

# AI Builder Bootcamp: 7 Days to Create, Automate, and Launch Your Own AI Projects

## Course Overview

This immersive, live 7-day bootcamp empowers you to build real-world AI systems and prototypes from scratch. Gain hands-on experience in AI setup, voice agents, cloning, media generation, automations, autonomous agents, and no-code product development. By the end, you'll have a portfolio of AI projects, including a fully launched MVP.

**Target Audience:** Beginners to intermediate AI enthusiasts, developers, entrepreneurs, or professionals looking to integrate AI into their work.

**Prerequisites:** Basic computer skills; no prior coding required (though familiarity with Python is a plus for Days 6-7).

**Format:** Live sessions (2-3 hours/day) with demos, Q&A, and breakout groups; daily assignments for practical application.

**Outcomes:** Build deployable AI tools, understand ethical AI practices, and network with peers. Certificate of completion provided.

**Resources:** Access to a shared Notion workspace for notes, code snippets, and community forum.

# Day 1: AI Playground & LLMs Setup

## What Will You Learn?

1. Set up your local AI development environment
2. Host Large Language Models (LLMs) using Ollama and Bolt
3. Run AI models locally without relying on the cloud
4. Get familiar with your core playground for the week

## Tools

1. OpenAI (for cloud comparison).
2. Ollama (core local LLM host).
3. Bolt.new (web-based playground for quick tests).
4. Bonus: VS Code with AI extensions for a seamless IDE.

**Estimated Time:** 2 hours live + 1-2 hours assignment.

## Assignments

### 1. Install and configure Ollama on your machine:

- Step 1: Download Ollama from the official website ([ollama.ai](https://ollama.ai)) and install it for your OS (Windows/Mac/Linux).
- Step 2: Open a terminal/command prompt and run `ollama run llama3` to pull and start the model.
- Step 3: Test the installation by querying the model with a simple prompt like "Explain AI in one sentence." Note any errors and troubleshoot using the Ollama documentation.
- Expected Outcome: A fully functional local LLM setup. Submit a screenshot of your terminal showing a successful query response.
- Tips: Ensure your machine has at least 8GB RAM; if issues arise, check GPU compatibility for faster performance.

## **2. Run a sample LLM locally and experiment with its capabilities:**

- Step 1: Use Ollama to run a model like Llama 3 (command: `ollama run llama3`).
- Step 2: Test with at least 5 diverse prompts, e.g., text generation ("Write a short story about a robot"), summarization ("Summarize this article: [paste a short text]"), and translation ("Translate 'Hello' to French").
- Step 3: Document the responses in a Notion page, noting strengths (e.g., creativity) and limitations (e.g., hallucinations).
- Expected Outcome: A report on model performance. Share your Notion link in the forum for peer feedback.
- Tips: Experiment with parameters like temperature (for creativity) via Ollama's API if comfortable.

## **3. Build a custom AI playground in Bolt.new tailored to your use case:**

- Step 1: Sign up for Bolt.new and create a new playground.
- Step 2: Integrate your Ollama model or use a built-in one; customize prompts for a specific scenario, e.g., "Business idea generator" with inputs like industry and target audience.
- Step 3: Test the playground with 3-5 inputs and refine based on outputs.
- Expected Outcome: A shareable playground link. Post it in the group forum with a brief description of your use case.
- Tips: Use Bolt.new's templates for inspiration; focus on real-world applicability like content creation or brainstorming.

## **Day 2: Voice Agents + MCPs (Multi-Channel Personas)**

### **What Will You Learn?**

1. Build interactive AI voice assistants that respond to natural language.
2. Connect voice inputs to automated actions across apps (e.g., calendar integration).
3. Configure Multi-Channel Personas (MCPs) for consistent AI behavior on voice, text, or web.
4. Real-world example: Create a virtual receptionist for customer queries.
5. Ethical tip: Ensure voice cloning respects consent and avoids deepfakes.

