

Veterans In Arms

Artificial Intelligence Boot Camp (AIBC)

FIELD MANUAL

Version: 2025-11-26

Table of Contents

1. Mission & Intent
 2. How to Use This Manual
 3. Course Structure & Expectations
 4. Week 1 – Data Foundations (SQL + Supabase)
 5. Week 2 – GPT Operations (LLMs & Function Calling)
 6. Week 3 – Automation with n8n
 7. Week 4 – Frontend Basics (Replit / Next.js)
 8. Week 5 – Environments & Secrets
 9. Week 6 – Monitoring & Reliability
 10. Week 7 – Analytics & Feedback Loops
 11. Week 8 – Advanced Agents & Tools
 12. Week 9 – Local LLMs
 13. Week 10 – Security & Access Control
 14. Week 11 – Content Systems
 15. Week 12 – Market & Career Alignment
 16. Week 13 – Capstone Week
 17. Bonus Week 14 – VIA Video Ad (Nano Banana Workflow)
 18. Appendix A – Tech Stack Quick Reference
 19. Appendix B – Glossary
-

1. Mission & Intent

Mission:

AIBC transforms transitioning service members into capable AI operators in 13 weeks. You will learn to design, build, deploy, and operate AI-enabled systems that solve real problems in the civilian world.

Intent:

By the end of this course, you will:

- Understand core AI concepts (LLMs, prompts, function calling).
- Know how to connect AI to databases, workflows, and frontends.
- Be able to ship small but real tools (like VIA) that others can use.
- Translate this capability into civilian job language and portfolios.

This is not an academic degree. This is a **technical MOS** forged for the AI era.

2. How to Use This Manual

- Treat each week as a **training block**.
- At the start of every week, read the corresponding section.
- During the week, work through the skills and build the **weekly deliverable**.
- At the end of the week, do a quick self-debrief:
 - What did I build?
 - What still confuses me?
 - What should I review on the weekend?

When in doubt: build something small and working rather than something big and theoretical.

3. Course Structure & Expectations

Duration: 13 core weeks + optional bonus week.

Pace: 5–10 hours per week minimum.

Style: Hands-on, project-driven, self-paced.

Each week includes:

- **Focus:** What this week is about.
- **Skills:** Specific abilities you will practice.
- **Deliverable:** A concrete artifact you produce.

If you miss a week, do not quit. Compress, simplify, and move forward. Progress beats perfection.

4. Week 1 – Data Foundations (SQL + Supabase)

Focus:

Understand how data is stored, queried, and structured so you can power AI tools with real information instead of hard-coded examples.

Skills:

- Tables, rows, primary keys, and relationships.
- Basic **SELECT**, **INSERT**, **UPDATE**, **DELETE**.
- Simple **JOINS** between related tables.
- Using Supabase as a hosted Postgres backend.

Deliverable:

Create a Supabase project with at least two related tables:

- Example: **veterans** and **claims**.
 - Insert at least 10 example records.
 - Prove to yourself you can query “all claims for a given veteran” using SQL.
-

5. Week 2 – GPT Operations (LLMs & Function Calling)

Focus:

Learn how GPT-style models work and how to call them from code, including sending and receiving structured data.

Skills:

- Designing prompts for clarity and structure.
- Understanding system / user / assistant roles.
- Calling GPT via API from a simple script or endpoint.
- Using function calling to request JSON output.

Deliverable:

Build a small script or API endpoint that:

- Accepts a short text description (e.g., "service story").
- Sends it to GPT with a function call definition.
- Receives back structured JSON (e.g., `{ issue_type, severity, service_connection_guess }`).

Save both the prompt and JSON response for reference.

6. Week 3 – Automation with n8n

Focus:

Turn individual actions into automated pipelines using triggers, branches, and error handling.

Skills:

- Setting up n8n (cloud or self-hosted).
- Creating HTTP, webhook, or schedule triggers.
- Connecting n8n to Supabase and GPT.
- Adding basic branching and error handling nodes.

Deliverable:

Build an n8n workflow that:

1. Triggers on a webhook or schedule.
2. Calls GPT with some input.
3. Writes results to Supabase.
4. Sends a notification (email, Discord, etc.) when complete.

Document this workflow with a screenshot and a short one-paragraph description.

7. Week 4 – Frontend Basics (Replit / Next.js)

Focus:

Build a simple but real web interface so non-technical users can interact with your AI tools.

Skills:

- React component fundamentals (props, state, basic events).

- Creating forms, text inputs, and buttons.
- Calling your own backend/API route from the frontend.
- Handling loading, success, and error states gracefully.

Deliverable:

Create a small web page that:

- Lets a user submit some input (e.g., description of service + symptoms).
- Sends that to your GPT backend.
- Displays the resulting structured data or draft text on screen.

This is your first "tiny VIA".

8. Week 5 – Environments & Secrets

Focus:

Ship without leaking keys. Treat configuration and secrets as first-class citizens.

Skills:

- Using environment variables for API keys and database URLs.
- Separating dev, staging, and production settings.
- Storing secrets in Replit / Vercel / n8n securely.
- Implementing a simple health-check route.

Deliverable:

- Deploy your app to a real URL.
 - Confirm all secrets are pulled from environment variables (not hard-coded).
 - Add a `/health` or `/status` endpoint that returns a simple "OK" JSON.
-

9. Week 6 – Monitoring & Reliability

Focus:

Make your system observable so you can debug issues instead of guessing.

Skills:

- Logging key events and errors.
- Viewing logs through your platform or log service.
- Adding user-friendly error messages in the UI.
- Setting up a basic uptime monitor on your main route.

