AISOD PAIED Program

Complete Manual 2026

Practical AI Engineering & Development

Free AI Engineering Course

AISOD Institute | info@aisodinstitute.tech | +264 81 497 1482
paied.aisodx.tech | aisodx.tech | aisodinstitute.tech

A Word from the CEO – Joel Tiago

A Word from the CEO – Joel Tiago

Welcome to the Practical AI Engineering & Development (PAIED) Program!

Dear Future AI Builders,

As the CEO of AISOD, I am incredibly proud to welcome you to this transformative journey. In a world where AI is reshaping everything, Namibia and Africa must not just consume technology—we must create and own it. That's why we built the PAIED Program: to empower beginners like you with practical, hands-on skills to design, code, automate, and launch real AI-powered products in hours—not years.

Our philosophy is simple and powerful: "They may not believe the papers, but you will get paid for the working projects." Here, there are no endless exams or abstract theory. Your success is measured by what you build and launch—websites, mobile apps, agents, custom models—all using ethical AI tools and our preferred stack (Flutter/Dart, React Native, Next.js, Supabase, Python, HTML, CSS and JavaScript, LoRa, TensorFlow, Haggingface, GitHub, Langchain and etc,...).

Over these 9 months, you'll go from zero experience to shipping professional-grade solutions that solve real problems in Namibia—from business automations to innovative apps. And with AISOD X as your ecosystem, your projects can earn income, attract funding, and impact communities.

This is more than a program—it's your entry into Africa's AI revolution. We're here to support you every step: mentors, community, live sessions, and tools to make complex things simple.

Dream Big. Believe in yourself. Build boldly. Launch fearlessly.

The future is yours to code.

Joel Tiago
CEO, AISOD Institute
Windhoek, Namibia
December 31, 2025

AISOD PAIED Program Syllabus (2026 Edition)

AISOD PAIED Program Syllabus (2026 Edition)

Program Name: Practical AI Engineering & Development (PAIED) Program
Delivered by: AISOD Institute
Launch Year: 2026
Start Dates:

Non-workers cohort (full-time learners, students, youth): January 6, 2026
Workers cohort (part-time, professionals): February 7, 2026

Duration: 9 Months (flexible pacing: 10–15 hours/week for non-workers; 8–12 hours/week for workers)
Format: Online/Hybrid Vocational Program (self-paced modules, live webinars, mentorship, peer community)
Target Audience: Absolute beginners with no prior coding or design experience — youth, school learners, women, career changers, working professionals, and entrepreneurs in Namibia and beyond. Basic computer literacy required.

Program Philosophy: "They may not believe the papers, but you will get paid for the working projects. Therefore, Learn by Building and Launching functioning projects in hours."

Success is measured only by completed, publicly published, and functioning projects (e.g., live websites, mobile apps, AI agents, GitHub repositories, design prototypes) — not by exams or theory tests. We use modern AI tools to simplify complex topics, enabling beginners to ship real products fast and prove their skills in the marketplace.

Key Outcomes:
• Build and launch multiple real-world AI-powered products (websites, mobile apps, automations, agents).
• Gain hands-on mastery of in-demand tools and languages.
• Develop a strong public portfolio of launched projects that attracts freelance clients, employers, or investors.
• Become capable of earning income through freelancing, selling digital products, or launching startups.

Prerequisites: None — 100% beginner-friendly. AI tools bridge all gaps.
Ethical AI Commitment: We teach and enforce responsible, ethical use of AI throughout the program (bias awareness, data privacy, transparency, originality in work).

Cost Structure (2026):

School Learners (via AI Clubs): N$1,334 per learner (one-time or instalments)
Standard Individual: N$200 per month
Free Access: Included for all AISOD X Premium subscribers
Scholarships: Available for underserved groups, women in tech, rural youth, and high-potential applicants

Preferred Technology Stack (Core Focus):

Mobile & Cross-Platform: Flutter + Dart (primary)
Web Development: Next.js + React
Alternative Mobile: React Native
Backend & Database: Supabase
AI & Scripting: Python Other tools introduced as needed, but these are prioritized for simplicity, speed, and modern industry demand.

Assessment & Progression:
• Fully project-based.
Each module ends with a public launch milestone (e.g., live URL, GitHub repo, app demo).
• Mentors review using clear rubrics (functionality, usability, ethical AI use, documentation).
• Progression is gated: Complete and publish the milestone to unlock the next module.
• No exams, no theory quizzes — real shipped work is the only proof.

Program Structure Overview (9 Months)

Month	Module Title	Core Focus	Primary Tools	Key Skills Gained
1	AI Essentials & Foundations	AI concepts, prompting, ethical use	Grok/Claude, VS Code, GitHub	Prompt engineering, tool setup
2	UI/UX Design & Graphics for Developers	Professional visuals and prototypes	Figma, AI image tools (ethical use)	User-centered design, prototyping
3	AI-Powered Web Development	Responsive websites with AI features	Next.js, React, Supabase, Vercel	Full-stack web, deployment
4	AI-Powered Mobile Development	Cross-platform mobile apps	Flutter + Dart (primary), Supabase	Mobile UI, state management, publishing
5	Core Programming Mastery with AI	Deep dive into key languages	Python, Dart, JavaScript/TypeScript	Clean code, APIs, debugging with AI
6	AI Agents & Automation Development	Autonomous agents and workflows	Python, LangChain, Supabase Functions	Agentic AI, multi-step automation
7	Model Fine-Tuning & Customization	Adapting LLMs ethically	Python, Hugging Face,

LoRA | Custom models, domain adaptation |
| 8 | Advanced AI Integrations | Real-world connections (MCP, APIs) | Supabase, APIs, MCP protocols | Secure integrations, scaling |
| 9 | Capstone Projects, Portfolio & Career Launch | Full products + professional launch | All tools + freelancing platforms | End-to-end builds, pitching, earning |

Detailed Module Breakdown

Month 1: AI Essentials & Foundations

Objectives: Build confidence in AI; master ethical prompting; set up professional environment.
Weekly Breakdown:
• Week 1: What AI can do in 2026; agentic AI trends; Namibia opportunities.
• Week 2: Advanced prompt engineering (chain-of-thought, role-playing).
• Week 3: Ethical AI use (bias, privacy, plagiarism avoidance).
• Week 4: Dev environment setup (VS Code, GitHub, Supabase account).
Milestone Project: Publish a GitHub repo with 10 powerful, reusable prompts + short demo video.

Month 2: UI/UX Design & Graphics for Developers

Objectives: Create beautiful, user-friendly interfaces using ethical AI assistance.
Weekly Breakdown:
• Week 1: UI/UX principles, user flows, accessibility.
• Week 2: Ethical AI graphics (prompting for original assets).
• Week 3: Figma prototyping (interactive mocks).
• Week 4: Design systems and iteration.
Milestone Project: Publish 3 complete app/web prototypes on Figma Community (used in Months 3–4).

Month 3: AI-Powered Web Development

Objectives: Build and deploy modern websites using Month 2 designs.
Weekly Breakdown:
• Week 1: Next.js + React fundamentals.
• Week 2: Styling, responsiveness, Supabase integration.
• Week 3: AI features (chatbots, recommendations).
• Week 4: Deployment (Vercel) and optimization.
Milestone Project: Launch a fully functional website (live URL) with Supabase backend.

Month 4: AI-Powered Mobile Development

Objectives: Build cross-platform mobile apps with Flutter/Dart.
Weekly Breakdown:
• Week 1: Flutter + Dart setup and basics.
• Week 2: Implementing designs from Month 2.
• Week 3: State management, Supabase auth/data.
• Week 4: Testing and publishing (APK/web build).
Milestone Project: Publish a working mobile app (live web build or APK share).

Month 5: Core Programming Mastery with AI

Objectives: Gain fluency in Python, Dart, and JavaScript/TypeScript.
Weekly Breakdown:
• Week 1: Python mastery (functions, OOP, libraries).
• Week 2: Dart deep dive (async, null safety).
• Week 3: Advanced React/Next.js patterns.
• Week 4: APIs, error handling, AI-assisted debugging.
Milestone Project: Enhance a previous project (Month 3 or 4) with new backend features; relaunch.

Month 6: AI Agents & Automation Development

Objectives: Build autonomous agents that perform real tasks.
Weekly Breakdown:
• Week 1: Agent concepts and planning.
• Week 2: Simple Python agents (e.g., email/social automation).
• Week 3: Multi-tool agents with memory.
• Week 4: Deployment and real-world testing.
Milestone Project: Launch a public AI agent demo (GitHub + live example).

Month 7: Model Fine-Tuning & Customization

Objectives: Ethically customize LLMs for specific use cases.
Weekly Breakdown:
• Week 1: Pre-trained models and fine-tuning basics.
• Week 2: Ethical data preparation.
• Week 3: LoRA fine-tuning (Python + Hugging Face).
• Week 4: Evaluation and integration.
Milestone Project: Fine-tune a model (e.g., local language support) and integrate into a prior app.

Month 8: Advanced AI Integrations

Objectives: Connect systems securely (MCP servers, multi-agent).
Weekly Breakdown:
• Week 1: API design and security.
• Week 2: MCP protocol for agent-tool communication.
• Week 3: Supabase Edge Functions scaling.
• Week 4: Multi-modal and real-time systems.
Milestone Project: Build and deploy an integrated AI system (app + agents + MCP).

Month 9: Capstone Projects, Portfolio & Career Launch

Objectives: Ship production-grade products; prepare for income generation.
Weekly Breakdown:
• Week 1: Project ideation and planning.
• Week 2–3: Full development (individual or small team).
• Week 4: Portfolio site, freelancing strategy, client pitching.
Milestone Projects: Launch 2–3 complete products; build public portfolio site. Final showcase event.

Support & Community
• Weekly live webinars (recorded).
• Dedicated mentorship (book sessions).
• Peer community for collaboration.
• 24/7 AI study assistant access.
• Ethical AI guidelines reinforced every module.

