



MANISH UPRETI

[WEBSITE](#)

☎ +91 7011834071 ✉ mupreti2002@gmail.com [LinkedIn](#) [Manish85-cell](#) [Leetcode](#)

Education

Netaji Subhas University of Technology

B.Tech in Computer Science and Engineering

CGPA: 8/10

2022 - 2026

Dwarka, Delhi, India

Work Experience

Amdocs

March 2025 – April 2025

Received mentorship from the Software Engineering Directors of Amdocs Mr. Ravinder Pal Singh and Mr. Nilesh Somvanshi for GenAI Hackathon project.

AlgoUniversity

May 2024 – Present

Software Development Extern

Remote

- Worked on an undisclosed project as part of the externship.

HEAT-WAVE | An Online Judge

[Source Code](#)

- An online judge that checks for compilation errors, passes test cases, and evaluates code on styling, readability, DRY (Do not Repeat Yourself), and comments.
- Important for writing readable code with minimal repetition, essential for real-world projects and team collaboration.
- Used **Django** (Python Web Framework) and integrated **Generative AI** (Gemini API) for evaluating code after submission
- Used **Docker** and **AWS** ECR, S3 and EC2 instance for deployment
- [Public Link](#), [Video Explanation](#), [HLD/Design](#)

- Received mentorship from senior engineers from Google London, Apple, Bytedance Singapore, and Alphagrep Singapore.

- Learned real-world software development and scalability.

PROJECTS

Glass-Media : Misinformation Detector and Fact Checker

[Source Code](#)

- Developed as part of Amdocs GenAI Hackathon 2025
- **AI** based application that flags news as real or fake and performs fact checking.
- Fine-tuned **BERT-base model** on a labelled dataset of fake/real news.
- Integrated Gemini API to classify user input and verify facts.
- Added **image-to-text** and **Hindi-to-English** translation for usability and regional support.
- [Public Link](#), [Video Explanation](#), [Presentation](#)

Sorting Simulator

[Source Code](#)

- Developed for visualizing how a sorting algorithm works good with some kind of data but not works with other.
- For example, Selection Sort performs good on data where the sorting key is small but the associated values are large, making swaps expensive in terms of time and space.
- As Selection Sort has the minimum number of swaps among comparison-based algorithms, it is more efficient in such scenarios.
- A **Python based application** built using Pygame and Tkinter library
- With the ability to **customize input data** to observe which algorithm performs better on different types of data.
- [Download App](#), [Video Explanation](#)

TECHNICAL SKILLS

- **Languages:** C, C++, Java, Python; **Frontend:** HTML, CSS, JavaScript; **Frameworks:** Django, Flask;
- **Databases:** MySQL, SQLite
- **Other Technologies:** Git, GitHub, Vercel, Netlify, Docker, AWS, NLP, LLM(BERT), Generative AI;
- **CS Fundamentals:** Object-oriented design, Data structures, Algorithm design and problem solving, Operating System, Networks

HONOURS and Awards

Amdocs GenAI Hackathon

2025

Among Top 8 Finalist out of 11000+ participants

National Science Olympiad / Techfest (IIT Bombay)

2018

Gold Medalist