HEMANT GHARAT

Panvel, Maharashtra, India 410206

📞 887-906-9414   ✉ Email   in LinkedIn   Github

Profile Summary

Proficient Python Developer skilled in building scalable applications, APIs, and data-driven solutions. Experienced in Python frameworks such as Django, Flask, and FastAPI, with knowledge of libraries like NumPy, Pandas, and Scikit-Learn. Strong in database management with MySQL and MongoDB. Well-versed in AWS, Docker, and CI/CD pipelines using Jenkins and Argo CD. Currently learning machine learning and focused on efficient, quality-driven development.

Education

University of Mumbai   **2015**
Bachelor of Science in Information Technology   *Mumbai, Maharashtra*

Experience

Flow Learning Solutions (Tailwnd)   **October 2022 – Present**
Software Developer   *Navi Mumbai, Maharashtra*

- Developed a machine learning application that incorporates advanced features such as speech recognition, real-time feedback and assisting user in improving their reading skills.
- Developed a Python backend application to implement various functionalities, integrating MongoDB database interaction and machine learning (ML) model execution.
- Developed a RESTful API using FastAPI, a modern Python web framework, to provide efficient communication between frontend and backend components of an application.
- Configured various AWS services such as Amazon EC2, Amazon S3, AWS IAM, Amazon SQS, Amazon ECR, Amazon Route 53, Amazon Lambda etc.
- Worked on Automatic Speech Recognition (ASR) model training using the Kaldi toolkit.
- Contributed as a team member to the development of Python scripts for various tasks, leveraging collective expertise to achieve project goals efficiently.
- Managed and deployed microservices using Kubernetes, ensuring high availability, scalability, and efficient orchestration of containerized applications.
- Built and optimized Docker containers for application deployment, improving the development process and enabling seamless integration between environments.
- Automated CI/CD pipelines with Jenkins and Argo CD, streamlining the build, test, and deployment processes for various applications.
- Integrated SonarQube for continuous code quality checks, ensuring compliance with coding standards and improving code maintainability.

Reliance Industries Limited   **August 2019 – October 2022**
Document Administrator and Support (EDMS)   *Navi Mumbai, Maharashtra*

- "J3 (Jamnagar Refinery Expansion (Phase -III)) Project", "REG (Reliance Engineering Group.) Project" and "EPCM Project" - These applications are designed to maintain the drawings documents.
- Products used, implemented, and supported: REIMS (Reliance Enterprise Information Management System), REIMS eRoom, IE Utilities and Repoint.
- Creating custom queries using DDL, DQL, and DML commands, Creating users, groups /roles for the distribution, and Creating ACL.
- Troubleshooting user issues and communicating with user and management.
- REIMS and eRoom: Administration, maintenance, ID Creation, User Training, Room training for vendors, report generation, handling queries.

Skills

Languages: Python, SQL, HTML/CSS, JavaScript
Machine Learning and Data Analysis Libraries: Scikit-Learn, NumPy, Pandas, Matplotlib, Seaborn
Web Frameworks: Django, Flask, FastAPI
Databases: MySQL, PostgreSQL, MongoDB, SQLite
Developer Tools: VS Code, Jupyter Notebook, Git, Docker, AWS (EC2, S3)
Data Visualization Tools: Power BI, Tableau
Other Technologies/Frameworks: Linux, Bash, Jenkins, GitHub, Bitbucket, SonarQube

Projects

E-commerce | *Django Project* (GitHub)
- Developed a fully functional e-commerce platform using Django, providing seamless user experiences for browsing, purchasing, and order tracking.
- Implemented secure user authentication, product search, and a cart system.
- Designed a responsive frontend using Bootstrap and managed backend operations with Django ORM and SQLite3.

Todo Application | *Django Project* (GitHub)
- A simple Django-based To-Do application that allows users to add, update, and delete tasks with completion status.
- Implemented task CRUD operations with Django ORM and SQLite3.
- Integrated Bootstrap for a responsive and user-friendly interface.
- Added task creation timestamps and task completion tracking for better organization.

Blogger App | *Django Project* (GitHub)
- Built a full-featured blogging platform using Django, allowing users to create, edit, and delete blog posts.
- Integrated a rich-text editor and comment system for interactive engagement.
- Designed a responsive UI with Bootstrap and managed data with Django ORM.

Expense Tracker - REST API | *Django REST Framework* (GitHub)
- Developed a RESTful API for tracking expenses and categorizing them using Django REST Framework (DRF).
- Implemented user authentication, expense CRUD operations, and analytics endpoints.
- Integrated JWT authentication for secure API access.

E-commerce | *Flask Project* (GitHub)
- Designed and implemented an e-commerce website using Flask, MySQL, and Bootstrap.
- Built features like product search, shopping cart, and order management.

Devanagari Transliteration API | *FastAPI Project* (GitHub)
- Developed a FastAPI service for converting ITRANS format text to Devanagari script.
- Implemented fuzzy matching to suggest similar Hindi words from a predefined list.
- Added CORS support for cross-origin requests, ensuring smooth integration with web applications.

Salary Prediction | *Simple Linear Regression* | *Machine Learning* (GitHub)
- Implemented a salary prediction model using Simple Linear Regression to estimate salaries based on years of experience.
- Preprocessed data using Pandas and visualized relationships using Matplotlib.
- Trained and tested the model using Scikit-Learn, achieving accurate salary predictions.
- Created regression plots to visualize training and test results effectively.

Email Classification | *Logistic Regression, TF-IDF & Flask* | *Machine Learning* (GitHub)
- Implemented an email classification system using Logistic Regression and TF-IDF to identify spam and ham emails.
- Developed a Flask API for real-time classification of emails using the trained model.
- Saved and loaded the trained model and vectorizer for efficient predictions.
- Utilized a dataset of labeled emails to train the classification model and evaluated its accuracy.