

UDDIT

📍 Jaipur, Rajasthan ✉ 2021umt1791@mnit.ac.in ☎ 745 688 68 77

🔗 <https://udditwork.github.io/PORTFOLIO-Uddit/> in lorduddit- 🌐 UDDITwork

Summary

Software engineer 🔗 with experience in full-stack development and AI/ML. Worked on scalable applications for 2 MIIC Incubator startups, contributed to LLMs' patent prosecution at INVENT IP and specializing in transformer models like OpenAI's GPT and Claude AI. Experienced in integrating AI/ML into software applications to create real world impact among users.

Education

Malaviya National Institute of Technology Jaipur <i>Metallurgical and Materials Engineering</i>	2021-2025
◦ CGPA: 7.1 (work 🔗)	
Girish Prasad Memorial College, Bareilly	2016-2017
◦ ICSE X :96%	
Girish Prasad Memorial College, Bareilly	2018-2019
◦ ISC XII :94%	

Experience

Application Developer <i>Zooner and The Vintage Foods, MIIC Incubator</i>	Jaipur, India 2022-2023
◦ Developed websites and mobile apps for 2 MIIC-funded startups using React.js, Vite, Next.js, Node.js, MongoDB for websites and Flutter for apps.	
Machine Learning Intern <i>Electroglobal Inc, Bhamashah Techno Hub</i>	Jaipur, India October-Dec 2024
◦ Developed an AI-powered chatbot with React.js front-end and Flask back-end for IoT projects' progress tracking in real time.	
◦ Integrated OpenAI API for NLP-based guidance and Firebase for user data storage.	
Associate Patent Trainee, USPTO <i>InventIP Legal</i>	Noida, UP January 2025-Current
◦ Analyzed Transformer-based LLM architectures for patent applications.	
◦ Developed an offline server-based implementation of the DeepSeek-R1 1.5B model to ensure privacy in patent claim drafting.	
◦ The model was trained for precise and context-aware outputs, optimizing it for legal and technical document generation.	


Achievements

Live Coding Contest by Qurious Bit Games	1 October 2024
Internship + PPO offer by AAIS Global	20 January 2025
github.com/UDDITwork 🔗	
Injective AI Agent Hackathon (dorahacks.io) 🔗	8 February 2025


Projects

1. Handwriting Character Recognition and Image Text Extraction Tool	
◦ github.com/UDDITwork/handwriting-character-Recognition 🔗	
◦ Developed a Handwriting-to-Text Conversion OCR WebApp using Deep Learning, Tesseract OCR, OpenCV, and Flask, enabling multilingual handwritten character recognition from images with real-time processing.	
◦ Tools Used: Deep Learning, Tesseract OCR, OpenCV, and Flask	


2.Resume Maker Applicaion

- github.com/UDDITwork/Career-Resume-Builder 
- Developed with Next.js 13, React, and Redux Toolkit, enabling resume import, parsing, and PDF generation. Express.js handles API requests, while spaCy and pdfplumber power NLP-based parsing. Jinja2/LaTeX generate structured resumes.
- Tools Used : TypeScript, Tailwind CSS, Next.js 13, PDF.js for parsing, and React-pdf for rendering and exporting resumes.


3.Next.js Ai Chatbot

- <https://nextjs-ai-chatbot-gules-five.vercel.app/> 
- Developed using Next.js and Vercel's AI SDK, leveraging the streamText function on the server for real-time responses and the useChat hook on the client for a seamless chat experience.
- Tools Used: Next.js, Vercel AI SDK, React, TypeScript, Tailwind CSS, and Node.js for efficient backend handling and API integration.


4.Shree Ai Document Analyzer

- <https://zupe.app/shreeai-document-analyzer> 
- Built using NLP and OCR technologies, where pdfplumber, PyMuPDF, and python-docx extract text from PDFs and DOCX files. spaCy, Hugging Face Transformers, and LangChain process and understand the extracted content for context-aware analysis.
- Tools Used: LLMs like OpenAI GPT or Llama can be used to generate intelligent responses based on document data. FastAPI or Express.js handles API requests, while Vector Databases (FAISS, Pinecone) enable efficient semantic search and retrieval-based question answering.

5.DSA Visualizer Tool

- <https://udditwork.github.io/DsAvisualizer/> 
- An interactive visualization tool used to understand the concept of addition or deletion of nodes and data in real time , for effective understanding of Binary Trees, Queues,Stack ,Linked List,Hash Map, Tree and other data structures.
- Tools used: Canvas, WebGL for animations, Three.js, React for front-end and Node.js for back-end.

6.Multi-Face Recognition Attendance System

- <https://zupe.app/trial-phase-face-recognition> 
- Tech Stack: Use Python with OpenCV for real-time image capture, DeepFace for facial recognition, Deep Learning (CNNs) for improved accuracy, and PostgreSQL to store student data and attendance records.
- Implementation: Capture a group image using OpenCV, detect and extract student faces with DeepFace, perform face pose augmentation (saved in PostgreSQL as embeddings), match faces against the database, and mark attendance automatically.

Technologies

Languages: C++,C,HTML5,CSS3,Javascript,Python,SQL

Backend: Node.js,Express.js, NestJS, Flask ,Gunicorn(for production server),Axios/Fetch API-for making API calls,REST for APIs and WebSockets for real-time communication.

Machine Learning and open source LLMs: Llama,Mistral and DeepSeek.

Deep Learning Frameworks: PyTorch and TensorFlow.

Frontend: Javascript libraries:JQuery and Alpine.js,CSS Frameworks: Bootstrap and Tailwind CSS.

Databases and Caching: PostgreSQL,MySQL,MongoDB,FireBase and Redis.

Media and Publications:

Research Gate:  **Youtube:**  **Articles:** 