



Barath B

Android Engineer

✉ barathjack77@gmail.com

📅 Dec 2005 📍 Trichy

▶ [Playstore Developer ID](#)

🔗 [Portfolio](#) 🔄 [3dbarath](#)

🌐 [3dbarath](#) 📞 +91 8015931576

🎓 EDUCATION

B.E CSE

Government Collage of Engineering
Srirangam
Trichy
Expected 2027

CGPA: 7.6

Relevant Coursework:

- Computer Networks
- OOPs
- OS
- Algorithms
- Data Structures

📄 CERTIFICATES

- Mobile Application Development Using Kotlin
- DevOps
- Digital Marketing
- Kubernetes
- Personal Branding and Development for Entrepreneurs

👤 SUMMARY

Built and shipped 3 production apps — Android, Desktop, and backend — as a solo engineer still in college. Specializes in Kotlin Multiplatform and full-stack Android systems, from custom TCP protocols to distributed Spring Boot backends serving 1,200 daily users.

📁 PROJECTS

GCES Hostel Management System 🔗

Android + Spring Boot · Live on Play Store · Deployed · 1,200 daily active users
A Live Institutional App [@gces.edu.in](https://gces.edu.in) 🔗
Jan 2026 - Mar 2026

End-to-end institutional app — the only meal and hostel management system for a 1,200-student government college.

- **Multi-module Clean Architecture** — :app (Android), :server (Spring Boot), :common (shared Kotlin DTOs) — strict separation with Hilt DI scoping ViewModels, repositories, and singleton managers
- **Predictive meal engine:** students confirm/skip meals daily; ShedLock **distributed JDBC locks** guarantee atomic midnight state transitions across server instances — zero race conditions at scale
- **Caffeine in-memory cache + Bucket4j token-bucket rate limiting** absorb concurrent rush-hour bursts, holding API response under **sub-100ms** at peak load
- **JWT stateless auth** + Spring Security; PostgreSQL with **Flyway versioned migrations**; backend deployed on Render (JAR) and HuggingFace Spaces (Docker) with post-deploy /version health checks
- Firebase **FCM** (push), **Crashlytics** (stability), **Performance Monitoring** (latency); AdMob AppOpen + Banner monetization; custom **Circular Reveal** theme transition in Compose

Stack: Kotlin · Jetpack Compose · MVI · Hilt · Retrofit · Coroutines/Flow · Room · DataStore · Spring Boot 3.5 · PostgreSQL · Flyway · JWT · ShedLock · Bucket4j · Firebase · AdMob
[Live on PlayStore](#) 🔗

SKILLS

Mobile



Jetpack Compose · MVVM/MVI · Clean Architecture · Hilt/Dagger · Coroutines & Flow · Room · Retrofit + OkHttp · DataStore · Kotlin Multiplatform (KMP) · Compose Multiplatform · Firebase (FCM, Crashlytics, Performance)

Backend



Spring Boot 3.5 · Spring Security · PostgreSQL · Flyway · JPA/Hibernate · JWT Auth · ShedLock · Bucket4j · Caffeine Cache · Docker

DevOps & Tooling



Git · Fastlane · GitHub Actions · AUR packaging · Bash · SQLite · jarsigner · ADB

Languages



Kotlin · Java · Python · C#

IN PROGRESS

Smart Expense Tracker

Redefining financial management with a focus on local data ownership. Category-wise spending breakdowns; WorkManager budget-threshold alerts via local notifications

Kotlin · Jetpack Compose · Room · DataStore · Compose Charts · WorkManager

LANGUAGES

Tamil



English



ConnectLnx

Kotlin Multiplatform · Android / Linux / Windows Play Store · GitHub Releases · AUR · *Closed Beta Dec 2025 – Jan 2026*

- The fastest way to move files between your devices — no cloud, no account, no limits.
- Engineered a **custom binary TCP protocol** with parallel multi-socket streaming — files chunked and distributed across up to **5 concurrent connections** for maximum LAN throughput
- **Zero-config discovery** via mDNS (`_connectlnx._tcp`) on both Android (NsdManager) and Desktop (jmDNS); QR code pairing as network-restriction fallback
- Handshake phase negotiates **buffer size and parallelism count** per session; typed binary header per chunk tracks type · filename · size · chunkIndex · offset
- Entirely non-blocking networking stack — Coroutines on Dispatchers.IO, SharedFlow streaming Progress / Completed / Error from NetworkManager → ViewModel → UI with atomic multi-socket write coordination
- Single codebase ships **4 distribution targets**: Play Store (.aab), Linux (.deb), Windows (.msi via GitHub Actions), AUR (connectlnx-bin)

Stack: Kotlin Multiplatform, Compose Multiplatform, Ktor, Coroutines/Flow, mDNS, ML Kit, FileKit, multiplatform-settings

Live on PlayStore 

Available for Desktop (Linux & Windows) 

Aztrox Release

Custom CI/CD Engine · Bash + SQLite + Fastlane · Internal Tool
Apr 2026

A release system built because no existing tool understood the full Aztrox stack.

- **Unified CLI** managing the complete lifecycle (build → sign → distribute) for 3 live projects across **6 distribution targets**: Google Play, GitHub Releases, AUR, Render, HuggingFace Spaces, Firebase
- **LIFO atomic rollback registry** — pipeline failures auto-revert version bumps, gradle.properties, and git tags without manual intervention
- **Parallel build orchestration**: .deb + .aab built locally while a release/v{version} branch triggers GitHub Actions to build .msi simultaneously
- **Preflight health system** validates dependencies, keystore passwords, .env vars, and directory paths before any build runs; every release logged to SQLite with auto-generated HTML audit reports

Stack: Bash · SQLite · Fastlane · GitHub Actions · jarsigner · gh CLI · ADB