

# Nitin Bhat

nitinbhat972@gmail.com | 9149480470 | Jammu, Jammu and kashmir | nitinbhat972

## SUMMARY

Computer Engineering Graduate and quick learner interested in systems programming, developer tooling, and AI, with hands-on experience building terminal applications and developer tools. Experienced with C, C++, Lua, Python, and Go, while exploring Rust through practical projects. Interested in low-level software, modular architecture, and learning new technologies by building and experimenting with real-world tools.

## EDUCATION

|                                                                                                         |                                         |
|---------------------------------------------------------------------------------------------------------|-----------------------------------------|
| <b>B. Tech Computer Engineering</b> , Dr. Babasaheb Ambedkar Technological University, Lonere<br>7 CGPA | 2022 – 2026<br>Lonere, Maharashtra      |
| <b>HSC (12th Grade)</b> , Kendriya vidyalaya Nagrota<br>86%                                             | 2021 – 2022<br>Jammu, Jammu and kashmir |
| <b>SSC (10th Grade)</b> , Kendriya vidyalaya Nagrota<br>89.2%                                           | 2019 – 2020<br>Jammu, Jammu and kashmir |

## SKILLS

|                                                                 |                                                                             |                                                                                    |                                                                   |
|-----------------------------------------------------------------|-----------------------------------------------------------------------------|------------------------------------------------------------------------------------|-------------------------------------------------------------------|
| <b>Languages</b><br>C, C++, Python, Go, Lua, Bash,<br>Rust, Zig | <b>Systems &amp; Tools</b><br>Linux, Git, GitHub, Docker,<br>GitHub Actions | <b>Development</b><br>DSA, OOP, Problem Solving,<br>Shell Scripting, Coding Agents | <b>Interests</b><br>Systems Programming,<br>Developer Tooling, AI |
|-----------------------------------------------------------------|-----------------------------------------------------------------------------|------------------------------------------------------------------------------------|-------------------------------------------------------------------|

## PROJECTS

- cwal** 
- Developed a high-performance **C** utility that extracts color palettes from wallpapers and generates dynamic themes for terminal applications, editors, and desktop components.
  - Implemented **customizable theme generation** with support for dark/light modes, saturation, contrast, transparency, templates, and application-specific configuration.
  - Added **Lua scripting support for custom color-processing backends**, enabling users to extend palette generation with their own algorithms.
  - Implemented an **INI-based configuration system** for persistent user settings, output paths, backends, random wallpaper selection, and application reload commands.
  - Built support for **surgical configuration injection**, allowing generated colors to be inserted into existing configuration files without overwriting unrelated user settings.
  - Integrated **automatic application reloading and template-based configuration generation** for seamless theme updates across the desktop environment.
  - Designed a custom build workflow using nob** to keep the project lightweight and provide a programmable C-based build system.
- notfm** 
- Developed a **terminal-based file manager in C23** using the **Notcurses** library for a responsive text-based user interface.
  - Implemented a **multi-pane TUI** with keyboard-driven navigation for browsing and interacting with files and directories.
  - Designed a **modular architecture** to separate UI components, navigation, and file-management functionality.
  - Worked with low-level **filesystem operations and C programming**, including directory traversal and file/directory handling.
  - Built an interactive event-driven interface for efficient terminal-based file management.
  - Structured the application for easier extension and maintenance by keeping core functionality independent from the UI layer.
- svar** 
- Built a minimal **terminal-based synced lyrics viewer in Rust** while exploring the Rust programming language and **Ratatui**.
  - Implemented an interactive TUI with keyboard controls for scrolling, re-centering live lyrics, and refreshing lyric data.
  - Experimented with **asynchronous programming** using **Tokio** and Crossterm event streams to handle terminal input and application tasks.
  - Integrated HTTP/JSON-based data fetching using **Reqwest** and **Serde**.
  - Explored **D-Bus integration** with zbus to interact with system/media services.
- riff** 
- Built a terminal-based music player in **C++23** while exploring modern C++ concepts and experimenting with **FTXUI** for terminal UI development.
  - Integrated **mpv** for audio playback and implemented controls for play/pause, track selection, next/previous playback, and seeking.
  - Added playback progress tracking and **synced lyrics** using the **LRCLIB API**.
  - Integrated **libcurl** for network communication and **nlohmann/json** for JSON parsing.
- cwal.nvim** 
- Developed a **Lua-based Neovim colorscheme** that consumes palettes generated by **cwal**  and dynamically applies them across the editor.
  - Implemented **automatic palette reloading** by monitoring generated colors and reapplying highlights when the palette changes.
  - Added highlight support for **Neovim UI, Treesitter, LSP diagnostics, completion**, and plugins including **Bufferline** and **Lualine**.
  - Implemented a **CwalReload user event** and a **palette-driven architecture** that separates theme generation in cwal from theme application in Neovim.
- Smart India Hackathon**
- 2025 — Offline RAG Desktop Application**
- Contributed to an **offline Retrieval-Augmented Generation (RAG)** desktop application supporting local document ingestion and querying.
  - Worked with **vector search** to retrieve relevant document context for AI-generated responses.
  - Integrated **on-device LLM inference** to enable document-based AI assistance without relying on cloud inference.
  - Performed **testing and debugging** to identify and resolve issues across the application workflow.
- 2024 — SAR Image Colorization**
- Contributed to a **deep learning model for SAR image colorization** to improve visual interpretation of radar imagery.
  - Worked on **image-processing and research components** of the project, supporting the development and experimentation process.
  - Contributed to evaluating approaches for transforming SAR imagery into more visually informative representations.