

VISHAL SINGHANIA

📞 +91 7782874447 | ✉ vishsinghania@outlook.com | 💻 [blog](#) | [in /singhaniav](#) | [🌐 /SinghaniaV](#) | [</> /bigV_](#)

EDUCATION

Manipal University Jaipur

Aug 2020 – July 2024 | Jaipur, Rajasthan

BTech in Computer & Communication Engineering | Minor in Data Science

Cum. GPA: 8.82 / 10.0

Relevant Coursework: Design & Analysis of Algorithms, Data Structures & Algorithms, Object Oriented Programming, Operating Systems, Database Systems, Computer Networks, Data Mining & Warehousing, Cloud Computing, Artificial Intelligence & Deep Learning, Big Data Analytics, Probability & Statistics, Discrete Mathematics.

Open Courseware: Graph Algorithms by **UC San Diego**, Programming in Python by **Python Institute**, CCNav7: Intro to Networks by **Cisco**, C++ Specialization by **UIUC**, System Administration by **Red Hat**.

TECHNICAL SKILLS

Programming & Scripting: C, C++, Python, Bash, SQL, R, MATLAB, Java, Golang

Backend & Databases: Django, Flask, FastAPI, PostgreSQL, MongoDB, Redis

Tools: Docker, Git, Postman, Kafka, RabbitMQ, Google Cloud Platform

Data Science & Machine Learning: SciPy, Pandas, NumPy, NLTK, Matplotlib, scikit-learn, TensorFlow

PROJECTS

Conduit 🌐 | Python, FastAPI, MongoDB, Docker, GitHub Actions

Backend + Databases

Backend logic implementation of a dynamic social blogging platform like **Medium.com**, influenced by the [RealWorld](#) GitHub initiative.

- **API Specifications and Features:** The system offers robust user authentication using **JSON Web Tokens (JWT)** for secure access. Users can manage their profiles, create, retrieve, update, and delete articles, and engage with content through commenting and favoriting features. A follow/followers system fosters community connections, while pagination enhances browsing efficiency.
- **Testing Framework:** The testing framework comprises unit testing, which scrutinizes individual components with **mock dependencies**, and integration testing, which verifies smooth interactions between components. Automated API tests ensure the validity of responses, status codes, and data integrity.
- **Continuous Integration (CI):** Integrated API testing in **GitHub Actions** for thorough validation.

WatsonX 🌐 | Python, NLTK

Machine Learning + Natural Language Processing

Built a question answering system similar to [IBM Watson](#). It operates on a corpus of text documents and aims to find the most relevant documents and passages to a given query.

- For document retrieval, the system uses **tf-idf (term frequency-inverse document frequency)** to rank them based on the frequency of query terms and their overall importance in the corpus.
- Passage retrieval is performed by subdividing the top document(s) into sentences. In scoring the passages, the system employs a combination of **inverse document frequency** and a **query term density** measure.

ALGORITHMIC COMPETITIONS/ACHIEVEMENTS

- **3 star (Div. 2)** | Peak rating of **1630** on CodeChef | **global rank of 293** on Starters 75 | [CodeChef/big_v](#).
- **Contributor** on Kaggle (a data science competition platform) | [Kaggle/bigvish](#).
- **Qualified Round 1** | ACM Semi Code (an Institute level **Code-a-thon**).
- Contribution recognized in **Hacktoberfest 2021**.
- Contributed to **Wikipedia** articles through an Institute level **Edit-a-thon**.