

## Week 1

1. Initialised a FastAPI project with a production-ready structure, Docker setup, a Python 3.11 environment, pre-commit hooks (black, flake8, mypy), and complete onboarding documentation.
2. Configured PostgreSQL 15 with SQLAlchemy + Alembic migrations, implemented core schemas (orders, payments, pixel events, webhooks, reconciliation), added an indexing strategy, and documented a backup plan.
3. Set up Redis 7 caching layer with connection pooling, TTL strategies, cache utilities, and invalidation logic for VPA mappings, product data, and customer sessions.
4. Established async processing stack with RabbitMQ queues (webhook, reconciliation, notifications, cart recovery), dead-letter handling, retry mechanisms, and Celery workers + Beat scheduler.
5. Implemented structured JSON logging, correlation IDs, request/response middleware, performance metrics tracking, and monitoring via Celery Flower and alert rules.
6. Built environment-based configuration management using Pydantic Settings with validated startup configs for DB, Redis, Shopify, WhatsApp, payments, and security credentials.
7. Hardened platform security with API gateway (rate limiting PROD-only, API key auth, CORS, compression), global exception framework, webhook signature verification (HMAC), encryption, and secure headers enforcement.

**Supervisor's Comments and Signature (with date & time):**