

Jacob Valdez

Data/ML Engineering, Robotics, Full-Stack

jacob@commandagi.com +1 (469) 968-9490 jvboi.dev github.com/JacobFV @jvboi

Independent ML & Robotics Research · Jul 2026 – Present

Developing [morphology-agnostic, contact-centric encoder-diffusion robotics control policy](#) in collaboration with [TalOS Robotics](#) (Aug 2026 – Present)

Developed/trained [IBM-1](#) topologically-constrained transformer foundation model for [SuperCognition Labs](#) (Jul 2026 – Present)

API / Integration Architect · [AGI, Inc.](#) · Jan 2026 – Apr 2026

Interfaces and integration across our API, SDKs, and other partner-facing surfaces. iOS, on-device llms, model quantization, agents, control plane, etc.

Owned api.agi.tech. First-responder to API infra / full-stack backend issues (Team across 3 countries/timezones, constant pushes directly to production)

Software Engineer · [AGI, Inc.](#) · Oct 2025 – Jan 2026

full-stack + infrastructure engineering across agentic systems: typescript, python, react/next.js, node.js, apis, distributed systems, cloud infrastructure, aws, gcp, s3, data engineering, containers, drizzle, pg/psql, databases, queues, browser/computer use, agent runtimes, tool use, model inference, llm integrations, multimodal models, evals, automation, control planes, observability, debugging, deployment, ci/cd, production operations. shipped rapidly across the stack in a high-velocity, production-first environment.

Software Engineer · [Breezy](#) · Apr 2025 – Oct 2025

Full-stack product engineering across a rails/next.js monorepo for a prosumer (single-person business) AI receptionist platform.

Independently owned features end-to-end from product requirements and implementation through integration, debugging, and production delivery.

Worked across frontend, backend, ai integrations, and application infrastructure; Left for infra role at AGI Inc

Applied Machine Learning Engineer · [Deepshard](#) ·

Sep 2024 – Dec 2024

Hardware and software for Truffle-1, a personal AI computer running models on-device: enclosure iteration across SLA, FDM and SLS printing against thermal constraints, alongside on-device ML work.

Full Stack Pipeline Engineer · [FLORA](#) · Jun 2024 – Jun 2024

typescript, python, modal, fal.ai, generative ai, diffusion models, computer art, confy ui, react, serverless architecture

Humanoid Robot Prototyping · Jan 2023 – Sep 2024

Prototyped hydraulically actuated, endoskeletal humanoid robot: KiCAD, FreeCAD, Blender, Python, 3D printing, mdf board CNC routing, 3/16" A16 plasma cutting, 100um + 300um trace PCB fabrication and SMT assembly (LCSC), 3018 mdf milling. Supplier outreach; worked with Dakings Rapid. Pre-llm engineering.

Software Engineer · [Motio, Inc.](#) · Aug 2022 – Jan 2023

Developed Soterre for Qlik Sense

Software Engineer (Intern) · [Motio, Inc.](#) · Jun 2022 – Aug 2022

Developed Soterre for Qlik Sense

Software Developer · [IT Lab](#) · [UT Arlington](#) · Jun 2021 – May 2022

Flask-based statistical visualization tool CoWiz; research group collaboration

Full stack Next-React-GraphQL stack: MLN Dashboard

Software Developer · [College of Social Work](#) · [UT Arlington](#) ·

Jun 2021 – May 2022

Maintain and enhance multi-platform (iOS and Android) data collecting application MyAmble using flutter and firebase and web administrator interface

Apr 2020 – Jun 2021

Pandemic + first year at the University of Texas at Arlington

Crew Trainer · [McDonald's](#) · May 2016 – Mar 2020

Led 30-minute monthly safety meetings

English and Spanish

B.S., Computer Science · [The University of Texas at Arlington](#) ·

2020 – 2022

CS coursework alongside heavy lab work, independent ML/robotics prototypes, and a steady research output. GPA 3.6/4.0.

A.A.S., Mathematics · [Navarro College](#) · 2016 – 2018

Math associate degree taken dual-credit during high school. GPA 3.9/4.0.

PROJECTS

TensorCode · Apr 2022 – Dec 2022, Sep 2026: **PROBLEM** Turning Python objects into tensors and back takes hand-written glue that breaks whenever a type changes. **TASK** Open-source neurosymbolic framework for Python and TypeScript that traces any value through ordinary code and trains only the learned steps. **OUTCOME 89.4% on banking77 after 3.3s of CPU training**; published on [PyPI](#) and [npm](#). [TensaCo/tensacode](#) [⊕](#) [tensorcode.dev](#) [⊕](#) [pypi/tensorcode](#) [⊕](#) [npm/tensorcode](#) · python, typescript, pytorch, differentiable programming

