking.

Week 12 – Finalize results and evaluate against the original goals: financial suitability, user constraints, and beginner comprehension.

Week 13 – Write and submit final report. Prepare presentation with visualizations and live demo.