## 🏗️ AIMaven Backend Engineering + AI Integration Internship Program

### 🗓️ One-Month Training Syllabus (Day-by-Day Detail)

### **Week 1: Python Foundations & Development Workflows**

**Day 1: Python Refresher (Functions, Lists, Dicts, OOP)**  
Learn or revise Python fundamentals including data types, loops, conditionals, functions, and class-based design. Focus on writing clean, reusable code.

**Tasks:** - Build a class-based invoice calculator (with line items, tax, total) - Practice on HackerRank/LeetCode (easy problems)

**Day 2: Error Handling, File I/O, Logging, Decorators**  
Understand exception handling in Python, work with files (CSV, JSON), and learn the importance of decorators and logging in real-world apps.

**Tasks:** - Build a custom logger that saves events to a file - Use decorators to time function execution

**Day 3: Virtualenv, Linting, Structuring Projects**  
Set up isolated Python environments using venv, follow PEP8 coding style, and organize projects with a modular structure.

**Tasks:** - Initialize a virtual environment - Create a simple project with src/, tests/, requirements.txt - Auto-format using black / ruff

**Day 4: Git & GitHub: Branching, PRs, Rebase**  
Learn version control fundamentals: making branches, committing, resolving merge conflicts, and opening pull requests.

**Tasks:** - Fork a repo, raise a PR, and resolve a merge conflict - Learn Git best practices and commit message guidelines

**Day 5: Python CLI with argparse**  
Understand how to build command-line tools using argparse and structure simple CLIs for utility scripts.

**Tasks:** - Build a CLI-based folder organizer that moves files by extension

**🧪 Mini Project:** CLI Expense Tracker with save/load support (JSON or CSV)

### **Week 2: FastAPI + Postgres + SQLAlchemy**

**Day 6: FastAPI Basics, Routing, Pydantic**  
Learn FastAPI basics including how to build routes, use query/path params, and perform request/response validation using Pydantic.

**Tasks:** - Create a basic Hello API with input validation - Explore Swagger UI

**Day 7: CRUD APIs + SQLite (SQLAlchemy / SQLModel)**  
Build RESTful APIs that connect to SQLite and perform full CRUD operations with an ORM.

**Tasks:** - Create a basic blog or task manager API (add/edit/delete tasks)

**Day 8: JWT Auth, Hashing Passwords, Middleware**  
Implement authentication using JWT tokens, secure password storage with hashing, and build login-protected endpoints.

**Tasks:** - Login/Register endpoints - Protected route for getting user profile

**Day 9: PostgreSQL + Alembic Migrations**  
Migrate your local DB to Postgres, create migration scripts using Alembic, and connect to a real database.

**Tasks:** - Switch from SQLite to Postgres - Run Alembic migrations

**Day 10: API Response Models, Error Handling, REST Standards**  
Standardize API outputs, handle exceptions properly, and follow best practices for designing APIs.

**Tasks:** - Add 422, 404, 401 error responses - Use response\_model for all endpoints

**🧪 Mini Project:** Task API with login system, CRUD operations, Postgres DB

### **Week 3: Django + Redis + Background Jobs**

**Day 11: Django Basics: Models, Admin, Templates**  
Build your first Django app. Learn models, migrations, admin panel, and how Django’s MVC architecture works.

**Tasks:** - Create a contact manager app - Enable it in Django Admin

**Day 12: Django REST Framework (DRF)**  
Expose your Django models via REST APIs using DRF and add serializers and viewsets.

**Tasks:** - Create API to list/create contacts - Test with Postman

**Day 13: Celery + Redis for Background Tasks**  
Setup asynchronous task processing using Celery and Redis. Useful for sending emails, reports, etc.

**Tasks:** - Send an email in the background after form submission

**Day 14: Redis Caching & Rate Limiting**  
Use Redis to cache expensive queries and control abuse with rate limiting.

**Tasks:** - Add Redis caching to contact list API

**Day 15: Testing with Pytest + DRF TestClient**  
Write tests for your APIs to ensure correctness and coverage.

**Tasks:** - Write unit tests for CRUD endpoints - Setup GitHub Actions to run tests

**🧪 Mini Project:** Django REST Job Board API with Celery email alerts

### **Week 4: AI Integration + Deployment**

**Day 16: Generative AI + Prompt Engineering 101**  
Understand basics of LLMs (OpenAI, Gemini), token usage, and prompt design to control outputs.

**Tasks:** - Call OpenAI/Gemini API using requests - Try completion and chat APIs

**Day 17: LangChain Basics: Prompt Templates + Tooling**  
Build your first AI tool using LangChain. Learn prompt chaining, memory, and tool integration.

**Tasks:** - Create a LangChain agent that answers product FAQs

**Day 18: Qdrant + Embeddings + Retrieval**  
Use SentenceTransformer to embed text data and store in Qdrant. Enable document-based search.

**Tasks:** - Index a PDF or Markdown file - Search and retrieve relevant chunks

**Day 19: Docker + Cloud Deployment**  
Learn Dockerfile, build images, and deploy your project using GCP Cloud Run.

**Tasks:** - Dockerize LangChain chatbot - Push to Cloud Run

**Day 20: Final Capstone Build & Demo Prep**  
Refine and finalize your team project. Prepare for demo with code review and deployment.

**Capstone Project:** - AI-enabled FAQ/chatbot API with FastAPI - Postgres + Qdrant DBs - LangChain integration - Dockerized & deployed

## 🔚 Final Outputs & Evaluation

* Weekly projects pushed to GitHub
* Final project demo day
* Peer code review
* Certificate of completion
* Opportunity for pre-placement offers (PPO)

Let’s build something awesome together. 🚀