TIRTH PATEL

Software Engineer | AI Systems & Backend

Toronto, ON • 437-774-9178 • ptirth339@gmail.com • linkedin.com/in/tirthpatel610 • github.com/TirthPatel6104 • Eligible to work in Canada

PROFESSIONAL SUMMARY

Software Engineer (BSc Computer Science, Trent University, Dec 2026, GPA 4.0) who ships tested, measured systems. Built a full-stack forestry growth-modeling platform on authentic provincial data for client scientists at the Ontario Ministry of Natural Resources and Forestry (FastAPI, SQLAlchemy, React, scikit-learn) with a 17-algorithm leakage-aware benchmark. Author of two open-source AI-infrastructure tools, ContextBridge (portable LLM memory with hybrid retrieval, PII redaction and an offline evaluation suite) and Substrate (an MCP server for agent memory and permission-gated execution). Strong in Python, TypeScript, REST APIs, SQL and CI/CD.

EXPERIENCE

Software Developer, Industry Partner Project | Client: Ontario Ministry of Natural Resources & Forestry (MNRF) **Sept 2025 – Apr 2026**

- Built the **Forest Growth & Yield Analytics System**, a full-stack platform (Python, FastAPI, SQLAlchemy 2.0, Alembic, React 19, TypeScript, Vite) that fits, benchmarks and serves stand-level basal-area growth models for 13 species and forest-unit groups on real MNRF inventory data.
- Implemented the production relative-growth-rate regression engine with biological sign-plausibility checks on every fitted coefficient, exposing per-plot predictions and 5–25-year projections through **25 REST endpoints**.
- Designed a **leakage-aware model benchmark** comparing 17 algorithms (OLS, ridge, lasso, spline GAM, SVR, random forest, XGBoost, LightGBM, MLP, mixed-effects) under plot-grouped, temporal and naive cross-validation with paired t-tests vs. the production equation, split-conformal prediction intervals and permutation importance, publishing a JSON + Markdown leaderboard across all 13 groups.
- Engineered a portable data layer (SQLite, PostgreSQL, SQL Server behind one Alembic chain) and asynchronous training/benchmark jobs (202 Accepted + polling) so long fits never block the API.
- Established CI quality gates: **89 pytest tests** with coverage, flake8 + mypy, ESLint + TypeScript build, Playwright E2E with axe accessibility scans, an Alembic migration sanity check and a benchmark smoke test on GitHub Actions.
- Worked directly with MNRF forestry scientists as clients, translating growth-and-yield requirements into specifications; wrote C4 system-design, benchmarking and security-disclosure documentation.

PROJECTS

ContextBridge — Portable, Privacy-Conscious Memory for LLMs | github.com/TirthPatel6104/ContextBridge v0.2.0 · MIT

Python 3.11+, Pydantic 2, FAISS, SQLite, Flask, Click, Chrome Extension (MV3), GitHub Actions

- Built an open-source tool that extracts structured memory from ChatGPT/Claude conversations into six typed categories, redacts secrets and PII before storage, and renders model-specific prompts (XML, Markdown, plain text), with versioned packages, rollback and conflict-flagged merging; one service layer serves a CLI, a local dashboard and a Chrome extension. **201 offline tests**, ruff, CI on Python 3.11–3.13.
- Implemented **explainable hybrid retrieval** (BM25 with optional embedding blending) reporting matched terms, score and reason per item; cuts injected context by **93.2% vs. the raw transcript** and 81.5% vs. full memory.
- Designed an **offline evaluation suite** (cb eval) measuring retrieval precision@3 / recall@3 / MRR (76.7% / 93.3% / 86.7%), duplicate-detection F1 (75.0%) and redaction recall (100%, 0 false positives); regression floors are asserted in CI.

Substrate — Local-First Intelligence Layer for AI Agents | github.com/TirthPatel6104/substrate v0.1 · MIT

Python (standard library only), SQLite FTS5, Model Context Protocol, JSON-RPC over stdio

- Built an **MCP server exposing 17 tools** that give any coding agent (Claude Code, Codex, custom) persistent typed memory with provenance, incremental file indexing with hybrid FTS5 + vector recall, and a durable task ledger with lease-based locking for multi-agent handoff; zero network calls by default.
- Implemented a deterministic, LLM-free **command-safety gate** (SAFE / NEEDS_CONFIRMATION / HARD_BLOCK) with a persistent approvals queue and audit log; regression tests pin known bypasses (command substitution, -c payloads, newline splitting) shut. 58 tests, CI on Python 3.10–3.12.

Privacy Bridge — HackTrent 2025 | *Next.js 16, React 19, TypeScript, Solana Web3.js, Supabase*

- Shipped a hackathon MVP of a cross-chain bridge executing real Solana wallet transactions (Phantom, Solflare, Backpack via Wallet Standard) and reconciling them against a Supabase transaction ledger with Explorer-linked history.

SKILLS

Languages: Python, TypeScript, JavaScript, SQL, Java, C#, Bash
Backend & APIs: FastAPI, Flask, Pydantic 2, SQLAlchemy 2.0, Alembic, REST API design, OpenAPI, background jobs, JSON-RPC
AI / LLM Engineering: RAG, BM25 + embedding retrieval, FAISS, LLM integration (OpenAI, Anthropic, Ollama), Model Context Protocol (MCP), prompt engineering, LLM evaluation (precision@k, MRR), PII redaction
Machine Learning & Data: scikit-learn, XGBoost, LightGBM, statsmodels, pandas, NumPy, grouped cross-validation, conformal prediction
Frontend & Databases: React 19, Next.js, Vite, Tailwind CSS, Radix UI, Recharts; PostgreSQL, SQLite (FTS5), SQL Server, Supabase
DevOps & Practices: Git, GitHub Actions CI/CD, Docker, pytest, ruff, mypy, ESLint, Playwright, axe; SDLC, Agile, code review, TDD

EDUCATION

Bachelor of Science, Computer Science | Trent University, Peterborough, ON · GPA 4.0/4.0 **Expected Dec 2026**