

# Ijabadeniyi Ifedayo

Web developer, UI/UX enthusiast, Writer.

Portfolio | [LinkedIn](#) | [GitHub](#) | Email: [ifedayoadeniyi@gmail.com](mailto:ifedayoadeniyi@gmail.com)

I am a highly dedicated and ambitious Frontend Engineer that loves to learn and build!

I am curious, impact-driven, detail-oriented, and thrive on collaboration and transparency. I am product-minded, care deeply about customer experiences, and believe in a future in which every company serves its customers with an experience that truly speaks to them.

I have 3 years of experience working with React, Next.js and TypeScript to write clean, maintainable, and testable code. I enjoy creative work that lets me exercise my design acumen, but also relish building CLI tools, and implementing backends using REST, GraphQL, and tRPC protocols (Big fan of the T3-stack).

## PROFESSIONAL EXPERIENCE

---

### **(Contract) Frontend Developer, BetaQuick**

*November 2021 - Present*

- Wrote scalable, performant, and accessible CSS, HTML, and JavaScript making up our suite of ReactJS-TypeScript reusable components.
- Worked closely with Product, UX/UI, and the Backend Engineering team to execute business deliverables.
- Encapsulated ubiquitous functionality using custom hooks, which improved reusability, and modularity of the codebase, resulting in increased speed and ease of refactors.
- Designed APIs for components – considering key details like balancing flexibility with design consistency, developer experience, and consistency across the framework.

### **(Part-time) Full-stack Developer, ZuriChat**

*January 2021 - September 2021*

- Delivered world-class user experiences and user interfaces while executing understandable, maintainable, and well-tested code using Jest and React-Testing-Library.
- Participated in code reviews and provided overall code quality suggestions, to help maintain the structure and quality of the codebase.
- Improved the performance and scalability of our GraphQL API, by applying data-loaders to field resolvers suffering from the N+1 query problem, resulting in a drastically reduced number of SQL queries hitting our database.
- Improved client performance and reduced the volume of unnecessary network requests executed per query and/or mutation, by taking advantage of Apollo-client's normalized caching methodology, resulting in a smoother and snappier user experience

### **Intern Web Developer (NodeJS), HotelsNG**

*June 2020 - October 2020*

- Designed, implemented, documented, and tested REST APIs that interfaced with web and mobile clients using Mocha, Chai, Swagger, ExpressJS, and MongoDB.
- Lead, and participated in code reviews for other interns.
- Enforced code quality using Prettier, ESLint, and pre-commit hooks.
- Launched a productivity tool that helps companies set and track goals across distributed teams.

## SKILLS

---

- **Core:** JavaScript, TypeScript, React, Next-Auth, GraphQL, Apollo-Client, TypeORM, Type-graphql, URQL, Next.js, Node.js, Git/GitHub/GitLab, Docker, PostgreSQL/MySQL, PlanetScale, MongoDB, Redis, Prisma, React-query, Shell scripting.
- **Testing & QA:** Jest, Cypress, React Testing Library (RTL), GitHub Actions.
- **Styling:** TailwindCSS, CSS-in-JS / Styled-Components.