Deliverable:

- Implement logging around your main workflow (start, success, failure).
 - Enable an uptime monitor hitting a key route every few minutes.
 - Prove you can identify at least one synthetic error in the logs.
-

10. Week 7 – Analytics & Feedback Loops

Focus:

Understand what users actually do, not what you hope they do.

Skills:

- Defining core events (e.g., visit → started → completed).
- Logging events into Supabase or an analytics tool.
- Reading simple funnels and identifying drop-offs.
- Designing one improvement based on real data.

Deliverable:

- Log at least three events for your app.
 - Create a simple report (even a SQL query) showing how many users reach each step.
 - Write a short “after action review” describing one change you’ll make.
-

11. Week 8 – Advanced Agents & Tools

Focus:

Move beyond single prompts into multi-step, tool-using AI systems.

Skills:

- Designing function/tool definitions for GPT.
- Letting the model choose which tool to call based on user input.
- Breaking big tasks into multiple steps with checkpoints.
- Caching or reusing intermediate outputs.

Deliverable:

- Implement an “agent” flow where GPT can call at least two tools (for example: `summarize_notes` and `query_claims_db`).
 - Demonstrate at least one example where the model chooses different tools depending on the user’s request.
-

12. Week 9 – Local LLMs

Focus:

Run models on your own hardware for experimentation, cost control, or privacy.

Skills:

- Installing Ollama or LM Studio.
- Downloading and running a local model.
- Prompting the local model for a specific task.
- Comparing speed and quality vs a hosted GPT-style model.

Deliverable:

- Create a short comparison document showing:
 - A sample prompt.
 - Response from a hosted model.
 - Response from a local model.
 - Note where the local model is “good enough” and where it struggles.
-

13. Week 10 – Security & Access Control

Focus:

Avoid embarrassing security mistakes when your tools touch real data.

Skills:

- Role-based access control (RBAC) basics.
- Using Supabase Row Level Security (RLS) or equivalent.
- Creating user roles (e.g., standard user vs admin).
- Safe prompt patterns that avoid echoing sensitive data.

Deliverable:

- Implement at least two roles in your app.
 - Protect one table with RLS so each user only sees their own rows.
 - Add a short “security notes” section to your README.
-

14. Week 11 – Content Systems

Focus:

Turn AI into a structured content engine instead of random one-off outputs.

Skills:

- Designing prompt templates for consistent tone.
- Generating multiple formats from one core idea (e.g., blog + email + checklist).
- Integrating with a simple CMS or knowledge base (Notion, Supabase, etc.).
- Reviewing and editing AI output before publishing.

Deliverable:

- Choose one topic (e.g., “Preparing for a C&P exam”).
 - Use your system to create at least three pieces:
 - A short article or guide.
 - An email draft.
 - A checklist or FAQ.
-

15. Week 12 – Market & Career Alignment

Focus:

Aim all of this capability at real opportunities.

Skills:

- Mapping your projects to job descriptions.
- Writing clear bullet points about your AI/automation skills.
- Creating a simple portfolio page or GitHub profile.
- Designing an outreach plan (emails, LinkedIn messages, etc.).

Deliverable:

- A one-page "career brief" summarising:
 - Your top 2–3 projects.
 - The tech you used.
 - The problems you solved.
 - A live portfolio or GitHub profile linking to these projects.
-

16. Week 13 – Capstone Week

Focus:

Pull everything together into a single deployed tool.

Skills:

- Scoping a realistic MVP (Minimum Viable Product).
- Choosing which features make it into v1.
- Polishing UX just enough to feel professional.
- Writing a README and simple user guide.

Deliverable:

- A deployed capstone project with:
 - Live URL.
 - Source code repo.
 - Short user guide (who it's for, what it does, how to use it).

This is what you show to hiring managers, potential clients, or collaborators.

17. Bonus Week 14 – VIA Video Ad (Nano Banana Workflow)

Focus:

Create a 30–45 second vertical ad promoting VIA (or your own project), using Nano Banana for imagery and a simple video editor.

Skills:

- Writing a short conversion-focused script (hook → pain → solution → CTA).
- Designing consistent characters and scenes with Nano Banana image prompts.
- Assembling a vertical video with motion, captions, and voice-over.
- Exporting in social-ready formats (TikTok, Reels, Shorts).

Deliverable:

- A finished ad video (1080×1920 MP4) with:
 - Clear message and CTA.
 - Legible on-screen text.
 - Either human or AI voice-over.
-

18. Appendix A – Tech Stack Quick Reference

Core Tools:

- **LLMs:** OpenAI GPT models (or equivalents), local LLMs via Ollama / LM Studio.
 - **Database:** Supabase (Postgres).
 - **Automation:** n8n.
 - **Frontend:** Replit / Next.js React apps.
 - **Deployment:** Replit, Vercel, or similar.
 - **Version Control:** Git + GitHub.
-

19. Appendix B – Glossary

- **LLM (Large Language Model):** A text model like GPT that predicts the next token and can be prompted to perform tasks.
 - **Function Calling / Tools:** A way to have an LLM decide when to call predefined functions/APIs and return structured JSON.
 - **Supabase:** Hosted Postgres database with auth and APIs.
 - **n8n:** Automation tool for connecting services and building workflows.
 - **Ollama / LM Studio:** Tools for running local LLMs on your own hardware.
 - **MVP:** Minimum Viable Product — the smallest version of a tool that still delivers value.
 - **RLS:** Row Level Security — database feature that restricts which rows a user can see.
-

End of Field Manual – AIBC 2025-11-26