Tensegra · Sep 2026: **PROBLEM** Agents reuse stale work, ignore valid tool results and waste compute, and one score can't say which failure happened. **TASK** PyTorch modules for graph-conditioned attention and protected symbolic execution, with causal diagnostics at each model-tool boundary. **OUTCOME 570-run** attention study; dependency-aware policies reached **0.93–0.98 sealed IID success** with near-zero invalid reuse. [⊕](#) [JacobFV/tensegra](#) [⊕](#) [huggingface.co](#) · python, pytorch, typed graphs, recurrent computation

Computerworld · Sep 2026: **PROBLEM** Computer-use agents train on real VMs that can't be replayed or cheaply forked, so RL is slow and bugs don't reproduce. **TASK** One deterministic Rust engine simulating machines, shells, apps, a browser and an internet, shipped native, on PyPI and on npm. **OUTCOME** 265 µs fork, 4.78 µs step across 5 machines, 93 services and 87 sites, with cross-platform pixel parity. 🐦 JacobFV/computerworld 📦 jacobfv.github.io/computerworld 🔄 pypi/computerworld 🔄 npm/computerworld · rust, wasm, pyo3, wasm-bindgen, cpu rasterizer, simulation

IBM-1 · Sep 2026: **PROBLEM** EEG, MEG and vision data are rarely trained together, so each brain model sees only one slice. **TASK** One shared 3.84M kernel lazily materialized into 16 models across EEG, MEG, vision and a musculoskeletal body. **OUTCOME** 63.5% image-to-EEG retrieval (127× chance); the body stays upright with 0.17 mm drift under a 5 N push. 🐦 Super-Cognition-Labs/IBM-1 🗨 Talk 📦 super-cognition-labs.github.io/IBM-1 · python, mne, opensim

langcurriculum · Aug 2026: **PROBLEM** Fixed reasoning benchmarks leak into training data, so scores can't separate reasoning from memorization. **TASK** Zero-dependency procedural curriculum: 180 self-grading lesson generators, rendered in 411 languages and five media. **OUTCOME** Unlimited contamination-free episodes **without using any data**, gated by 4,026 tests. 🐦 JacobFV/formal-language-curriculum 📦 jacobfv.github.io · python, sqlite, unification grammar, dsp

jterm · Aug 2026: **PROBLEM** When a machine crashes, terminals lose the command you typed but never ran. **TASK** Mirrored every readline edit to crash-safe snapshots, with tmux-style splits that never restart the shell. **OUTCOME** Survives SIGKILL with scrollbar and unsent commands intact; 155 tests, installers for macOS, Windows and Linux. 🐦 JacobFV/jterm 📦 jacobfv.github.io/jterm · rust, tauri, typescript, react, xterm.js, codemirror, three.js

SC-WBD · Aug 2026: **PROBLEM** EEG, fMRI, stimulation and behaviour each see a different slice of the brain, so their results rarely combine. **TASK** 414-region whole-brain dynamics model fusing all four through physics forward operators, released behind a claim gate. **OUTCOME** Four public checkpoints; the 26.3M-param SC-WBD-003 **beat every baseline on 27 held-out participants**. 🐦 JacobFV/sc-wbd 📦 sc-wbd.pages.dev · pytorch, python, mne, nibabel, e-field solvers

fieldratchet · Jun 2026: **PROBLEM** PCB layout is done by hand, and none of that intuition compounds back into the tool. **TASK** Self-improving layout compiler whose oscillator placement engine mutates itself, gated by an independent EMI solver. **OUTCOME** Beat simulated annealing by 78% on a 187-component board; an evolved optimizer improved an unseen board ~24%. **Finalist in AI Engineer World's Fair Hackathon 2026**. 🐦 JacobFV/fieldratchet · typescript, remotion, electron, kicad, mojo/max, kuramoto oscillators, emi simulation

canvas-engineering · Jan 2026 – May 2026: **PROBLEM** World models hide their causal structure in weights, so nobody can check what is allowed to influence what. **TASK** Compiler that lowers a typed schema of latent regions, connectivity and clocks into attention masks on a stock diffusion transformer. **OUTCOME** Across 236 runs, matched dense accuracy with 19.6% of the connections and ~5× faster training. 🐦 commandAGI/canvas-engineering 🗨 Post · python, pytorch, diffusion transformers, cogvideox, pydantic, world models, neurosymbolic, research

SynthUX · May 2026: **PROBLEM** Computer-use agents need grounded training data, but recording full VMs is expensive. **TASK** Generator that expands goals into grammar trees and replays them as real input in three browser desktop simulators. **OUTCOME** Published a [Hugging Face dataset](#) of 1,949 frames: 81.1% app-native, 0 validator errors. 🐦 JacobFV/synthux · python, node tree grammars, svelte simulators, hugging face datasets, computer-use agents, synthetic data

browser-os · windows-web-next · macos-web-next · Nov 2025 – May 2026: **PROBLEM** Computer-use agents need desktops that are inspectable and instantly resettable, which real OSes aren't. **TASK** Svelte desktop shell (windows, filesystem, app lifecycle) with Windows 11 and macOS simulations and 20+ apps. **OUTCOME** Live on Vercel as SynthUX's substrates: 100% app-native on macOS, 81.1% across all three. 🐦 JacobFV/browser-os 🐦 JacobFV/windows-web-next 🐦 JacobFV/macos-web-next 📦 desktop-shell.vercel.app 📦 windows-web-next.vercel.app 📦 macos-web-next.vercel.app · typescript, svelte, vercel, html2canvas, computer-use agents