### **Tools**

1. ElevenLabs (for voice synthesis and cloning).
2. VAPI (for voice API integration).
3. Bonus: Twilio for SMS/channel expansion.

**Estimated Time:** 2.5 hours live + 2 hours assignment.

### **Assignments**

1. **Create a personalized voice assistant using ElevenLabs and VAPI:**
  - Step 1: Sign up for ElevenLabs and upload a 1-minute voice sample to clone your voice (ensure it's clear audio).
  - Step 2: Set up a VAPI account and create a new voice agent, linking it to your ElevenLabs voice.
  - Step 3: Configure basic intents, e.g., greeting responses. Test by calling the agent via phone or web.
  - Expected Outcome: A working voice assistant. Record a 30-second demo call and upload to the forum.
  - Tips: Use free tiers initially; review ElevenLabs' ethics guidelines for voice usage.
2. **Design and implement voice-activated workflows:**

- Step 1: In VAPI, add actions like "Set reminder" that integrate with Google Calendar via API.
- Step 2: Define 3 workflows, e.g., weather check (pull from an API), note-taking (save to Notion), and simple Q&A.
- Step 3: Test each workflow with voice inputs and debug any recognition issues.
- Expected Outcome: Documented workflows with test logs. Share a flowchart or description in Notion.
- Tips: Use natural language processing best practices; handle accents by adjusting sensitivity settings.

### 3. Build an MCP that operates across platforms:

- Step 1: Define a persona (e.g., "Friendly Tech Support Bot") with consistent traits like tone and knowledge base.
- Step 2: Integrate with multiple channels: voice via VAPI, text via Slack/Webhook, and email via Zapier.
- Step 3: Test cross-channel consistency by querying the same question in each and comparing responses.
- Expected Outcome: A demo video showing multi-channel interaction. Post to the forum for review.
- Tips: Ensure seamless handoffs between channels; use tools like Twilio for expansion if needed.

## **Day 3: Build Your Own AI Clones**

### **What Will You Learn?**

1. Create personalized AI clones fed with your data for specialized tasks.
2. Deploy clones as support bots, content creators, or personal assistants.
3. Train models using structured (e.g., databases) and unstructured data (e.g., documents).
4. Progressive build: Integrate Day 1's LLM setup with custom data.
5. Ethical tip: Handle sensitive data securely to prevent biases.

### **Tools**

1. ChatGPT (for base model customization).
2. Notion (as a knowledge base for data ingestion).
3. Bonus: Pinecone for vector database storage.

**Estimated Time:** 2 hours live + 2 hours assignment.

### **Assignments**

1. **Create a custom AI clone using your own knowledge base:**
  - Step 1: Compile a knowledge base in Notion (e.g., 5-10 pages of personal notes, resumes, or articles).
  - Step 2: Use ChatGPT's custom GPT builder to upload and train on this data.
  - Step 3: Fine-tune prompts to reflect your "clone" personality.
  - Expected Outcome: A testable clone. Submit a link to your custom GPT.
  - Tips: Anonymize sensitive info; use structured formats like tables for better ingestion.
2. **Deploy your AI clone as a functional chatbot:**
  - Step 1: Use Streamlit or a similar tool to create a web interface for your clone.
  - Step 2: Integrate the custom GPT via API keys.
  - Step 3: Host on a free platform like Replit and test accessibility.

- Expected Outcome: A live chatbot link. Share in the forum with usage instructions.
- Tips: Add user authentication if dealing with personal data.

### **3. Train your clone to handle specific tasks using custom data:**

- Step 1: Identify 3 tasks (e.g., FAQ answering, content summarization).
- Step 2: Upload additional data (structured CSV or unstructured PDFs) and retrain.
- Step 3: Evaluate with 10 test queries, scoring accuracy (e.g., 1-5 scale). Refine as needed.
- Expected Outcome: A performance report. Upload to Notion for peer review.
- Tips: Use tools like Pinecone for scalable retrieval; monitor for biases in responses.

