

# Biswas Khatiwada

biswas06@iastate.edu | +1 (929) 665-4532 | Queens, New York  
[GitHub](#) | [LinkedIn](#) | [Portfolio](#)

## EDUCATION

Iowa State University  
Bachelor of Science in Computer Science

Expected December 2028  
Ames, Iowa

## TECHNICAL SKILLS

**Languages:** TypeScript, JavaScript, Python, Swift, SQL, Bash

**Frontend:** React, Next.js, SwiftUI, Tailwind CSS, accessibility

**Backend and data:** Node.js, FastAPI, REST APIs, GraphQL, WebSockets, PostgreSQL, Supabase, ChromaDB

**Tools and testing:** Git, Docker, Playwright, XCTest, pgTAP, Model Context Protocol (MCP)

## EXPERIENCE

**Open Source Engineer | Parable** | [Link](#)

June 2026 - Present

- Built and shipped an open-source Next.js catalog with 88 React components and 78 complete templates across HTML/JavaScript, Astro, Next.js, SvelteKit, and Vite.
- Published components through shadcn-compatible registry endpoints that resolve dependencies and install editable source into the consuming project.
- Built live previews, MDX documentation, reduced-motion behavior, and a stateless HTTP MCP endpoint with four typed discovery tools.

**Product Engineer | Mosaic** | RevenueCat Shipaton 2026 Hackathon | [Link](#)

August 2026

- Built a SwiftUI iOS hackathon product with invitation-only organizer and participant flows, a disposable camera, fixed reveal, widgets, and an on-device photo recap.
- Integrated RevenueCat Purchases with a custom SwiftUI payroll and Supabase UUID identity; gated premium controls on authenticated server reconciliation instead of client state.
- Designed Supabase Edge Functions, PostgreSQL Row Level Security, private media storage, and independent consent controls; validated behavior with 34 Swift tests, 48 pgTAP checks, and 14 XCTest UI flows.

## PROJECTS

**Forge AI | Full-Stack App Generator**

September 2026 - Present

- Building a Next.js and React platform that converts natural-language specifications into runnable full-stack applications with isolated development workspaces, live previews, streamed build logs, and an embedded browser editor.
- Implementing a TypeScript stream parser for Anthropic API events that validates file operations, blocks unsafe paths, writes files incrementally, and repairs incomplete output before execution.
- Designing an AWS control plane with PostgreSQL, SQS, S3, EC2 Auto Scaling, Firecracker isolation, 10-second health reconciliation, warm environments, and a 30-minute idle reaper.

**Relay | Multi-Provider AI Coding Workspace**

September 2026 - Present

- Building a Tauri desktop workspace with React, TypeScript, and a Rust process supervisor that orchestrates Codex CLI, Claude Code, and OpenCode through local credentials.
- Implementing one provider interface for session startup, streamed events, messages, approvals, interruption, restoration, and event replay across supported coding agents.
- Designing a Git worktree and branch for every thread, with terminal output, test results, inline diff review, commit and push controls, and one-click pull-request creation.

**Specter | GPU-Accelerated macOS Terminal**

September 2026 - Present

- Building a native macOS terminal with Swift, AppKit, SwiftUI, POSIX PTYs, and a Metal and CoreText rendering pipeline for windows, tabs, splits, profiles, and searchable scrollbar.
- Implementing a VT100/xterm parser, bounded screen buffer, Unicode grapheme handling, resize and signal propagation, and deterministic child-process cleanup.
- Creating conformance, PTY integration, fuzz, golden-rendering, and performance tests for escape sequences, high-throughput output, large scrollbar histories, and input-to-frame latency.