PAIED Program Detailed Curriculum for Month 1 – AI Essentials & Foundations

PAIED Program: Detailed Curriculum for Month 1 – AI Essentials & Foundations

Month Overview This foundational month sets the stage for your PAIED journey by demystifying AI and equipping you with essential skills to build and launch functioning projects quickly. Aligned with our philosophy—"They may not believe the papers, but you will get paid for the working projects. Therefore: Learn by Building and Launching functioning projects in hours"—we emphasize ethical AI use from day one. You'll explore AI's potential in 2026, master prompting to generate code/designs ethically, address real-world ethical challenges, and set up a development environment focused on our preferred stack (Python, Supabase, Flutter/Dart React and Next.js).

No prior experience needed—AI tools like Grok or Claude will assist every step or the AISOD tutors. By month's end, you'll launch your first public project: a GitHub repo of ethical, reusable prompts with a demo video proving functionality.

Ethical AI Focus: All activities stress responsible use—e.g., citing AI-generated content, avoiding biased prompts, respecting privacy.

Resources Provided: Access free tools via AISOD X (if Premium) or open links. Weekly live webinars (1 hour, recorded) for Q\&A. Mentor office hours (book via platform). Community forums for peer support.

Weekly Effort: 10–12 hours (non-workers) or 8–10 hours (workers). Break it into 2–3 sessions/week.

Week 1: What AI Can Do in 2026 – Trends, Agentic AI, and Namibia Opportunities

Objectives: Understand AI's evolution by 2026; identify agentic AI trends; explore local applications for Namibia (e.g., agriculture, education, startups). Build excitement by launching a simple AI-assisted task.

Lessons & Topics (2–3 hours/day):
AI Basics & 2026 Landscape (Day 1–2): Define AI, ML, LLMs, and agentic systems (AI that plans/executes autonomously). Project 2026 trends: Multimodal AI (text+image+video), self-improving agents, edge AI for low-resource areas like Namibia.
Agentic AI Trends (Day 3): How agents automate workflows (e.g., business bots, data analysis). Real-world examples: AI in African agriculture (crop prediction) or education (personalized learning).

Namibia-Specific Opportunities (Day 4): AI for local challenges—e.g., water management, tourism apps, youth entrepreneurship. Tie to AISOD X ecosystem for earning.

Resources:

Free Course (EN): IBM's "AI for Everyone: Master the Basics" (2–3 hours; free certificate).
Free Course (PT): IBM "IA para Todos" - Versão em português disponível com legendas.
Video (EN): AI Explained - "What is Artificial Intelligence? In 5 minutes" - Introduction to AI concepts.
Video (PT): O que é Inteligência Artificial? - Conceitos básicos de IA.
Video (EN): Crash Course AI - "What is AI?" - Comprehensive AI overview.
Video (PT): Curso Rápido de IA - Visão geral completa de IA.
Video (EN): Google AI Essentials Course - Free course covering AI fundamentals.
Video (PT): Fundamentos de IA - Google - Curso gratuito de fundamentos.
Video (EN): Agentic AI Explained 2026 - Search for latest agentic AI trends.
Video (PT): IA Agêntica Explicada - Tendências de IA agêntica.
Reading (EN): DataCamp's "How to Learn AI From Scratch in 2026" guide (focus on basics section).
Video (EN): AI in Africa - Opportunities and Challenges - Context for Namibia applications.
Video (PT): IA na África - Oportunidades - Contexto para aplicações.

Hands-On Activities (3–4 hours):
• Experiment with free AI tools (e.g., Grok): Prompt for "2026 AI trends in Namibia agriculture" and summarize ethically (cite sources).
• Build/Launch Mini-Project: Use AI to generate a simple Namibia-focused idea (e.g., "AI bot for crop advice") and share in community forum.

Assignment: Journal 3 AI trends relevant to your goals; discuss ethical implications (e.g., job displacement in Namibia). Submit to mentor for feedback.

Week 2: Advanced Prompt Engineering

Objectives: Master prompting to create functioning outputs in hours (e.g., code snippets, ideas). Emphasize ethical techniques to avoid harm/misinformation.

Lessons & Topics (2–3 hours/day):
Prompt Basics to Advanced (Day 1–2): Zero-shot vs. few-shot; chain-of-thought (CoT) for reasoning; role-playing (e.g., "Act as a Namibian AI developer").
Techniques for 2026 (Day 3): Structured prompts (JSON outputs); iterative refinement; tool-calling for agents.
Ethical Prompting (Day 4): Avoid biased language; ensure originality; prompt for diverse

perspectives (e.g., "Include Namibia viewpoints").

Resources:

Video: Prompt Engineering Full Course - freeCodeCamp - Complete prompt engineering tutorial.
Video: Advanced Prompt Engineering Techniques - Search for latest techniques.
Video: Chain of Thought Prompting Explained - CoT methodology.
Tutorial: Lakera's Prompt Engineering Guide (step-by-step with examples).
Interactive Guide: Promptingguide.ai (free catalog; practice CoT).
Video: Zero-Shot vs Few-Shot Prompting - Comparison tutorial.
Ethical Add-On: IBM's Prompt Engineering resources (focus on responsible use).

Hands-On Activities (3–4 hours):
• Practice 10 prompts in Grok/Claude: E.g., "Generate ethical Python code for a simple Namibia weather app" (cite AI, check bias).
• Build/Launch Mini-Project: Prompt AI for a business idea; refine with CoT; post refined version in forum.

Assignment: Create 5 advanced prompts (e.g., role-playing for app ideas); test functionality; submit with ethical notes.

Week 3: Ethical AI Use

Objectives: Deep dive into responsible AI; apply to real scenarios; ensure all future projects prioritize ethics.

Lessons & Topics (2–3 hours/day):
Core Ethical Principles (Day 1–2): Bias detection, privacy (e.g., data handling in Supabase), transparency (citing AI use).
2026 Challenges (Day 3): AI in Namibia (e.g., cultural bias in models); global issues like deepfakes.
Practical Application (Day 4): Ethical checklists for prompting/coding; case studies (e.g., AI in education).

Resources:

Video: AI Ethics Explained - Crash Course - Introduction to AI ethics.
Video: Bias in AI Systems - Understanding and detecting bias.
Video: Privacy and AI - What You Need to Know - Data privacy in AI.
Guide: AI Ethics 101: A Beginner's Guide (practical tips).

Lesson Plan: PBS "AI Unlocked – Ethical Considerations" (free activities for beginners).
Video/Resource: Cornell's Ethical AI Course (interactive scenarios).
Video: Deepfakes and AI Misinformation - Understanding 2026 challenges.

Hands-On Activities (3–4 hours):
• Analyze a prompt for bias; rewrite ethically.
• Build/Launch Mini-Project: Ethical AI audit of a simple tool (e.g., prompt checker); share findings.

Assignment: Develop a personal ethical AI manifesto; apply to a sample project.

Week 4: Dev Environment Setup

Objectives: Configure tools for building; focus on preferred stack; launch a basic setup test.

Lessons & Topics (2–3 hours/day):
Core Setup (Day 1–2): Install VS Code, GitHub; Python/Supabase basics.
Flutter/Dart Integration (Day 3): Setup Flutter; connect to Supabase.
Ethical & Best Practices (Day 4): Secure configs; version control ethics.

Resources:

Video: VS Code Setup for Beginners - Complete VS Code installation and setup.
Video: Git and GitHub Tutorial for Beginners - Version control basics.
Video: Python Installation and Setup - Python environment setup.
Video: Supabase Setup Tutorial - Creating your first Supabase project.
Guide: Supabase Flutter Quickstart (step-by-step).
Video: How to Setup Supabase in Flutter - Complete integration tutorial.
Video: Flutter Installation Guide - Flutter SDK setup.
Docs: Supabase Environments Guide (for AI/Python).
Video: Next.js Setup Tutorial - React and Next.js installation.

Hands-On Activities (3–4 hours):
• Install and test stack; create Supabase project.
• Build/Launch Mini-Project: Simple Python script \+ Flutter hello world, deployed ethically.

Assignment: Document your setup in a GitHub README; include ethical notes (e.g., data privacy).

Month 1 Milestone Project

Project: Build a GitHub repo with 10 ethical, functioning prompts (e.g., for Namibia apps). Include a 2–3 min demo video showing real outputs. Launch publicly; mentor approves to unlock Month 2\. (4–6 hours total.) Rubric: Functionality (50%), Ethical Use (30%), Documentation (20%).

This curriculum ensures you launch something valuable in hours, proving skills for real pay. Questions? Post in the community\!

PAIED Program Detailed Curriculum for Month 2 – UI UX Design & Graphics for Developers

PAIED Program: Detailed Curriculum for Month 2 – UI/UX Design & Graphics for Developers

Month Overview This month shifts from AI theory to visual creation, teaching you to design professional, user-friendly interfaces and graphics that make your future apps/websites stand out. Following our philosophy—"They may not believe the papers, but you will get paid for the working projects"—you'll use ethical AI tools to generate assets quickly, then refine them manually for originality and quality. By the end, you'll launch 3 complete, interactive prototypes publicly on Figma Community—these become mandatory blueprints for your Month 3–4 coding projects (gated progression).

Focus on mobile-first design (our priority for Flutter/Dart apps), accessibility, and ethical practices (e.g., original compositions, bias-free visuals). No prior design skills needed—AI assists heavily, but you learn to own the output.

Ethical AI Focus: Always cite/modify AI-generated graphics; avoid copyrighted styles; ensure inclusivity (e.g., diverse representations).

Resources Provided: Free Figma account (start here: figma.com). Weekly live webinars (recorded). Mentor feedback. Community forum for sharing drafts.

Weekly Effort: 12 hours (non-workers) or 10 hours (workers). Split into daily 2-hour sessions.

