

# Pradeep Gaur

✉ ps652227@gmail.com | 🌐 pradeep652227 | 🌐 pradeep-gaur-0384331a1 | ✖ pradeep\_gaur1

## Summary

A Computer Science Geek having experience in developing full-stack applications with right attitude toward solving problems in every step (learning, implementation, and debugging).

## Technical Skills

- >**Frontend**:- Javascript, React JS
- >**Backend**:- Node JS, Express
- >**Database**:- MongoDB, NoSQL, SQL

## Programming

- > C
- >C++

## Additional Skills | Expertise

- >OOPS, Data Structures and Algorithms, Git
- >SEO (Technical)
- >Content Writing

## Courses

- > C++ from Beginner to Beyond
- >Complete Web Development Bootcamp
- >Chai Aur React

## Education

B.Tech Computer Science and Engineering   BTU, Bikaner	08/2019 – 06/2023	8.9 CGPA
12 <sup>th</sup> Class   Oxford International Academy, Jaipur	2018-2019	68%
10 <sup>th</sup> Class   St.Thomas Sr.Sec. School, Jaipur	2016-17	9.00 CGPA

## Languages

- > English | Hindi

## Projects

### ->Full Stack Blog Application

An Application that allows a user to create a blog-post, edit it, and delete it with Authentication (signup/login) and separate posts (public/private).

Key technologies utilized include React Redux and React Router for front-end development, along with Express for server creation and handling GET and POST routes.

File Handling through Multer | CRUD operations by using the concepts of Schemas, Collections of MongoDB | Used Hashing + Salting to Encrypt Passwords.

### ->Authentication and Security

A web app created by using Node JS, Express, and MongoDB that allows the user to register and login with the registered details using passport, mongoose, and OAuth keeping in mind the security of the user details.

### ->Keeper App React

A React app built to keep the notes of the user using the basic and most used functionalities- hooks.

### ->To-Do App Node JS

To-Do application built by using Node JS, Express, and MongoDB. Separate routes for specific goals, storing the items of each route in the database, handling all the GET and POST routes, and retrieving the items on loading.