

# Varun R Mallya

B.Tech sophomore at IIT Roorkee | Developer at SDSLabs

JP Nagar  
Bangalore, Karnataka-560078  
[varunrmallya@gmail.com](mailto:varunrmallya@gmail.com)  
[varun-r-mallya.github.io](https://varun-r-mallya.github.io)  
[Linkedin](#)  
[Github](#)

## EXPERIENCE

### SDSLabs, IIT Roorkee — *Developer*

Feb 2024 - PRESENT

We work with various languages & frameworks including C++, OpenGL, SDL, React.JS, Golang, Rust. We build projects that are open source and participate in various CTFs and hackathons.

## EDUCATION

### IIT Roorkee, Uttarakhand — *B.Tech Mechanical Engineering*

August 2023 - August 2027

Semester 1 SGPA: **9.3**

Semester 2 SGPA: **10**

Current CGPA: **9.636** (Current department rank 1)

### Deeksha CFL PU College, Bangalore — *IInd PUC*

January 2020 - February 2022

PCMCs, graduating with 97.33% with 100s in Physics, Chemistry, Mathematics and 99 in Computer Science. State rank 13

JEE Mains AIR 4193 (99.63 percentile)

KCET Rank 31

JEE Advanced AIR 3656

## ACHIEVEMENTS

### ETHOnline 2024 — *Won the DIMO and Sign Protocol Tracks* <https://ethglobal.com/showcase/zerocabs-j27ig>

Wrote a completely decentralized taxi service using sponsor tracks with Web3 technologies and learnt a lot about blockchain in general. Made with a team of 5.

## SKILLS

Software Engineering

C++ Object Oriented  
Programming

Golang Server Programming

JavaScript with React.js

Docker

Linux

Python for AI and automation

Git

Makefile

x86-Assembly

Figma

## Other notable points

Competitive Programming  
Online Judge(s): [Codeforces](#)

Member of the Debating  
Society IIT Roorkee:

Participated in various  
international level debating  
tournaments.

WatchOut IIT Roorkee:

Wrote editorials, scraped data  
and managed editorial teams  
to get news for the whole  
campus.

## PROJECTS

## Particle Simulation — *Particle simulation written with SDL2 in C++*

[github.com/varun-r-mallya/ParticleSim](https://github.com/varun-r-mallya/ParticleSim)

Wrote collision mathematics and simulated and optimized a particle simulator with Object Oriented C++. As a side project to this, I also wrote a [damped bouncing ball simulator](#).

## Chromatica — *A 2D platformer game made in Godot*

<https://github.com/varun-r-mallya/WTFxIDGC>

Wrote recursive algorithms to implement a very peculiar game mechanic made for a game jam. Also wrote game logic in GDScript. Can be played [here](#)

## Zeus — *A project to learn how modern container runtimes work*

<https://github.com/sdslabs/Zeus>

Involves threading and concurrency concepts with Goroutines and involves Linux internals such as namespaces and cgroups. Made under SDS Labs.

## MentoGram — *A mentorship web app*

[github.com/varun-r-mallya/mentogram-final](https://github.com/varun-r-mallya/mentogram-final)

Written on the MERN stack, it makes use of WebRTC for real time video transmission, has an internally implemented file manager in NodeJS, basic CRUD login functions and chat implemented with Socket.io with frontend written in ReactJS

Also uses Docker for program compilation and y.js for collaborative programming.

## MVC-LMS — *A Library Management System written in Go*

[github.com/varun-r-mallya/MVC-LMS-SDS](https://github.com/varun-r-mallya/MVC-LMS-SDS)

Written in Go, it uses the Go html template engine to render pages with MariaDB/MySQL as the database. It is a full fledged Library manager with features including JSON Web Tokens, secure routes, password hashing. Is completely Dockerized with Docker Compose along with config files for Apache to virtually host on your computer.

## Review Reply Bot — *NLP based reply bot*

[github.com/varun-r-mallya/byop](https://github.com/varun-r-mallya/byop)

Uses TF-IDF to tokenize and vectorize datasets for reviews and sorts and replies to reviews using Latent Dirichlet Allocation for topic extraction. Also uses NLTK as an alternate means for tokenization.

## Plant Watering IOT system — *ESP8266 based IOT system*

[github.com/varun-r-mallya/Plant\\_watering\\_system](https://github.com/varun-r-mallya/Plant_watering_system)

Uses an ESP8266 to connect to the internet and uses [this](#) as the frontend to control pumps and get data from soil moisture and temperature sensors to a Google Sheets backend that also functions as a database for the frontend application. Uses Google App Script.

## LANGUAGES

English, Hindi, Konkani,  
Kannada, Sanskrit

## REFERENCES

[Ankit Bansal, Associate Professor, IIT Roorkee](#)

[Nikil Kumar Singh, Assistant Professor, IIT Roorkee](#)

[Tarun Gangwar, Assistant Professor, IIT Roorkee](#)