Week 1: UI/UX Principles, User Flows, Accessibility, and Mobile-First Design

Objectives: Grasp core principles; learn mobile-first (design for small screens first); prioritize accessibility for real-world impact.

Lessons & Topics (2–3 hours/day):
UI vs. UX Basics (Day 1): UI (visuals) vs. UX (experience); hierarchy, balance, contrast.
Mobile-First & Responsive Principles (Day 2): Why start mobile (faster loading, Namibia's high mobile use); breakpoints, flexible grids.
User Flows & Wireframing (Day 3): Mapping user journeys (e.g., login !’ dashboard); low-fidelity sketches.
Accessibility Essentials (Day 4): Color contrast (WCAG standards), screen reader support, keyboard navigation; ethical inclusivity.

Resources:

Video (EN): UI vs UX Design - What's the Difference? - Clear explanation of UI/UX basics.
Video (PT): UI vs UX Design - Qual a Diferença? - Explicação clara dos conceitos básicos.
Video (EN): Mobile-First Design Principles - Complete mobile-first tutorial.
Video (PT): Design Mobile-First - Tutorial completo de design mobile-first.
Video (EN): Figma Tutorial for Beginners - Getting started with Figma.
Video (PT): Tutorial de Figma para Iniciantes - Começando com Figma.
Video (EN): Wireframing Tutorial - How to create wireframes.
Video (PT): Tutorial de Wireframe - Como criar wireframes.
Video (EN): Accessibility in Design - WCAG Basics - Accessibility essentials.
Video (PT): Acessibilidade em Design - Fundamentos de acessibilidade.
Video (EN): User Flow Mapping Tutorial - Creating user journeys.
Video (PT): Mapeamento de Fluxo de Usuário - Criando jornadas do usuário.
Guide (EN): Figma's Free "Design Basics" Guide (mobile-first section).
Course (EN): Great Learning Free Figma Course (accessibility module).

Hands-On Activities (3–4 hours):
• Sketch a simple user flow on paper/phone notes (e.g., Namibia tourism app).
• Build/Launch Mini-Project: Ethical prompt for a mobile layout idea; sketch low-fi wireframe in Figma.

Assignment: Create a basic wireframe for a simple app (e.g., task list); include accessibility notes. Share in forum.

Week 2: Ethical Graphics Creation (Icons, Logos, Banners via AI Tools)

Objectives: Generate and refine graphics ethically; focus on originality for commercial use.

Lessons & Topics (2–3 hours/day):
Graphics Fundamentals (Day 1): Icons, logos, banners; vectors vs. raster.
Ethical AI Image Tools (Day 2): Free/open alternatives (e.g., Leonardo.Ai free tier, Flux via Hugging Face demos, Craiyon for basics).
Prompting for Graphics (Day 3): Ethical prompts (e.g., "Original flat icon set for Namibia farming app, diverse users").
Refinement & Originality (Day 4): Import to Figma; manual edits for uniqueness.

Resources:

Video (EN): Vector vs Raster Graphics Explained - Understanding graphics types.
Video (PT): Gráficos Vetoriais vs Raster - Entendendo tipos de gráficos.
Video (EN): Icon Design Tutorial - Creating professional icons.
Video (PT): Tutorial de Design de Ícones - Criando ícones profissionais.

Video (EN): Logo Design Basics - Logo design principles.
Video (PT): Fundamentos de Design de Logo - Princípios de design de logo.
Video (EN): AI Image Generation Tools Tutorial - Using Leonardo.Ai or similar tools.
Video (PT): Tutorial de Ferramentas de IA para Imagens - Usando Leonardo.Ai ou ferramentas similares.
Video (EN): Figma Graphics and Icons - Working with graphics in Figma.
Video (PT): Gráficos e Ícones no Figma - Trabalhando com gráficos no Figma.
Tool (EN): Leonardo.Ai or Flux free tiers (ethical, commercial-safe options).
Course (EN): Udemy Free "Learn Figma" (graphics section).

Hands-On Activities (3–4 hours):
• Generate 5 icons ethically; import/edit in Figma.
• Build/Launch Mini-Project: Create a logo for your future app idea.

Assignment: Build an asset library (10 graphics); document ethical sources/process.

Week 3: Prototyping in Figma (Interactive Mockups)

Objectives: Turn designs into clickable prototypes; add interactions.

Lessons & Topics (2–3 hours/day):
Figma Interface Deep Dive (Day 1): Layers, components, auto-layout.
Building Screens (Day 2): From wireframes to high-fi (using Week 2 graphics).
Interactions & Prototyping (Day 3): Links, overlays, animations.
Testing Prototypes (Day 4): Share links; gather feedback ethically.

Resources:

Video (EN): Figma Prototyping Tutorial - Complete prototyping guide.
Video (PT): Tutorial de Prototipagem no Figma - Guia completo de prototipagem.
Video (EN): Figma Components and Auto-Layout - Advanced Figma features.
Video (PT): Componentes e Auto-Layout no Figma - Recursos avançados do Figma.
Video (EN): Interactive Prototypes in Figma - Adding interactions and animations.
Video (PT): Protótipos Interativos no Figma - Adicionando interações e animações.
Video (EN): Mobile App Prototype Tutorial - Step-by-step mobile prototyping.
Video (PT): Tutorial de Protótipo de App Mobile - Prototipagem mobile passo a passo.
Video (EN): Figma Layers and Organization - Managing complex designs.
Templates (EN): Figma Community Templates (mobile prototypes).

Hands-On Activities (3–4 hours):
• Build 3 screens for a sample app; add basic interactions.

• Build/Launch Mini-Project: Prototype a login flow.

Assignment: Complete one full prototype draft; test with a peer.

Week 4: Design Systems and Iteration

Objectives: Create reusable components; iterate based on feedback.

Lessons & Topics (2–3 hours/day):
Design Systems (Day 1): Components, styles, variables.
Iteration & Feedback (Day 2): Ethical critique (inclusive language).
Final Polish (Day 3): Export assets; prepare for publishing.
Namibia Context (Day 4): Localize designs (e.g., cultural colors).

Resources:

Video (EN): Design Systems Tutorial - Creating design systems.
Video (PT): Tutorial de Design Systems - Criando sistemas de design.
Video (EN): Figma Design Tokens and Variables - Using variables in Figma.
Video (PT): Variáveis e Tokens no Figma - Usando variáveis no Figma.
Video (EN): Component Libraries in Figma - Building reusable components.
Video (PT): Bibliotecas de Componentes no Figma - Construindo componentes reutilizáveis.
Video (EN): Exporting Assets from Figma - Preparing assets for development.
Video (PT): Exportando Assets do Figma - Preparando assets para desenvolvimento.
Templates (EN): Figma Community (free mobile templates).
Tutorial (EN): GeeksforGeeks Figma Tutorial (2026 update).

Hands-On Activities (3–4 hours):
• Build a component library; apply to prototypes.
• Iterate based on mentor feedback.

Assignment: Finalize prototypes; prepare for launch.

Month 2 Milestone Project

Project: Design and publish 3 interactive prototypes (e.g., e-commerce app, task manager, Namibia tourism app) on Figma Community/Behance. Include ethical graphics, mobile-first layout, accessibility checks. Share live links + short demo video. (6–8 hours total.) Rubric: Visual Quality (40%), Functionality/Interactions (30%), Ethical & Accessibility (20%), Documentation (10%). Mentor approval unlocks Month 3—use these exact designs as

coding blueprints!

You're now ready to turn designs into live apps. Excited for your launches—share in the community! Ø=Þ€

PAIED Program Detailed Curriculum for Month 3 – AI Powered Web Development

PAIED Program: Detailed Curriculum for Month 3 – AI-Powered Web Development

Month Overview This is where you start coding and launching real websites—turning your Month 2 Figma prototypes into live, functioning web apps. True to our philosophy—"They may not believe the papers, but you will get paid for the working projects"—you'll build full-stack sites quickly using AI assistance (e.g., Grok/Claude for code generation). We prioritize Next.js + React for modern, fast web apps, integrated with Supabase for backend (auth, database). Focus on responsive, mobile-first design (building on Month 2) and simple AI features (e.g., chatbots or recommendations).

By the end, you'll launch a complete, deployed website (live URL on Vercel) using one of your Month 2 prototypes. This proves your skills—share the link to attract freelance gigs!

Ethical AI Focus: Use AI for code suggestions ethically (modify outputs, cite if sharing); ensure inclusive features (e.g., accessible UI).

Resources Provided: Free Vercel/Supabase accounts. Weekly live webinars (recorded). Mentor code reviews. Community for debugging help.

Weekly Effort: 12 hours (non-workers) or 10 hours (workers). Use AI to speed up—aim to launch in hours per feature.

Week 1: Next.js + React Fundamentals

Objectives: Set up a Next.js project; understand components, routing, and styling.

Lessons & Topics (2–3 hours/day):
Why Next.js in 2026 (Day 1): Server-side rendering, SEO, performance; vs. plain React.
Project Setup (Day 2): Create app with create-next-app; folder structure (app router).
Basic Components (Day 3): Functional components, props, JSX.
Styling & Responsiveness (Day 4): Tailwind CSS (built-in); mobile-first media queries.

Resources:

Video (EN): Next.js 14 Full Course for Beginners - Complete Next.js tutorial.
Video (PT): Curso Completo de Next.js para Iniciantes - Tutorial completo de Next.js.
Video (EN): React Basics for Beginners - React fundamentals.
Video (PT): React Básico para Iniciantes - Fundamentos do React.