## **Day 4: AI Video, Audio & Image Generation**

### **What Will You Learn?**

1. Generate AI-powered videos, voiceovers, and animations from text prompts.
2. Clone voices and create realistic avatars for engaging content.
3. Design visuals and creative assets, including style transfer and editing.
4. Real-world example: Produce a marketing video for your Day 3 clone.
5. Ethical tip: Attribute AI-generated content and avoid misleading representations.

### **Tools**

1. MidJourney (for image generation).
2. Runway (for video AI).
3. Pictory (for video editing from text).
4. D-ID (for avatar creation).
5. ElevenLabs (for audio).
6. Bonus: CapCut AI for quick edits.

**Estimated Time:** 3 hours live + 2-3 hours assignment.

### **Assignments**

1. **Create a short AI-generated video (30-60 seconds) using custom prompts:**
  - Step 1: Craft a detailed prompt in Runway or Pictory (e.g., "A 45-second explainer on AI ethics with futuristic visuals").
  - Step 2: Generate and edit the video, adding transitions or text overlays.
  - Step 3: Export and review for quality.
  - Expected Outcome: Uploaded video file. Share link in forum.
  - Tips: Iterate prompts for better results; use free credits wisely.
2. **Build a system to clone images/avatars with specific parameters:**
  - Step 1: In D-ID or MidJourney, define parameters (e.g., age, style, emotion).

- Step 2: Generate 3 variations and select the best.
- Step 3: Integrate into a simple script or tool for reuse.
- Expected Outcome: A set of cloned images with parameter docs.  
Post to Notion.
- Tips: Ensure ethical use; test on public domain images first.

### 3. **Generate audio content with controlled voice characteristics:**

- Step 1: Use ElevenLabs to clone a voice and adjust traits (e.g., accent, speed).
- Step 2: Create a 1-minute audio script (e.g., podcast intro).
- Step 3: Combine with Day 1 LLM for script generation and test playback.
- Expected Outcome: Audio file with notes on controls. Upload for feedback.
- Tips: Balance realism with ethics; use noise reduction tools.

## Day 5: AI Automations Deep Dive

### What Will You Learn?

1. Build full automation stacks to orchestrate complex workflows.
2. Automate tasks like content creation, email outreach, or data analysis.
3. Use triggers for event-driven AI actions (e.g., new email → summarize → respond).
4. Progressive build: Automate outputs from previous days (e.g., generate media on schedule).
5. Ethical tip: Monitor automations for errors to prevent unintended actions.

### Tools

1. Zapier (for no-code integrations).
2. Make (advanced automation).
3. Notion (data hub).
4. Google Sheets (for data triggers).
5. Bonus: Airtable for database automations.

**Estimated Time:** 2.5 hours live + 2 hours assignment.

### Assignments

1. **Design and implement an end-to-end automation workflow:**
  - Step 1: Map out the workflow (e.g., form submission → AI analysis → notification).
  - Step 2: Build in Zapier: Connect apps like Google Forms, ChatGPT, and email.
  - Step 3: Test with sample data and log results.
  - Expected Outcome: Active Zap link and test report. Share in a forum.
  - Tips: Use error handling branches.
2. **Create a content generation and distribution pipeline:**
  - Step 1: Set up triggers (e.g., daily schedule).
  - Step 2: Integrate Day 1 LLM for generation, then distribute to social/email.

- Step 3: Run a full cycle and evaluate output quality.
- Expected Outcome: Pipeline demo video. Post to Notion.
- Tips: Add approval steps for quality control.

### **3. Set up trigger-based automations for business processes:**

- Step 1: Identify a process (e.g., lead scoring).
- Step 2: Use Make to build with conditions and loops.
- Step 3: Simulate 5 triggers and refine.
- Expected Outcome: Documentation and shareable link. Upload for review.
- Tips: Integrate with Google Sheets for data logging.

