Ayusman Avisek Nanda

B.Tech CSE student | low-level systems programming, compilers, Android, and reverse engineering

Keonjhar, Odisha | 8249823670 | ayusman.a.nanda@gmail.com

github.com/AyusmanNanda | linkedin.com/in/ayusman-avisek-nanda-112a422b0

EDUCATION

Government College of Engineering, Keonjhar — Expected 2027

B.Tech, Computer Science and Engineering — CGPA 8.53 (till 6th sem)

Class XII: D.D. Higher Secondary School, Keonjhar, 2022 — 61.5%

Class X: D.N. High School, Keonjhar, 2020 — 69%

EXPERIENCE

Intern, Velotech Innovations Pvt. Ltd. — 14 May 2026 – 14 Aug 2026

- Built Patient Router (see Projects) end to end: synthetic data, model training and evaluation, Flask API and React dashboard.

PROJECTS

MDDS: Macular Disease Detection *Kotlin, Jetpack Compose, TensorFlow Lite (in progress, 7th-sem, team of 5, team lead)* — GitHub org

- Each member trains their own model to classify retinal OCT images as CNV, DME, DRUSEN or NORMAL. I trained a bidirectional Mamba + CNN hybrid (96.7% accuracy) and an AutoKeras model (92.4%).
- Built the Android app with a teammate: runs a TensorFlow Lite model on-device and flags low-confidence results as invalid.
- Built the model manager: pulls the latest model from GitHub Releases, verifies its SHA-256 hash, updates without an app release.

Axiom: College Management System *React, Node.js, PostgreSQL (6th-sem team project, team of 5, team lead)* — GitHub · Live demo

- Web platform for Admin, Faculty and Student roles. I built the Admin section: student, faculty and course management, attendance and marks, and Excel bulk import.
- Release builds for web, Android (Capacitor) and Windows/Linux (Electron) from one codebase, with GitHub Actions CI/CD (10+ tagged releases).
- Built authentication and login, including Google OAuth on web, Android and desktop, each with a different sign-in flow.

Patient Router *Python, Flask, scikit-learn, React (internship project)* — GitHub · Live demo

- Suggests a hospital department from symptoms, vitals and history: Gradient Boosting model (picked after comparing seven models) trained on synthetic data, 98.8% test accuracy, 91.2% on hand-written edge cases, with emergency rules and an optional Gemini fallback. Includes a Flask API, React dashboard, correction feedback loop, Electron build and CI. Not for real clinical use.

TinyC Compiler *C++* — GitHub

- Compiler for a small C-like language (int, arithmetic, print): lexer, recursive descent parser, AST, IR, x86-64 assembly.

Frida Manager *Vite, TypeScript, React, Capacitor, Java, Magisk (v1.0.0 released)* — GitHub

- Connects a React UI to native code: a custom Java Capacitor plugin runs shell scripts as root (via su) to start, stop and check a native frida-server on rooted Android phones. Shipped as v1.0.0.
- Built the Magisk module that bundles frida-server with its start, stop and status scripts.

Transaction System *C/C++, Node.js* — GitHub

- Node.js/Express web UI drives a C/C++ banking CLI by spawning the compiled binary and talking to it over stdin/stdout; the CLI handles accounts, auth, transfers and file-based storage.

UniversalCompiler *C++, JSON (research prototype)* — GitHub

- Research prototype for a universal language and compiler abstraction layer: a custom .unic source plus a JSON config of syntax mappings is transformed into Java or Python, then compiled and run. C++ components handle config loading, alias/code processing and compile-and-run. Next stage: IR and LLVM backend.

SKILLS

Strong: C, C++, Java, Python, Git, Linux (daily driver) **Working knowledge:** Data Structures & Algorithms, JavaScript / TypeScript, React, Node.js and Express, REST APIs

Used in projects: Kotlin and Jetpack Compose, Flask, PostgreSQL, Capacitor, Electron, TensorFlow Lite, scikit-learn, x86-64 assembly, Ghidra and Apktool, Magisk modules, Linux kernel builds, GitHub Actions

OPEN SOURCE & SECURITY

- **Founder, InTab Tools:** student-run open-source org building InTab PDF, privacy-first PDF tools that run fully on-device (no uploads, no accounts). Early stage: web app (React, TypeScript) underway.
- **WhatsApp Android trojan analysis:** reverse engineered the APK with Ghidra and Apktool, found hardcoded credentials and API keys, and traced how it runs. GitHub.
- **Kali NetHunter:** contributed a kernel config for the miatoll device and compiled kernels; mentioned in the Kali Linux 2024.4 release shout-outs.