Video (EN): Next.js vs React Explained - Understanding the difference.
Video (PT): Next.js vs React Explicado - Entendendo a diferença.
Video (EN): Next.js App Router Tutorial - Modern Next.js routing.
Video (PT): Tutorial de App Router do Next.js - Roteamento moderno do Next.js.
Video (EN): React Components and Props - Component basics.
Video (PT): Componentes e Props do React - Conceitos básicos de componentes.
Video (EN): Tailwind CSS Tutorial - Styling with Tailwind.
Video (PT): Tutorial de Tailwind CSS - Estilização com Tailwind.
Docs (EN): Official Next.js Docs (Getting Started).

Hands-On Activities (3–4 hours):
• Set up a blank Next.js app; add basic pages from your Month 2 prototype.
• Build/Launch Mini-Project: Ethical AI use—prompt for component code; modify manually.

Assignment: Build a multi-page site skeleton (home, about); share localhost screenshot in forum.

Week 2: Styling, Responsiveness, and Supabase Integration

Objectives: Make sites beautiful/responsive; add backend with Supabase.

Lessons & Topics (2–3 hours/day):
Advanced Styling (Day 1): Tailwind utilities; custom themes.
Responsive Design (Day 2): Flex/grid; mobile-first breakpoints.
Supabase Setup (Day 3): Create project; auth (email/password).
Database Basics (Day 4): Tables, rows; fetch data in Next.js (server components).

Resources:

Video (EN): Supabase + Next.js Complete Tutorial - Full integration guide.
Video (PT): Tutorial Completo Supabase + Next.js - Guia completo de integração.
Video (EN): Supabase Authentication Tutorial - Setting up auth.
Video (PT): Tutorial de Autenticação Supabase - Configurando autenticação.
Video (EN): Responsive Design with Tailwind - Mobile-first responsive design.
Video (PT): Design Responsivo com Tailwind - Design responsivo mobile-first.
Video (EN): Flexbox and Grid Layout Tutorial - CSS layout fundamentals.
Video (PT): Tutorial de Flexbox e Grid - Fundamentos de layout CSS.
Video (EN): Supabase Database Setup - Creating and managing databases.
Video (PT): Configuração de Banco de Dados Supabase - Criando e gerenciando bancos de dados.
Guide (EN): Supabase Next.js Quickstart Guide.

Hands-On Activities (3–4 hours):
• Implement your Month 2 UI in code; add responsive navigation.
• Build/Launch Mini-Project: Connect Supabase—simple user registration form.

Assignment: Functional login page with Supabase auth; test responsiveness.

Week 3: AI Features Integration

Objectives: Add simple AI-powered elements (e.g., chatbot, recommendations).

Lessons & Topics (2–3 hours/day):
AI APIs Basics (Day 1): Free tiers (e.g., Grok API if available, or OpenAI/Hugging Face).
Chatbot Implementation (Day 2): Embed a basic chat interface.
Recommendations/Other Features (Day 3): Fetch data + simple AI logic (e.g., content suggestions).
Ethical Integration (Day 4): Privacy (no unnecessary data); transparency (label AI-generated content).

Resources:

Video (EN): Building AI Chatbot in Next.js - Complete chatbot implementation.
Video (PT): Construindo Chatbot de IA no Next.js - Implementação completa de chatbot.
Video (EN): OpenAI API Integration Tutorial - Integrating AI APIs.
Video (PT): Tutorial de Integração OpenAI API - Integrando APIs de IA.
Video (EN): Hugging Face API Tutorial - Using Hugging Face models.
Video (PT): Tutorial de API Hugging Face - Usando modelos do Hugging Face.
Video (EN): AI Recommendations System - Building recommendation features.
Video (PT): Sistema de Recomendações com IA - Construindo recursos de recomendação.
Video (EN): Privacy in AI Applications - Ethical AI integration.
Video (PT): Privacidade em Aplicações de IA - Integração ética de IA.
API (EN): Hugging Face Inference API (free chat models).
Tutorial (EN): Add AI Chat to Next.js.

Hands-On Activities (3–4 hours):
• Integrate a basic AI chatbot or recommendation feed.
• Build/Launch Mini-Project: Working AI feature in your site.

Assignment: Complete AI feature with ethical disclosure note.

Week 4: Testing, Optimization, and Deployment

Objectives: Polish and launch your site publicly.

Lessons & Topics (2–3 hours/day):
Testing Basics (Day 1): Manual checks; accessibility tools.
Performance Optimization (Day 2): Image optimization, lazy loading.
Vercel Deployment (Day 3): Connect GitHub; custom domains.
Post-Launch (Day 4): Analytics; iteration for real users.

Resources:

Video (EN): Deploy Next.js to Vercel - Complete deployment guide.
Video (PT): Deploy Next.js no Vercel - Guia completo de deploy.
Video (EN): Next.js Performance Optimization - Speed optimization techniques.
Video (PT): Otimização de Performance Next.js - Técnicas de otimização de velocidade.
Video (EN): Image Optimization in Next.js - Optimizing images.
Video (PT): Otimização de Imagens no Next.js - Otimizando imagens.
Video (EN): Accessibility Testing Tools - Testing for accessibility.
Video (PT): Ferramentas de Teste de Acessibilidade - Testando acessibilidade.
Video (EN): Lighthouse Performance Testing - Using Lighthouse.
Video (PT): Teste de Performance com Lighthouse - Usando Lighthouse.
Video (EN): GitHub Integration with Vercel - CI/CD setup.
Video (PT): Integração GitHub com Vercel - Configuração CI/CD.
Docs (EN): Vercel Docs (Next.js deployment).
Tool (EN): Lighthouse for testing.

Hands-On Activities (3–4 hours):
• Optimize and deploy to Vercel.
• Build/Launch Mini-Project: Final tweaks; prepare launch demo.

Assignment: Deploy site; share live URL in forum.

Month 3 Milestone Project

Project: Launch a fully functional website (e.g., portfolio, e-commerce mockup, Namibia business directory) using your Month 2 prototype. Include Supabase backend (e.g., user data) and one AI feature. Deploy on Vercel (live URL required). Add README with ethical notes. (6–8 hours total.) Rubric: Functionality (50%), Responsiveness/Design (20%), AI Integration (15%), Ethical Practices (10%), Deployment Quality (5%). Mentor approval unlocks Month 4—your first paid-project-ready site!

Launch it and share the link—clients pay for working sites like this! Ø=Þ€

PAIED Program Detailed Curriculum for Month 4 – AI Powered Mobile Development

PAIED Program: Detailed Curriculum for Month 4 – AI-Powered Mobile Development

Month Overview Welcome to mobile! You'll now transform your Month 2 Figma prototypes into real cross-platform mobile apps using our primary stack: Flutter + Dart. This builds directly on Month 3 (web with Next.js/Supabase), shifting to native-feeling apps that run on Android/iOS/web. Per our philosophy—"They may not believe the papers, but you will get paid for the working projects"—focus on launching quickly with AI help (e.g., prompting for Dart code). Integrate Supabase for backend (auth, data)—no separate servers needed.

By month's end, you'll publish a working mobile app (e.g., APK share or web build) based on your prototypes. This is freelance-ready—clients pay for apps like these!

Ethical AI Focus: Generate code ethically (modify outputs); ensure app features promote inclusivity (e.g., offline support for Namibia's connectivity).

Resources Provided: Free Flutter/Supabase setup. Weekly live webinars (recorded). Mentor reviews. Community for troubleshooting.

Weekly Effort: 13 hours (non-workers) or 11 hours (workers). Use AI to accelerate—target functional builds in hours.

Week 1: Flutter + Dart Setup and Basics

Objectives: Install Flutter; understand Dart syntax; build simple screens.

Lessons & Topics (2–3 hours/day):
Flutter in 2026 (Day 1): Why Flutter/Dart (single codebase, fast performance, Namibia mobile dominance).
Setup (Day 2): Install Flutter SDK, Android Studio/VS Code; create first project (flutter create).
Dart Fundamentals (Day 3): Variables, functions, async/await.
First Widgets (Day 4): Stateless/Stateful widgets; MaterialApp basics.

Resources:

Video (EN): Flutter Tutorial for Beginners - Complete Flutter course.
Video (PT): Tutorial de Flutter para Iniciantes - Curso completo de Flutter.
Video (EN): Flutter Installation and Setup - Setting up Flutter SDK.

Video (PT): Instalação e Configuração do Flutter - Configurando o Flutter SDK.
Video (EN): Dart Programming Language Tutorial - Dart fundamentals.
Video (PT): Tutorial de Dart - Fundamentos de Dart.
Video (EN): Flutter Widgets Explained - Understanding widgets.
Video (PT): Widgets do Flutter Explicados - Entendendo widgets.
Video (EN): Stateful vs Stateless Widgets - Widget types explained.
Video (PT): Stateful vs Stateless Widgets - Tipos de widgets explicados.
Video (EN): MaterialApp Basics - Material Design in Flutter.
Video (PT): MaterialApp Básico - Material Design no Flutter.
Docs (EN): Official Flutter Docs (Install & First App).

Hands-On Activities (3–4 hours):
• Run flutter doctor; build "Hello World" app.
• Build/Launch Mini-Project: Ethical AI—prompt for basic Dart widget code; refine.

Assignment: Simple screen (e.g., home page) matching one Month 2 prototype.

Week 2: Implementing Designs from Month 2

Objectives: Translate Figma to Flutter UI; build navigation.

Lessons & Topics (2–3 hours/day):
Layout Widgets (Day 1): Scaffold, AppBar, Column/Row.
Custom UI (Day 2): Buttons, cards, images from Month 2 graphics.
Navigation (Day 3): Routes, Navigator 2.0 basics.
Theming (Day 4): Colors, fonts; dark mode ethics.