## **Day 6: Building Autonomous AI Agents**

### **What Will You Learn?**

1. Understand agent architectures, including reasoning loops and tool integration.
2. Build agents for multi-step tasks like research or decision-making.
3. Deploy agents for scraping, summarizing, or analysis (with legal considerations).
4. Progressive build: Enhance Day 5 automations with agent intelligence.
5. Ethical tip: Ensure agents comply with web scraping laws and data ethics.

### **Tools**

1. AutoGPT (for autonomous task execution).
2. LangChain (for chaining LLMs).
3. CrewAI (for multi-agent collaboration).
4. 9Cloud (cloud deployment).
5. Bonus: Hugging Face for model hosting.

**Estimated Time:** 3 hours live + 3 hours assignment.

### **Assignments**

1. **Build an autonomous research agent using AutoGPT:**
  - Step 1: Install AutoGPT and configure it with your Day 1 LLM.
  - Step 2: Define a research task (e.g., "Gather top 5 AI trends in 2025").
  - Step 3: Run the agent, monitor steps, and compile output.
  - Expected Outcome: Research report. Share in a forum.
  - Tips: Set task limits to avoid infinite loops.
2. **Create a multi-step workflow agent for task automation:**
  - Step 1: Use CrewAI to define agents (e.g., researcher + summarizer).
  - Step 2: Build workflow (e.g., input → research → output).
  - Step 3: Test with variations and debug.
  - Expected Outcome: Agent code/script and demo. Post to Notion.

- Tips: Use LangChain for chaining.

### **3. Implement a data processing agent with LangChain:**

- Step 1: Prepare a dataset (e.g., CSV from Google Sheets).
- Step 2: Build agent to process (e.g., clean, analyze, summarize).
- Step 3: Run on sample data and evaluate accuracy.
- Expected Outcome: Processed data file and report. Upload for feedback.
- Tips: Handle API rate limits; ensure compliance.

## **Day 7: Vibe Coding — Build Without Writing Code**

### **What Will You Learn?**

1. Ship products rapidly using no-code platforms for UI/UX design.
2. Integrate GPT, APIs, logic flows, and automations visually.
3. Build and launch your own AI product in one day as a capstone.
4. Capstone: Combine all days into an MVP, e.g., an AI-powered app.
5. Ethical tip: Test for accessibility and user privacy in your product.

### **Tools**

1. Softr (for app building).
2. Bubble (visual programming).
3. Make (integrations).
4. Framer (prototyping).
5. Bonus: Adalo for mobile apps.

**Estimated Time:** 3 hours live + 4 hours assignment (capstone focus).

### **Assignments**

- 1. Build a functional MVP using no-code tools:**
  - Step 1: Choose a platform like Bubble and design UI (e.g., dashboard for AI clone).
  - Step 2: Add core features from prior days (e.g., voice input).
  - Step 3: Test functionality end-to-end.
  - Expected Outcome: MVP prototype. Share link.
  - Tips: Use templates for speed.
- 2. Integrate AI capabilities into your no-code application:**
  - Step 1: Connect APIs (e.g., ChatGPT, ElevenLabs).
  - Step 2: Add logic flows (e.g., user query → AI response).
  - Step 3: Debug integrations.
  - Expected Outcome: Integrated app demo. Post video.
  - Tips: Use Make for complex flows.
- 3. Launch a complete product by the end of the day:**

- Step 1: Host on a domain (e.g., free Bubble subdomain).
- Step 2: Add user features like sign-up.
- Step 3: Gather feedback via forum poll.
- Expected Outcome: Launched product with launch notes. Share for review.
- Tips: Ensure mobile responsiveness.

## **Course Wrap-Up**

- Review key takeaways and portfolio building.
- Q&A session on career applications of AI skills.
- Next steps: Join the alumni community for ongoing projects and updates.