Chem-0 · May 2026: **PROBLEM** A real chemistry bench brings calibration, occlusion and contact that no API abstracts away. **TASK** MCP stack letting an agent drive a low-cost SO-101 arm from a live camera, with IK, YOLO vial analysis and an Electron console. **OUTCOME** The agent posed a real arm from camera feedback; the failed vial grasp pinned the blockers: calibration, compliance, fast local control. 🐦 JacobFV/chem-0 🗨 Post · lerobot so-101, python, opencv, placio, yolo, mcp, typescript, electron, robotics

Trash Sorter · May 2026: **PROBLEM** Trash sorting is a valuable robot job, once calibration, perception and grasping hold up. **TASK** Calibrated, teleoperated and tuned a LeRobot SO-101 arm with camera perception to pick and sort trash. **OUTCOME** Prototype built by a team of four in one weekend at [Founders Inc Physical AI Hack](#). · lerobot, so101 arm, computer vision, robotics, teleoperation

Lunar Rover · Apr 2026: **PROBLEM** Off-world robots can't be joysticked, so autonomy has to survive relays and ground control. **TASK** Regolith-collecting rover on a four-node stack (viewer, ground control, satellite relay, robot) with LLM route planning. **OUTCOME** Working rover and mission demo in one weekend at [RoboHacks at Y Combinator](#). 🗨 Post · robotics, llm route planning, mars robots, distributed control, autonomy

The Tensor Computer · Feb 2026: **PROBLEM** LLM agents spend 20–50 tokens on what one machine instruction does; could programs be learned as machine code instead? **TASK** Rebuilt a full von Neumann computer (ALU, cache, virtual memory, GPU) as differentiable JAX tensor ops. **OUTCOME** 4 independent backends agree on 59 conformance vectors; a self-hosting C compiler matches the host's output byte for byte. · python, jax, c, rust, c++, differentiable programming, program synthesis, research

precisionBOM · Jan 2026: **PROBLEM** Hardware teams source parts by hand across spreadsheets and vendor pages. **TASK** Next.js app over a Python multi-agent service that matches each BOM line against DigiKey's live catalog. **OUTCOME** Shipped [precisionbom.com](#): live stock, price breaks and alternates in one pass, exported as an orderable BOM. **First place (\$1000) at AI Agents & MCP Hardware Hackathon**. 🐦 precision-bom/precisionBOM 📦 precisionbom.com · next.js, python, multi-agent systems, digikey api, hardware procurement

bsbr · Mar 2025: **PROBLEM** Attention cost grows quadratically with sequence length, making long context expensive. **TASK** Pip-installable attention: exact within chunks, retrieval over compressed chunk states, plus a GPT-2 converter. **OUTCOME** **5.6× faster than linear attention** and **30× faster than DeltaNet** at 1,024 tokens.  JacobFV/bsbr  Post  pypi/bsbr · python, pytorch, transformers, hugging face, pypi, research

jnumpy · Sep 2021 – Nov 2021: **PROBLEM** UT Arlington ML courses banned third-party frameworks, so every assignment meant rewriting backprop. **TASK** Built my own numpy deep-learning library with arbitrary-DAG autodiff and neural-network and RL abstractions. **OUTCOME** Carried my homework across **three ML courses**; later added connect-4 RL and a C port.  JacobFV/jnumpy · python, numpy, autograd, deep learning, reinforcement learning

DesparadosAEye · Jan 2021 – May 2021: **PROBLEM** Mobile chatbots need cloud servers that cost money and expose user data. **TASK** Led a 5-person team porting Facebook's BlenderBot-small to TFLite for on-device Android inference. **OUTCOME** **335M-parameter chatbot running fully on-device**; presented at UTA Innovation Day 2021.  kmosoti/DesparadosAEYE · kotlin, java, sqlite, android, tensorflow lite, blenderbot, hugging face

SKILLS (ATS)

Languages: Python, TypeScript, SQL, Rust, C, Java, Kotlin

AI / ML: deep learning, transformers, diffusion models, world models, reinforcement learning, foundation models, computer vision, LLMs, PyTorch, JAX, Hugging Face

Agents: AI agents, multi-agent systems, computer use, tool use, Model Context Protocol (MCP), evals

Robotics / Hardware: robot learning, imitation learning, manipulation, LeRobot, MuJoCo, embedded firmware, ESP32, PCB design, KiCAD, CAD, 3D printing

Infrastructure / Web: distributed systems, data pipelines, PostgreSQL, AWS, GCP, Cloudflare Workers, Vercel, CI/CD, React, Next.js, Node.js, FastAPI

Mobile: iOS, Android, React Native, Flutter