

Devesh Kumar

Email: deves125@gmail.com

LinkedIn: in/deveshcodes

Portfolio: devesh.codes

Mobile: +91-829-5457-993

Github: github.com/d-e-v-esh

SKILLS SUMMARY

- **Languages:** TypeScript, JavaScript, SQL, Go, Java
- **Frameworks:** React, Next.js, GraphQL, Framer Motion, Node.js, SASS, Android, Redux
- **Tools:** Git, GitHub, Linux, PostgreSQL, MongoDB, Redis, Docker, VSCode, VIM
- **Soft Skills:** Leadership, Writing, Public Speaking

EXPERIENCE

- **Google Summer of Code - Open Bioinformatics Foundation (OBF)** Remote
May 2022 - Sep 2022
Student Developer (Full-time)
 - Engineered a robust user authentication and role-based authorization system using **Apollo GraphQL** mutations.
 - Ensured data security with **bcrypt** to hash and salt passwords for protection of user credentials stored in database.
 - Facilitated rigorous testing of all mutations through the use of **Jest**, ensuring high-quality, fault-free code.
 - Utilized **Redis** database for server-side session data storage, enabling efficient tracking of user cookies.
 - Implemented One-To-Many relationship between users and policy data in **MongoDB**, enhancing data organization.
 - Automated the backend deployment process using **Docker** and **GitHub Actions** onto a **DigitalOcean linux server**, significantly reducing manual input and increasing deployment efficiency.
 - Used **Docusaurus** to create a comprehensive documentation website for the project, enabling easy access to information for developers and users.
 - Integrated the backend with **Gmail API** to send verification emails to users, enhancing the system's verification process.
- **Accord Project** Remote
Feb 2021 - May 2021
Open-source Contributor (Part-time)
 - **Markdown Editor Enhancement:** Successfully integrated a WYSIWYG (What You See Is What You Get) interface into the SlateJS markdown editor. This included the **addition of markdown shortcuts** for all headings (1-6), code block, page break, and inline code components, achieved through the use of SlateJS plugins.
 - **UI/UX Improvements:** Redesigned and applied consistent styling to all heading elements (1-6) and the styling dropdown menu of the markdown editor, ensuring a visually appealing and intuitive interface.
 - **Bug Resolution:** Identified and resolved numerous bugs, enhancing the overall user experience for end users and simplifying the process for developers to introduce new features.

PROJECTS

- **AlgoViz - Android Application (Java, OOP, Material Design, Android Canvas, A*, Dijkstra):** AlgoViz is an android application that I developed to showcase my comprehension of multiple pathfinding algorithms through their visualization in a grid-based interface.
 - **Versatility in Search Modes:** Enables both diagonal and non-diagonal search executions on the grid.
 - **Algorithm Speed Control:** Incorporated a feature to control the operational speed of algorithms in real-time.
 - **Step-by-Step Execution:** Included a step function to allow users to progress through the algorithm's path incrementally for in-depth understanding.
- **Portfolio Application - Next.js App (Next.js [React Framework], UI/UX, Framer Motion):** Developed a comprehensive portfolio application utilizing **Next.js** and incorporating **Framer Motion** for seamless animations. This platform highlights my diverse skill set and accomplished projects, while demonstrating my proficiency in crafting an engaging user experience and showcasing my keen understanding of web design principles.
- **Reddit Clone - FullStack Next.js App (TypeScript, Next.js, URQL, Express, GraphQL, Redis, PostgreSQL, Docker, Linux):** Engineered a comprehensive full-stack application, mirroring Reddit's functionality and design. This robust application was built employing a variety of modern technologies:
 - **Frontend Development:** Leveraged TypeScript React, NextJs, and URQL to ensure a responsive, user-friendly interface.
 - **Backend Development:** Utilized Express, GraphQL, and PostgreSQL, all orchestrated by Dokku on the server, to manage data processes and operations efficiently.
 - **Server Management:** The server, powered by NodeJS, utilized GraphQL for optimized data manipulation and querying, backed by PostgreSQL as the primary database system. This setup ensured robust and efficient server operations.
- **Song-Viz - NPM Package (React, ViteJS, TypeScript, WebAudio API, HTML Canvas):** Song-Viz is a versatile React component, developed using **TypeScript** and packaged with **ViteJS**.
 - Utilizes WebAudio API for detailed audio analysis and HTML Canvas for audio visualization.
 - The package offers an extensive range of input props, enabling users to customize the application to their individual requirements.
 - First music visualizer on NPM built with functional React Components.
- **Blog Tutorials:**
 - How to make the smooth scrolling effect (Momentum Scrolling) in React?
 - How to make custom smooth scrolling indicators in React?
 - Implementing Page Transitions in Next.js Using Framer-Motion