Resources:

Video (EN): Flutter Layout Tutorial - Column, Row, Stack.
Video (PT): Tutorial de Layout Flutter - Column, Row, Stack.
Video (EN): Flutter Navigation Tutorial - Navigator 2.0 basics.
Video (PT): Tutorial de Navegação Flutter - Conceitos básicos do Navigator 2.0.
Video (EN): Flutter Theming and Colors - Custom themes.
Video (PT): Temas e Cores no Flutter - Temas personalizados.
Video (EN): Flutter Custom UI Components - Building custom UI.
Video (PT): Componentes UI Personalizados Flutter - Construindo UI customizada.
Video (EN): Flutter AppBar and Scaffold - Basic app structure.
Video (PT): AppBar e Scaffold Flutter - Estrutura básica do app.
Cookbook (EN): Flutter Cookbook (Layouts).

Hands-On Activities (3–4 hours):
• Recreate 3–5 screens from your prototype.
• Build/Launch Mini-Project: Multi-screen app with navigation.

Assignment: Complete app with navigation; share emulator screenshot.

Week 3: State Management, Supabase Auth/Data

Objectives: Handle dynamic data; add backend with Supabase.

Lessons & Topics (2–3 hours/day):
State Management (Day 1): Provider or Riverpod basics (simple for beginners).
Supabase Auth (Day 2): Email/password signup/login.
Database Ops (Day 3): CRUD with Supabase client.
Realtime Features (Day 4): Subscriptions for live updates.

Resources:

Video (EN): Flutter State Management Tutorial - Provider/Riverpod basics.
Video (PT): Tutorial de Gerenciamento de Estado Flutter - Conceitos básicos de Provider/Riverpod.
Video (EN): Supabase Flutter Authentication - Complete auth setup.
Video (PT): Autenticação Supabase Flutter - Configuração completa de autenticação.
Video (EN): Supabase Database with Flutter - CRUD operations.
Video (PT): Banco de Dados Supabase com Flutter - Operações CRUD.
Video (EN): Flutter Provider Tutorial - State management with Provider.
Video (PT): Tutorial Flutter Provider - Gerenciamento de estado com Provider.
Video (EN): Realtime Features in Flutter - Live data updates.
Video (PT): Recursos em Tempo Real Flutter - Atualizações de dados ao vivo.
Docs (EN): Supabase Flutter Docs (Auth & Database).

Hands-On Activities (3–4 hours):
• Add user auth; store/fetch data (e.g., profile).
• Build/Launch Mini-Project: App with login + data persistence.

Assignment: Complete authentication flow with data persistence.

Week 4: Testing and Publishing (APK/Web Build)

Objectives: Test thoroughly; publish shareable build.

Lessons & Topics (2–3 hours/day):
Basic Testing (Day 1): Unit/widget tests; manual on emulator.
Offline/Performance (Day 2): Cached data; ethical battery use.
Building (Day 3): APK for Android; web build.
Simple AI Touch (Day 4): Optional basic AI (e.g., local recommendation).

Resources:

Video (EN): Flutter Testing Tutorial - Unit and widget tests.
Video (PT): Tutorial de Testes Flutter - Testes unitários e de widgets.
Video (EN): Building APK in Flutter - Android APK creation.
Video (PT): Criar APK no Flutter - Criação de APK Android.
Video (EN): Flutter Web Build - Web deployment.
Video (PT): Build Web Flutter - Deploy web.
Video (EN): Flutter Performance Optimization - App optimization.
Video (PT): Otimização de Performance Flutter - Otimização de apps.
Video (EN): Offline Support in Flutter - Caching strategies.
Video (PT): Suporte Offline Flutter - Estratégias de cache.
Docs (EN): Flutter Docs (Build & Release).

Hands-On Activities (3–4 hours):
• Generate APK/web build; test on device/emulator.
• Build/Launch Mini-Project: Final polish; prepare for launch.

Assignment: Deploy build; share APK/web link.

Month 4 Milestone Project

Project: Publish a working mobile app (e.g., task app, Namibia directory) using your Month 2 prototype. Include Supabase (auth/data) and responsive UI. Share APK/web link + short demo video. (6–8 hours total.) Rubric: Functionality (50%), UI Fidelity to Design (20%), Backend Integration (15%), Ethical/Accessibility (10%), Build Quality (5%). Mentor approval unlocks Month 5—your first mobile product clients will pay for!

Launch it and celebrate—next: deeper programming! Ø=Þ€

PAIED Program Detailed Curriculum for Month 5 – Core Programming Mastery with AI

PAIED Program: Detailed Curriculum for Month 5 – Core Programming Mastery with AI

Month Overview Now that you've launched web and mobile apps, this month deepens your programming fluency in our priority languages: Python (AI/scripting), Dart (Flutter/mobile), and JavaScript/TypeScript (Next.js/web). You'll apply concepts to real problems, using AI tools ethically for debugging and ideas—proving our philosophy: "They may not believe the papers, but you will get paid for the working projects."

Focus on clean, efficient code that powers freelance-ready apps. By the end, you'll refactor and enhance a prior project (Month 3 or 4), relaunching it with advanced features.

Ethical AI Focus: Use AI for suggestions/explanations only; always modify code for originality; document contributions.

Resources Provided: Free VS Code extensions (e.g., Python/Dart). Weekly live webinars (recorded). Mentor code reviews. Community for pair programming.

Weekly Effort: 13 hours (non-workers) or 11 hours (workers). AI accelerates learning—target clean code in hours.

Week 1: Python Mastery (Functions, OOP, Libraries)

Objectives: Build robust Python scripts; understand object-oriented programming for reusable code.

Lessons & Topics (2–3 hours/day):
Advanced Functions (Day 1): Lambdas, decorators, generators.
OOP Deep Dive (Day 2): Classes, inheritance, polymorphism.
Key Libraries (Day 3): Requests (APIs), Pandas (data).
Ethical Coding (Day 4): Clean code principles; avoiding plagiarism in AI-generated scripts.

Resources:

Video (EN): Python Advanced Functions - Lambdas, decorators, generators.
Video (PT): Funções Avançadas Python - Lambdas, decoradores, geradores.
Video (EN): Python OOP Complete Tutorial - Object-oriented programming.
Video (PT): Tutorial Completo OOP Python - Programação orientada a objetos.
Video (EN): Python Classes and Inheritance - OOP deep dive.

Video (PT): Classes e Herança Python - Mergulho profundo em OOP.
Video (EN): Python Requests Library - API calls with Requests.
Video (PT): Biblioteca Requests Python - Chamadas de API com Requests.
Video (EN): Pandas Tutorial for Beginners - Data manipulation.
Video (PT): Tutorial Pandas para Iniciantes - Manipulação de dados.
Video (EN): Clean Code Principles - Writing maintainable code.
Video (PT): Princípios de Código Limpo - Escrevendo código manutenível.
Docs (EN): Official Python Docs (Advanced Topics).

Hands-On Activities (3–4 hours):
• Build a class-based script (e.g., Namibia data tool).
• Build/Launch Mini-Project: Ethical AI—prompt for function ideas; implement manually.

Assignment: Python module with OOP; test locally.

Week 2: Dart Deep Dive (Async, Null Safety)

Objectives: Master Dart for reliable Flutter apps.

Lessons & Topics (2–3 hours/day):
Null Safety (Day 1): Sound null safety; avoiding crashes.
Async Programming (Day 2): Futures, async/await, streams.
Advanced Features (Day 3): Extensions, mixins.
Best Practices (Day 4): Error handling; ethical async patterns.

Resources:

Video: Dart Null Safety Tutorial - Understanding null safety.
Video: Dart Async Programming - Futures and async/await.
Video: Dart Streams Tutorial - Working with streams.
Video: Dart Extensions and Mixins - Advanced Dart features.
Video: Error Handling in Dart - Try-catch patterns.
Docs: Dart Docs (Null Safety & Async).

Hands-On Activities (3–4 hours):
• Async data fetching in a Flutter snippet.
• Build/Launch Mini-Project: Dart script with null-safe async operations.

Assignment: Complete async implementation with error handling.

Week 3: Advanced React/Next.js Patterns

Objectives: Handle complex state and patterns in web apps.

Lessons & Topics (2–3 hours/day):
Custom Hooks (Day 1): Reusable logic.
State Management (Day 2): Context, reducers.
Server Actions (Day 3): Next.js 14+ patterns.
Optimization (Day 4): Memoization, ethical performance.

Resources:

Video (EN): React Custom Hooks Tutorial - Creating reusable hooks.
Video (PT): Tutorial Custom Hooks React - Criando hooks reutilizáveis.
Video (EN): React Context API Tutorial - State management with Context.
Video (PT): Tutorial Context API React - Gerenciamento de estado com Context.
Video (EN): React useReducer Hook - Advanced state management.
Video (PT): Hook useReducer React - Gerenciamento avançado de estado.
Video (EN): Next.js Server Actions - Server-side actions.
Video (PT): Server Actions Next.js - Ações do lado do servidor.
Video (EN): React Memoization - Performance optimization.
Video (PT): Memoização React - Otimização de performance.
Docs (EN): Next.js Docs (Advanced Patterns).

Hands-On Activities (3–4 hours):
• Custom hook for API calls.
• Build/Launch Mini-Project: Next.js page with advanced state.

Assignment: Implement complex state management pattern.

Week 4: APIs, Error Handling, AI-Assisted Debugging

Objectives: Connect apps reliably; debug efficiently.

Lessons & Topics (2–3 hours/day):
REST APIs (Day 1): Fetching, POST/PUT.
Error Handling (Day 2): Try-catch, custom errors.
Debugging Tools (Day 3): VS Code debugger; AI for fixes.
Integration (Day 4): Secure, ethical API use.

Resources:

Video (EN): REST API Tutorial - Complete API guide.
Video (PT): Tutorial REST API - Guia completo de API.
Video (EN): Fetch API Tutorial - Making API calls.
Video (PT): Tutorial Fetch API - Fazendo chamadas de API.
Video (EN): Error Handling in JavaScript - Try-catch patterns.
Video (PT): Tratamento de Erros JavaScript - Padrões try-catch.
Video (EN): VS Code Debugging Tutorial - Using debugger.
Video (PT): Tutorial de Debug VS Code - Usando o debugger.
Video (EN): AI-Assisted Debugging - Using AI for debugging.
Video (PT): Debug Assistido por IA - Usando IA para debug.
Tool (EN): Postman for API testing.

