# AA Engine Project — Objectives & Guardrails

## 1. Project Goal

Build a production-grade pipeline that:  
- Forecasts ETF returns with XGBoost (Part 1)  
- Translates forecasts into Black–Litterman investor views for portfolio optimization (Part 2)  
- Validates strategies via a realistic historical backtester with frictions and constraints (Part 3)  
  
This will serve as the foundation of a new mutual fund launch.

## 2. File Discipline

- Always work with the most recent version of each module.  
- If the assistant does not have a file, it will explicitly ask for it.  
- No assumptions that we have the latest code unless explicitly attached by the user.

## 3. Honesty Over Guesses

- If information is missing or unverifiable (even after searches), state so clearly.  
- Suggest ways to resolve the gap rather than guessing.  
- Avoid misleading or filler responses.

## 4. Production-Quality Standards

- Code must be clean, concise, and idiomatic (PEP 8 + type hints + docstrings).  
- Modular boundaries enforced: data/, features/, models/, portfolio/, backtest/, utils/.  
- Config-driven and reproducible (deterministic seeds, run manifests).  
- Unit and integration tests for all critical functionality.  
- Readability and extensibility prioritized: minimal cleverness, maximum clarity.  
- No ad-hoc hacks in core package — prototypes live in experiments/ and are refactored later.

## 5. Workflow Discipline

- Each part (forecaster, optimizer, backtester) will be built and tested step by step.  
- No moving on until acceptance criteria are met for the current part.  
- Continuous iteration and review: code must remain maintainable as the system scales.

## Usage Note

This document should be used as a quick-handover whenever starting a new chat with ChatGPT. Simply copy and paste the contents so the assistant is aligned with objectives and guardrails.