

# KRISHNA KHARE

Kolkata, West Bengal | +91 9836609626 | krishnakhare2003@gmail.com  
<https://www.linkedin.com/in/krishna-khare-819422237/> | <https://krishna-khare.github.io/Portfolio/>

## SUMMARY

Results-Driven Web Developer: Skilled in designing and implementing dynamic, user-friendly websites and interfaces with a focus on efficiency and functionality.

Proficient in Programming: Experienced in diverse programming languages and frameworks, enabling the delivery of robust and scalable solutions.

Creative Problem Solver: Adept at developing innovative web designs and layouts that align with modern trends and user expectations.

Fast Learner: Enthusiastic about embracing new technologies and tools to stay ahead in the rapidly evolving tech landscape.

Team-Oriented and Dedicated: Committed to contributing to a collaborative work environment, while consistently meeting deadlines and achieving project goals.

Career Objective: Seeking an entry-level role in a fast-paced organization to apply technical expertise and passion for web development while growing professionally.

**TECHNICAL SKILLS:** HTML5, CSS, JavaScript, Bootstrap, React.js, Node.js, PHP, MySQL, .Net, Java

## PROFESSIONAL EXPERIENCE

CodSoft | Web Development Internship Remote

11/2023 - 12/2023

Tech Elecon pvt.ltd | Front-End Developer Intern Anand, Gujarat

September 2024 - Present

## EDUCATION & OTHER

**B. Tech in Computer Science & Engineering - 2021 to Current**

Dr. Sudhir Chandra Sur Institute of Technology Kolkata, West Bengal

**INTERESTS:** Problem solving, Music, Hiking, Reading

## PROJECTS

### ● Book Search Application

Purpose: A web-based application that allows users to search for books.

Search Functionality: Users can input a search query to search for books by title, author, or keyword.

API Integration: The app fetches book data from a public API (like Google Books API) to display results.

Result Display: Books are displayed with details such as title, author, and book cover.

### ● Full-Stack Blog Management Application

Developed a React-based frontend for creating, viewing, and managing blog posts with an intuitive and responsive user interface.

Designed a RESTful API backend using ASP.NET Web API, deployed to Azure App Service, enabling CRUD operations for blogs.

Integrated the frontend and backend seamlessly with proper routing and API calls.

Deployed the React app on Azure App Service for live demonstration.

Incorporated modern web development practices, such as state management and component-based architecture, to ensure scalability and maintainability.

- **Netix homepage clone**

Purpose: A clone of Netix's homepage to mimic its UI and layout.

Design: Replicates the visual design of Netflix's homepage, including the iconic dark theme.

Responsive Layout: The layout adjusts smoothly across different screen sizes and devices.

Content Sections: Features multiple content rows showcasing movie and TV show thumbnails.

Hover Effects: Thumbnails have hover effects similar to Netflix's interaction.

- **Weather Application**

Purpose: A web-based application that provides real-time weather information.

Search Functionality: Users can search for current weather conditions by entering a city name.

API Integration: Utilizes a weather API to fetch real-time data.

- **React News Portal with Movie API Integration**

Developed React News Portal: Designed and developed a user-friendly signup and login system using React.

Secure Authentication: Integrated secure authentication to manage user access efficiently.

API Integration: Implemented API integration to fetch and display the latest news post-login.

Responsive UI: Built a responsive UI for displaying news articles, ensuring compatibility across devices.

Efficient State Management: Emphasized efficient state management for a seamless user experience.

Custom Movie API: Developed and integrated a custom ASP.NET Web API to manage and display movie data, including titles, directors, release years, and poster URLs.

Continuous Verification: Ensured continuous authentication and authorization for secure access to movie data.

Real-Time Data Fetching: Utilized Axios in React for asynchronous API requests, ensuring real-time data fetching and dynamic content display.