Hands-On Activities (3–4 hours):
• Debug a buggy script with AI help.
• Build/Launch Mini-Project: API-integrated feature.

Assignment: Complete API integration with error handling.

Month 5 Milestone Project

Project: Refactor and enhance a prior project (e.g., add advanced backend to Month 3 site or async features to Month 4 app). Relaunch publicly (live URL/APK). Include ethical AI notes in README. (6–8 hours total.) Rubric: Improvement Quality (40%), Code Cleanliness (30%), New Features (20%), Ethical Documentation (10%). Mentor approval unlocks Month 6—your code is now professional-grade!

Mastered the core—next: building AI agents! Ø=Þ€

PAIED Program Detailed Curriculum for Month 6 – AI Agents & Automation Development

PAIED Program: Detailed Curriculum for Month 6 – AI Agents & Automation Development

Month Overview This month introduces agentic AI—autonomous systems that plan, reason, and execute multi-step tasks (a 2026 hotspot). You'll build practical agents and automations using Python (primary for AI) and LangChain frameworks, optionally tying into Supabase Functions for backend deployment. Aligning with our philosophy—"They may not believe the papers, but you will get paid for the working projects"—focus on launching real agents (e.g., business bots) that solve Namibia-relevant problems.

By the end, you'll launch a public AI agent demo (GitHub repo + live example), ready for freelancing (e.g., client automations).

Ethical AI Focus: Ensure agents avoid bias; handle data privately; transparency in actions.

Resources Provided: Free Hugging Face/LangChain access. Weekly live webinars (recorded). Mentor reviews. Community for collaboration.

Weekly Effort: 14 hours (non-workers) or 12 hours (workers). AI tools speed prototyping.

Week 1: Agent Concepts and Planning

Objectives: Understand agentic AI; learn planning/reasoning loops.

Lessons & Topics (2–3 hours/day):
Agentic AI in 2026 (Day 1): Agents vs. simple bots; reasoning engines.
Core Components (Day 2): Tools, memory, planning (ReAct pattern).
Use Cases (Day 3): Namibia examples (e.g., farming advisor agent).
Ethical Planning (Day 4): Avoiding harmful actions; user consent.

Resources:

Video (EN): Agentic AI Explained - Understanding agentic systems.
Video (PT): IA Agêntica Explicada - Entendendo sistemas agênticos.
Video (EN): LangChain Agents Tutorial - Building AI agents.
Video (PT): Tutorial LangChain Agents - Construindo agentes de IA.
Video (EN): ReAct Pattern Explained - Reasoning and acting.
Video (PT): Padrão ReAct Explicado - Raciocínio e ação.
Video (EN): AI Agent Architecture - System design.

Video (PT): Arquitetura de Agentes de IA - Design de sistemas.
Video (EN): Planning in AI Agents - Agent planning loops.
Video (PT): Planejamento em Agentes de IA - Loops de planejamento.
Docs (EN): LangChain Docs (Agents Intro).

Hands-On Activities (3–4 hours):
• Diagram a simple agent workflow.
• Build/Launch Mini-Project: Plan an agent for a personal task.

Assignment: Complete agent plan with ethical review.

Week 2: Building Simple Bots (e.g., Email/Social Automation)

Objectives: Create basic single-task agents.

Lessons & Topics (2–3 hours/day):
LangChain Setup (Day 1): Install; basic chains.
Tool Integration (Day 2): Custom tools (e.g., email API).
Simple Agents (Day 3): Zero-shot ReAct agent.
Testing (Day 4): Run loops ethically.

Resources:

Video (EN): LangChain Setup Tutorial - Installation and basics.
Video (PT): Tutorial de Configuração LangChain - Instalação e conceitos básicos.
Video (EN): LangChain Tools Tutorial - Creating custom tools.
Video (PT): Tutorial de Ferramentas LangChain - Criando ferramentas personalizadas.
Video (EN): Building AI Bots with LangChain - Bot development.
Video (PT): Construindo Bots de IA com LangChain - Desenvolvimento de bots.
Video (EN): Email Automation with Python - Email bots.
Video (PT): Automação de Email com Python - Bots de email.
Video (EN): ReAct Agent Implementation - Zero-shot agents.
Video (PT): Implementação ReAct Agent - Agentes zero-shot.
Guide (EN): LangChain Python Quickstart.

Hands-On Activities (3–4 hours):
• Build an email summarizer agent.
• Build/Launch Mini-Project: Simple bot script.

Assignment: Complete bot with demo run.

Week 3: Multi-Agent Systems and Integrations

Objectives: Handle complex, multi-tool agents.

Lessons & Topics (2–3 hours/day):
Memory & State (Day 1): Conversation memory.
Multi-Tool Agents (Day 2): Search + calculation tools.
Supabase Integration (Day 3): Store agent state/data.
Ethical Multi-Step (Day 4): Logging actions transparently.

Resources:

Video (EN): Multi-Agent Systems Tutorial - Coordinating multiple agents.
Video (PT): Tutorial Sistemas Multi-Agente - Coordenando múltiplos agentes.
Video (EN): LangChain Memory Tutorial - Conversation memory.
Video (PT): Tutorial Memória LangChain - Memória de conversação.
Video (EN): Supabase with LangChain - Storing agent state.
Video (PT): Supabase com LangChain - Armazenando estado do agente.
Video (EN): Multi-Tool Agents - Agents with multiple tools.
Video (PT): Agentes Multi-Ferramenta - Agentes com múltiplas ferramentas.
Video (EN): Agent State Management - Managing agent state.
Video (PT): Gerenciamento de Estado de Agentes - Gerenciando estado do agente.
Library (EN): LangGraph for multi-agent (advanced).

Hands-On Activities (3–4 hours):
• Multi-tool agent (e.g., research + summarize).
• Build/Launch Mini-Project: Agent with 3+ tools.

Assignment: Complete multi-tool agent implementation.

Week 4: Testing in Real Scenarios and Deployment

Objectives: Deploy agents reliably.

Lessons & Topics (2–3 hours/day):
Real-World Testing (Day 1): Edge cases; error recovery.
Deployment Options (Day 2): Supabase Edge Functions; Streamlit demo.
Scaling (Day 3): Rate limits; cost ethics.
Final Polish (Day 4): User interface for agent interaction.

Resources:

Video (EN): Supabase Edge Functions Tutorial - Deploying functions.
Video (PT): Tutorial Supabase Edge Functions - Deploy de funções.
Video (EN): Streamlit Tutorial - Building agent UIs.
Video (PT): Tutorial Streamlit - Construindo UIs de agentes.
Video (EN): AI Agent Deployment - Production deployment.
Video (PT): Deploy de Agentes de IA - Deploy em produção.
Video (EN): Error Handling in Agents - Robust agent design.
Video (PT): Tratamento de Erros em Agentes - Design robusto de agentes.
Video (EN): Scaling AI Agents - Performance optimization.
Video (PT): Escalando Agentes de IA - Otimização de performance.
Guide (EN): Supabase Edge Functions Guide.

Hands-On Activities (3–4 hours):
• Deploy a test agent.
• Build/Launch Mini-Project: Prepare milestone demo.

Assignment: Deploy agent; share live demo link.

Month 6 Milestone Project

Project: Launch a public AI agent (e.g., Namibia business advisor bot handling multi-steps). GitHub repo + live demo (Streamlit/Supabase). Include ethical guidelines README. (6–8 hours total.) Rubric: Autonomy/Planning (40%), Functionality (30%), Integration (15%), Ethical Practices (10%), Demo Quality (5%). Mentor approval unlocks Month 7—agents clients pay big for!

Your agents are alive—next: customizing models! Ø=Þ€

PAIED Program Detailed Curriculum for Month 7 – Model Fine Tuning & Customization

PAIED Program: Detailed Curriculum for Month 7 – Model Fine-Tuning & Customization

Month Overview This month empowers you to customize pre-trained AI models ethically for specific tasks—e.g., Namibia-focused chatbots or local language support. Using Python and Hugging Face (with LoRA for efficient tuning on limited hardware), you'll adapt models without massive resources. True to our philosophy—"They may not believe the papers, but you will get paid for the working projects"—focus on launching tuned models integrated into prior apps (e.g., from Months 3–6).

By the end, you'll fine-tune a model, integrate it into an existing project, and publish it publicly (Hugging Face repo + demo)—a high-value skill for freelance clients!

Ethical AI Focus: Prioritize bias-free data; privacy (anonymize); transparency (document changes); inclusivity (e.g., underrepresented languages).

Resources Provided: Free Hugging Face account (hub for models). Weekly live webinars (recorded). Mentor reviews. Community for dataset sharing.

Weekly Effort: 14 hours (non-workers) or 12 hours (workers). Use free Colab for compute.

Week 1: Model Basics (Pre-Trained vs. Custom)

Objectives: Understand model types; why/when to fine-tune.

Lessons & Topics (2–3 hours/day):
Pre-Trained Models (Day 1): Open-source LLMs (e.g., Mistral, Llama variants via HF).
Full vs. Parameter-Efficient Tuning (Day 2): LoRA introduction (low-rank adapters—efficient for beginners).
When to Customize (Day 3): Domain adaptation (e.g., Namibia business queries).
Ethical Foundations (Day 4): Model cards; responsible sourcing.

Resources:

Video (EN): Hugging Face Transformers Tutorial - Working with models.
Video (PT): Tutorial Hugging Face Transformers - Trabalhando com modelos.
Video (EN): Fine-Tuning LLMs Explained - Fine-tuning basics.
Video (PT): Fine-Tuning de LLMs Explicado - Conceitos básicos de fine-tuning.
Video (EN): LoRA Explained - Low-rank adaptation.

