Gaurav Thakur

📞 945 984 9894 | 🌐 https://gauravthakur.in | in @thegauravthakur | ✉ gthakur581@gmail.com

Experience
March 2021 - Present

Zepto

Software Engineer II   October 2024 - Present

- Optimized the ZeptoNow web application's Core Web Vitals via multiple performance initiatives including **HTTP streaming** to reduce **TTFB**, migrating third-party scripts to **worker threads** to free the main thread, and refining JS bundling to ship only necessary code.
- Implemented **performance budgeting** to monitor and control initial JavaScript bundle sizes. Established automated PR blockers using GitHub Actions to prevent merges **exceeding budget thresholds**, and deployed a **remote Lighthouse pipeline** to block PRs that don't meet performance score requirements.
- Led the domain migration from **zeptonow.com** to **zepto.com** while preserving existing **domain authority** and **SEO rankings**. Implemented proper **canonical tags** and redirects, and ensured seamless functionality of all critical flows (authentication, checkout, etc.) across both domains using a controlled rollout strategy.
- Built and deployed a Rakhi microsite end-to-end as sole developer, from initial setup to deployment on S3 and serving via Cloudflare. Optimized **user experience** via close collaboration with design and product teams, implementing performance enhancements like **layout shift prevention**, strategic **preloading of pages and Lottie animations**, and smart keyboard input switching for PAN fields.
- Improved cart functionality to maintain **item persistence** across store changes while supporting **multiple delivery addresses**, preventing cart loss during location updates.
- Improved development workflow efficiency by integrating **Turbopack** for accelerated hot reload and adopting **pnpm** for faster dependency management with reduced disk usage.
- Ensured website reliability through New Relic monitoring and alerting while implementing cryptographic **request signing** to authenticate frontend-to-backend communications as an additional security layer.

Software Engineer I   November 2023 - October 2024

- Added several features in Zeptonow, such as dynamic header, search suggestions, Zepto Pass, and various other **server-driven widgets** that are displayed on different pages.
- Worked on migrating the codebase to the latest **Next.js 14.2** (at that time it was the latest version).
- Rewrote several key functional areas of the app, including search, post-search, product description page, etc.
- Migrated several pages to **React Server Components**, such as the product description page and buy-pass page.
- Actively participated in making key architectural decisions for the app.

JioMart

Software Engineer   August 2022 - November 2023

- Created a merchandising dashboard and a server-driven CX application using **NextJS** and **TailwindCSS**
- Implemented project and build setup for multiple **NPM libraries**, incorporating **auto versioning** and **release mechanisms** to automate package building and release, resulting in a 95% boost in deployment efficiency
- Implemented a search feature in JioMart, enabling users to quickly find their desired items. This entailed designing the feature's functionality (HLD and LLD) and developing client-side SDKs for seamless integration with the application.
- Brought significant improvements to the several codebases by introducing best practices and optimizing our development process
- Actively contributed as a speaker in "Inspect Element", an organisation-wide knowledge sharing platform designed to educate individuals about frontend technologies

MAQ Software

Software Engineer   March 2022 - August 2022

- Owned complete features from gathering information to handling deployment to the production
- Turned Figma mockups to performant functional components using **React** (with **Typescript**) and **TailwindCSS**
- Created several unit test cases with **RTL** and **Jest** to match the flow of actual users while using the application
- Owned various functional areas like QA, code, and UI/UX to assure the project met all standards
- Managed the entire project alone, including direct communication with the client

Associate Software Engineer   March 2021 - March 2022

- Turned Figma mockups to performant functional components using **React** (with **Typescript**) and **TailwindCSS**
- Created several unit test cases with **RTL** and **Jest** to match the flow of actual users while using the application

Education

Lovely Professional University

Bachelor of Technology in Computer Science and Engineering

- **Coursework**: Data Structures, Algorithms, Databases and MEAN stack

Skills

- **Languages**: HTML, CSS, JavaScript and TypeScript
- **Libraries and Frameworks**: NextJS, React, TailwindCSS, Zustand, React Server Components, Jest, React Query, React Native and NodeJS
- **Tools** : VIM

Projects

Ensite.io   ⎘

Technology Used : NextJS, TailwindCSS, Zustand, Partial Pre Rendering, Postgres, React Server Components

An intuitive web page builder that requires no coding, enabling you to craft stunning, responsive, and inclusive websites effortlessly

Key Technical Highlights:

- Built as a Next.js React Server Components (RSC) application with **multi-tenant architecture**, where each deployed website operates as a dynamic route (/site/[domain]/:sub-path) for the main application.
- Ensured that each user interaction feels responsive with very low I**nteraction to Next Paint (INP)** scores, providing instant feedback for every click or user action.
- Enabled rapid publishing: Websites **deploy** in **2-3 seconds** (or under **1 second** without new widgets) using **Incremental Static Regeneration (ISR)** for optimal efficiency.
- Optimized performance for both the admin panel and deployed sites; **ISR-generated sites** are fully static, served via **CDN** without active server rendering, reducing costs and latency (e.g., delivering from the nearest edge location like BOM for Indian users, even if the server is in the USA).
- Deployed website ships only the JavaScript and CSS for used components even if editor supports 1000s of components.
- Applied **CSS optimizations** using **atomic Tailwind CSS**, ensuring only necessary styles are included with shared classes (e.g., "text-red-700" appears once across widgets) for a minimal footprint.
- Leveraged **server-side streaming**, **parallel routes**, and caching for peak performance across nearly every page.