Video (PT): LoRA Explicado - Adaptação de baixo rank.
Video (EN): Pre-trained Models Overview - Understanding model types.
Video (PT): Visão Geral de Modelos Pré-treinados - Entendendo tipos de modelos.
Video (EN): Hugging Face Hub Tutorial - Finding and using models.
Video (PT): Tutorial Hugging Face Hub - Encontrando e usando modelos.
Course (EN): Hugging Face Course (Fine-Tuning Chapter).
Blog (EN): LoRA Explained Visually.

Hands-On Activities (3–4 hours):
• Explore HF models; run inference on a base model.
• Build/Launch Mini-Project: Choose a base model + task (e.g., local Q&A).

Assignment: Complete model selection with ethical rationale note.

Week 2: Ethical Data Preparation

Objectives: Curate/clean datasets responsibly.

Lessons & Topics (2–3 hours/day):
Dataset Sources (Day 1): HF datasets; public domain.
Cleaning & Formatting (Day 2): JSONL for instruction tuning.
Bias Mitigation (Day 3): Diversity checks; debiasing tools.
Privacy & Ethics (Day 4): Anonymization; consent.

Resources:

Video: Hugging Face Datasets Tutorial - Working with datasets.
Video: Data Cleaning for ML - Preparing data.
Video: Bias Detection in Datasets - Identifying bias.
Video: Data Anonymization - Privacy protection.
Video: JSONL Format Tutorial - Instruction tuning format.
Guide: HF Datasets Library Guide.

Hands-On Activities (3–4 hours):
• Prepare a small ethical dataset (50–100 examples).
• Build/Launch Mini-Project: Cleaned dataset upload (private repo).

Assignment: Complete dataset with ethical audit.

Week 3: LoRA Fine-Tuning (Python + Hugging Face)

Objectives: Run efficient tuning with PEFT/LoRA.

Lessons & Topics (2–3 hours/day):
PEFT Setup (Day 1): Install libraries (peft, transformers).
LoRA Config (Day 2): Rank, alpha parameters.
Training Script (Day 3): Trainer API; Colab for GPU.
Monitoring (Day 4): Loss curves; overfitting ethics.

Resources:

Video (EN): LoRA Fine-Tuning Tutorial - Complete LoRA guide.
Video (PT): Tutorial Fine-Tuning LoRA - Guia completo de LoRA.
Video (EN): PEFT Library Tutorial - Parameter-efficient tuning.
Video (PT): Tutorial Biblioteca PEFT - Fine-tuning eficiente em parâmetros.
Video (EN): Fine-Tuning on Google Colab - Free GPU training.
Video (PT): Fine-Tuning no Google Colab - Treinamento gratuito com GPU.
Video (EN): Training Loss Curves - Monitoring training.
Video (PT): Curvas de Loss no Treinamento - Monitorando o treinamento.
Video (EN): Overfitting Prevention - Avoiding overfitting.
Video (PT): Prevenção de Overfitting - Evitando overfitting.
Docs (EN): HF PEFT Docs (LoRA Example).

Hands-On Activities (3–4 hours):
• Fine-tune on your dataset.
• Build/Launch Mini-Project: Trained LoRA adapter.

Assignment: Complete training with initial tests.

Week 4: Evaluation and Deployment

Objectives: Test quality; deploy/integrate tuned model.

Lessons & Topics (2–3 hours/day):
Evaluation Metrics (Day 1): Perplexity; human review.
Bias Testing (Day 2): Ethical checks post-tuning.
Deployment (Day 3): HF Inference; merge LoRA.
Integration (Day 4): Into prior app/agent.

Resources:

Video (EN): Model Evaluation Tutorial - Testing model quality.

Video (PT): Tutorial de Avaliação de Modelos - Testando qualidade do modelo.
Video (EN): Hugging Face Spaces Tutorial - Deploying demos.
Video (PT): Tutorial Hugging Face Spaces - Deploy de demos.
Video (EN): Model Inference Tutorial - Using fine-tuned models.
Video (PT): Tutorial de Inferência de Modelos - Usando modelos fine-tunados.
Video (EN): Integrating Models into Apps - Production integration.
Video (PT): Integrando Modelos em Apps - Integração em produção.
Video (EN): Model Cards Tutorial - Documenting models.
Video (PT): Tutorial Model Cards - Documentando modelos.
Platform (EN): HF Spaces for demos.

Hands-On Activities (3–4 hours):
• Deploy to HF; integrate.
• Build/Launch Mini-Project: Final tests; prepare milestone.

Assignment: Deploy model; share HF repo link.

Month 7 Milestone Project

Project: Fine-tune a model (e.g., for Namibia Q&A or local domain); integrate into a prior project (e.g., agent/app). Publish on Hugging Face (repo + Space demo). Include model card with ethical notes. (6–8 hours total.) Rubric: Performance Improvement (40%), Ethical Handling (30%), Integration (15%), Documentation/Deployment (15%). Mentor approval unlocks Month 8—custom models clients pay premium for!

You've now created your own AI—next: advanced integrations! Ø=Þ€

PAIED Program Detailed Curriculum for Month 8 – Advanced AI Integrations

PAIED Program: Detailed Curriculum for Month 8 – Advanced AI Integrations

Month Overview This month focuses on connecting AI systems to the real world—secure APIs, multi-agent coordination, and MCP (Model Context Protocol) servers for dynamic tool/agent interactions. Building on Months 6–7 (agents \+ fine-tuned models), you'll create robust, production-like systems using Supabase (Edge Functions, auth), Python, and optional Flutter/Next.js frontends. Per our philosophy—"They may not believe the papers, but you will get paid for the working projects"—emphasize launching integrated demos that solve complex tasks.

By the end, you'll build and deploy an end-to-end integrated AI system (e.g., app with agents \+ fine-tuned model \+ MCP tools)—freelance gold\!

Ethical AI Focus: Secure data handling; transparent tool use; avoid over-reliance on external APIs.

Resources Provided: Free Supabase/Colab. Weekly live webinars (recorded). Mentor reviews. Community debugging.

Weekly Effort: 15 hours (non-workers) or 13 hours (workers). Use AI for scaffolding.

Week 1: Integration Basics (Tools/APIs)

Objectives: Master secure API connections; tool-calling in agents.

Lessons & Topics (2–3 hours/day):
API Fundamentals (Day 1): REST/GraphQL; headers, auth.
Secure Calls (Day 2): Environment variables; Supabase secrets.
Tool Use in Agents (Day 3): LangChain tools (e.g., weather/search).
Ethical API (Day 4): Rate limits; privacy compliance.

Resources:

Video (EN): REST API Tutorial - Complete REST API guide.
Video (PT): Tutorial REST API - Guia completo de REST API.
Video (EN): GraphQL Tutorial - GraphQL basics.
Video (PT): Tutorial GraphQL - Conceitos básicos de GraphQL.
Video (EN): API Authentication Methods - JWT, OAuth, API keys.

Video (PT): Métodos de Autenticação API - JWT, OAuth, chaves API.
Video (EN): Environment Variables Tutorial - Secure configuration.
Video (PT): Tutorial Variáveis de Ambiente - Configuração segura.
Video (EN): LangChain Tools Tutorial - Creating and using tools.
Video (PT): Tutorial Ferramentas LangChain - Criando e usando ferramentas.
Video (EN): API Rate Limiting - Best practices.
Video (PT): Rate Limiting de API - Melhores práticas.
Docs (EN): Supabase Edge Functions Docs.
Guide (EN): LangChain Tools Guide.

Hands-On Activities (3–4 hours):
• Connect agent to public API (e.g., weather).

Assignment: Agent with 2 external tools.

Week 2: MCP Servers and Secure Comms

Objectives: Implement MCP for dynamic agent-tool links.

Lessons & Topics (2–3 hours/day):
MCP Overview (Day 1): 2026 standard for context/tool routing.
Server Setup (Day 2): Python/FastAPI or Supabase Functions.
Protocol Handling (Day 3): Context passing; tool registration.
Security (Day 4): Encryption; auth tokens.

Resources:

Video (EN): MCP Protocol Explained - Understanding MCP.
Video (PT): Protocolo MCP Explicado - Entendendo MCP.
Video (EN): FastAPI Tutorial - Building API servers.
Video (PT): Tutorial FastAPI - Construindo servidores API.
Video (EN): Supabase Functions Tutorial - Serverless functions.
Video (PT): Tutorial Supabase Functions - Funções serverless.
Video (EN): API Security Best Practices - Encryption and auth.
Video (PT): Melhores Práticas de Segurança API - Criptografia e autenticação.
Video (EN): Context Passing in APIs - Managing context.
Video (PT): Passagem de Contexto em APIs - Gerenciando contexto.
Spec (EN): MCP Spec (emerging 2026 docs via HF/LangChain).

Hands-On Activities (3–4 hours):
• Basic MCP server; connect agent.

Assignment: Agent using MCP for tool calls.

Week 3: Security & Scaling (Supabase Edge Functions)

Objectives: Build resilient, scalable integrations.

Lessons & Topics (2–3 hours/day):
Edge Functions (Day 1): Serverless Python in Supabase.
Auth/Security (Day 2): Row-level security; JWT.
Scaling Patterns (Day 3): Caching; rate limiting.
Monitoring (Day 4): Logs; error ethics.

Resources:

Video: Supabase Edge Functions Complete Guide - Serverless deployment.
Video: Row-Level Security Tutorial - Database security.
Video: JWT Authentication Tutorial - Token-based auth.
Video: Caching Strategies - Performance optimization.
Video: Rate Limiting Implementation - Protecting APIs.
Video: API Monitoring and Logging - Observability.
Guide: Supabase Functions Scaling Guide.

Hands-On Activities (3–4 hours):
• Deploy agent logic to Edge Functions.

Assignment: Secure integrated backend.

Week 4: Multi-Modal and Real-Time Systems

Objectives: Handle text/image/video; real-time flows.

Lessons & Topics (2–3 hours/day):
Multi-Modal Basics (Day 1): Vision/language models.
Real-Time (Day 2): WebSockets; Supabase Realtime.
Combined Systems (Day 3): Agent \+ multi-modal.
Final Ethics (Day 4): Consent for media; transparency.

Resources:

Video (EN): Multi-Modal AI Explained - Text, image, video models.

Video (PT): IA Multi-Modal Explicada - Modelos de texto, imagem, vídeo.
Video (EN): Vision-Language Models - Combining modalities.
Video (PT): Modelos Visão-Linguagem - Combinando modalidades.
Video (EN): WebSockets Tutorial - Real-time communication.
Video (PT): Tutorial WebSockets - Comunicação em tempo real.
Video (EN): Supabase Realtime Tutorial - Live data updates.
Video (PT): Tutorial Supabase Realtime - Atualizações de dados ao vivo.
Video (EN): Real-Time AI Systems - Building live AI apps.
Video (PT): Sistemas de IA em Tempo Real - Construindo apps de IA ao vivo.
Models (EN): HF Multi-Modal Models.

Hands-On Activities (3–4 hours):
• Add real-time/multi-modal to agent.

Assignment: Prepare milestone integration.

Month 8 Milestone Project

Project: Build/deploy an integrated AI system (e.g., Flutter/Next.js app \+ fine-tuned model \+ multi-agent via MCP \+ Supabase backend). Public live demo \+ GitHub. Include ethical/security notes. (6–8 hours total.) Rubric: Integration Complexity (40%), Security/Ethics (25%), Functionality (20%), Scalability (10%), Demo (5%). Mentor approval unlocks Month 9—your most advanced project yet\!

Systems connected—final month: capstones and career launch\!

PAIED Program Detailed Curriculum for Month 9 – Capstone Projects, Portfolio & Career Launch

PAIED Program: Detailed Curriculum for Month 9 – Capstone Projects, Portfolio & Career Launch

Month Overview This final month is your graduation showcase—synthesizing everything from Months 1–8 into 2–3 production-grade, AI-powered products. You'll plan, build, iterate, and launch full projects (e.g., a complete Flutter app with fine-tuned model, agents, MCP integrations, Supabase backend). True to our philosophy—"They may not believe the papers, but you will get paid for the working projects"—this month emphasizes shipping polished, public work that attracts real clients, employers, or investors.

You'll also build a professional portfolio site and learn strategies to monetize your skills. No new theory—just application and launch\!

Ethical AI Focus: Full ethical audit for capstones (bias checks, privacy, transparency); responsible pitching.

Resources Provided: All prior tools; portfolio templates. Weekly live showcases (recorded). Mentor 1:1 sessions. Community pitch practice.

Weekly Effort: 15 hours (non-workers) or 13 hours (workers). Focus on shipping.

Week 1: Project Ideation and Planning

Objectives: Choose impactful ideas; plan end-to-end.

Lessons & Topics (2–3 hours/day):
Idea Generation (Day 1): Namibia-relevant problems (e.g., tourism app with agent, farming advisor).
Scope & Feasibility (Day 2): Use prior skills; avoid over-engineering.
Project Planning (Day 3): Milestones, tech stack (Flutter/Next.js \+ Supabase \+ agents \+ tuned model).
Ethical & Impact Audit (Day 4): Inclusivity plan; sustainability.

Resources:

Video (EN): Project Planning Tutorial - Planning software projects.
Video (PT): Tutorial de Planejamento de Projetos - Planejando projetos de software.
Video (EN): Agile Project Management - Scrum and Kanban basics.

Video (PT): Gestão Ágil de Projetos - Conceitos básicos de Scrum e Kanban.
Video (EN): Tech Stack Selection - Choosing the right tools.
Video (PT): Seleção de Tech Stack - Escolhendo as ferramentas certas.
Video (EN): Project Scope Management - Avoiding scope creep.
Video (PT): Gestão de Escopo de Projeto - Evitando expansão de escopo.
Templates (EN): Notion Project Templates or Trello Templates for planning.
Examples (EN): Past PAIED examples (community showcase).

Hands-On Activities (3–4 hours):
• Brainstorm 5 ideas; select 2–3.

Assignment: Detailed plan (features, timeline, ethical notes); mentor approval.

Week 2–3: Full Development and Iteration

Objectives: Build, integrate, and refine capstones.

Lessons & Topics (2–3 hours/day across weeks):
Integration Mastery (Week 2 Day 1–2): Combine agents, fine-tuned models, MCP, multi-modal if relevant.
Polish & Features (Week 2 Day 3–4): UI from Month 2; advanced state/error handling.
Testing & Feedback (Week 3 Day 1–2): User testing; peer reviews.
Iteration (Week 3 Day 3–4): Fix issues; optimize performance/ethics.

Resources:

Video: Full-Stack Integration Tutorial - Connecting all components.
Video: User Testing Methods - Testing your products.
Video: Code Review Best Practices - Getting feedback.
Video: Performance Optimization - Optimizing apps.
Video: Error Handling Best Practices - Robust error handling.
Codebases: All previous months' codebases.
Guides: Vercel Deployment and Flutter Build Guides.

Hands-On Activities (4–5 hours/day):
• Daily builds; weekly demos to community.

Assignment: Weekly progress updates; mid-month demo.

Week 4: Portfolio Site, Launch, and Career Strategies

Objectives: Showcase work professionally; prepare for income.

Lessons & Topics (2–3 hours/day):
Portfolio Building (Day 1): Next.js/Flutter web site with project demos.
Launch Prep (Day 2): Live URLs, GitHub READMEs, demo videos.
Monetization (Day 3): Freelance platforms (Upwork); pricing projects; client pitches.
Next Steps (Day 4): Job applications; startup ideas; community contributions.

Resources:

Video (EN): Portfolio Website Tutorial - Building your portfolio.
Video (PT): Tutorial de Site Portfolio - Construindo seu portfolio.
Video (EN): Next.js Portfolio Template - Using templates.
Video (PT): Template Portfolio Next.js - Usando templates.
Video (EN): GitHub README Best Practices - Writing great READMEs.
Video (PT): Melhores Práticas GitHub README - Escrevendo ótimos READMEs.
Video (EN): Demo Video Creation - Creating demo videos.
Video (PT): Criação de Vídeo Demo - Criando vídeos demo.
Video (EN): Freelancing for Beginners - Starting as a freelancer.
Video (PT): Freelancing para Iniciantes - Começando como freelancer.
Video (EN): Upwork Tutorial - Using Upwork platform.
Video (PT): Tutorial Upwork - Usando a plataforma Upwork.
Video (EN): Pricing Your Services - Setting rates.
Video (PT): Precificação de Serviços - Definindo preços.
Video (EN): Client Pitching - Pitching your services.
Video (PT): Apresentação para Clientes - Apresentando seus serviços.
Templates (EN): Next.js Portfolio Starters or Flutter Web Templates.
Guides (EN): Freelancing guides (Namibia focus).

Hands-On Activities (3–4 hours):
• Build/deploy portfolio; final capstone launches.

Assignment: Complete portfolio; pitch recording.

Month 9 Milestone Projects

Projects: Launch 2–3 complete capstone products (e.g., full AI-powered mobile app with custom model \+ agents \+ MCP; web dashboard). Deploy publicly (live URLs/APKs). Build and launch a professional portfolio site showcasing all work with case studies, demo videos, and contact form.

Examples of Strong Capstones:
• Namibia tourism app (Flutter \+ fine-tuned local language model \+ agent for recommendations).
• Business automation suite (Next.js dashboard \+ multi-agent workflows).

Rubric: Innovation/Complexity (30%), Functionality & Polish (30%), Integration of Skills (20%), Ethical Documentation (10%), Portfolio Quality (10%).

Final Showcase: Virtual graduation event—present capstones live; receive feedback from mentors/peers.

Congratulations in advance—you've built a portfolio that gets you paid for working projects\! This is your launchpad to freelance, employment, or entrepreneurship. Ø=Þ€

Program Complete\! You've now mastered Practical AI Engineering & Development. Share your launches—we can't wait to celebrate\!

Conclusion Your Journey to Practical AI Mastery (PAIED Program)

Conclusion: Your Journey to Practical AI Mastery

Congratulations on completing the Practical AI Engineering & Development (PAIED) Program\!

Over 9 intense months, you've transformed from a complete beginner into a capable AI builder—launching real websites, mobile apps, agents, fine-tuned models, and integrated systems. You've embodied our core philosophy: "They may not believe the papers, but you will get paid for the working projects." Your public portfolio of launched, functioning projects is undeniable proof—ready to attract freelance clients, employers, or even investors in Namibia and beyond.

tiktok.com

instagram.com

facebook.com

africanshapers.com

You've mastered ethical AI use, prioritized tools like Flutter/Dart, Next.js, Supabase, and Python—and shipped products in hours using modern techniques. This isn't just skills; it's empowerment for Namibia's digital future.

thenewstack.io

facebook.com

ripenapps.com

startuptalky.com

toolfolio.io

What's Next?

Monetize: List projects on freelance sites; pitch to local businesses.
Contribute: Join AISOD X community; mentor others.
Grow: Build on capstones—turn one into a startup.
Celebrate: Share your portfolio (\PAIEDGraduate)—the world needs your Namibia-built AI solutions\!

You're now part of a new generation of African AI creators. The future is yours to build—and get paid for.

Thank you for trusting AISOD. Keep launching\!

Important Notice

This manual is provided as a comprehensive guide to the AISOD PAIED Program. For the most up-to-date information, live support, and interactive learning experience, please visit paied.aisodx.tech or contact AISOD Institute directly.

Remember: "They may not believe the papers, but you will get paid for the working projects."

Start your AI engineering journey today!