Madhuben & Bhanubhai Patel Institute of Technology

(A Constituent College of CVM University)

New V. V. Nagar

COMPUTER ENGINEERING DEPARTMENT

Mini Project Proposal

on

CareerIQ

Submitted By

Name of Student : Mohamad Nadeem Memon

Enrolment Number :12402040703013

Name of Student :Rudra Trivedi

Enrolment Number :12302040701148

MINI PROJECT (202040601)

A.Y. 2025-26 EVEN TERM

INTRODUCTION

Problem Statement

The job market for Data Science, Artificial Intelligence, Machine Learning, and Cloud roles is rapidly evolving. Job seekers often struggle to understand which skills are in demand, what roles are trending, and how experience and location affect job availability. There is a lack of a centralized, interactive system that analyzes real job market data and presents insights in an easy-to-understand manner.

Project Summary And Introduction

This project proposes a Job Market Intelligence Web Application that addresses the challenges faced by job seekers in the rapidly evolving technology sector. The system collects and processes real job market datasets, performs Exploratory Data Analysis (EDA), and provides interactive visualizations and filters. It helps users explore job roles, skills, locations, and experience levels through an intuitive interface. The system is built using Python, Pandas, and Streamlit, making it lightweight, interactive, and user-friendly. By transforming raw job market data into meaningful insights, this application bridges the gap between data analysis and actionable intelligence for students, job seekers, and educators.

Aim And Objective Of Project

The primary aim of this project is to develop an intelligent job market analysis system that empowers users with data-driven insights about career opportunities in Data Science, AI, ML, and Cloud domains. The specific objectives are:

To analyze real-world job market datasets collected from multiple sources and extract meaningful patterns and trends. To unify multiple datasets into a single structured format ensuring consistency and reliability of information. To extract meaningful insights using Exploratory Data Analysis techniques including statistical analysis and visualization. To build an interactive dashboard for users that provides seamless navigation and exploration of job market data. To help users explore job trends using filters and search options based on role category, location, experience level, and job titles. To provide clear insights into skill demand and job availability patterns across different regions and experience levels. To create a practical tool that serves as a learning resource for students while being valuable for actual job seekers.

System Analyses

Motivation

The motivation behind this project stems from the growing complexity of the job market in technology sectors. Students and professionals often find it challenging to identify which skills to develop, which locations offer the best opportunities, and how their experience level aligns with market demands. Traditional job portals provide listings but lack comprehensive analytical insights that can guide career decisions. This project aims to fill that gap by providing a data-driven approach to understanding job market trends. By analyzing thousands of job listings, the system reveals patterns that are not immediately visible to individual job seekers. The insights generated can help users make informed decisions about skill development, location preferences, and career planning. Furthermore, this project provides hands-on experience with real-world data analysis, web development, and deployment, making it an excellent learning opportunity for students entering the field of data science and analytics.

Brief Literature Survey

The field of job market analysis using data science techniques has gained significant attention in recent years. Various studies have explored the use of web scraping, natural language processing, and machine learning to extract insights from job postings. Research has shown that analyzing job descriptions can reveal skill demand patterns, salary trends, and geographic preferences in the job market. Several platforms have attempted to provide job market intelligence, but most focus on specific regions or job types. Academic research has demonstrated the effectiveness of exploratory data analysis in identifying trends and patterns in employment data. Studies have also highlighted the importance of data standardization when working with multiple job listing sources, as each platform uses different formats and terminologies. The integration of interactive visualization tools like Streamlit has been recognized as an effective approach for making data insights accessible to non-technical users. Previous work in this domain has primarily focused on either data collection or visualization, but rarely combines both aspects into a unified, user-friendly system. This project builds upon existing research by creating a comprehensive solution that addresses both technical and user experience aspects of job market analysis.

Design: Analysis, Design Methodology And Implementation Strategy

H/W And S/W Requirement

Hardware Requirements:

The system requires minimal hardware resources, making it accessible for development and deployment. A standard personal computer or laptop with at least 4GB RAM and 2GB free disk space is sufficient for development. For deployment, a basic cloud server or local machine capable of running Python applications is required. The system is designed to be lightweight and does not require specialized hardware or GPU acceleration.

Software Requirements:

Programming Language: Python 3.8 or higher is used as the primary development language due to its extensive library support for data analysis and web development. Libraries: Pandas for data manipulation and preprocessing, NumPy for numerical computations, Matplotlib and Seaborn for data visualization and creating insightful charts and graphs. Web Framework: Streamlit is used to build the interactive web application, providing a simple yet powerful interface for data exploration. IDE: VS Code serves as the integrated development environment, offering excellent support for Python development with extensions for debugging and version control. Data Source: Kaggle provides publicly available job market datasets from various sources including Naukri job listings and global data science salary datasets. Version Control: Git is optionally used for tracking changes and collaboration between team members. Operating System: The application is platform-independent and can run on Windows, macOS, or Linux systems. Browser: Any modern web browser (Chrome, Firefox, Safari, Edge) is required to access the Streamlit dashboard.

Implementation

System Flow

The system follows a structured flow that ensures data quality and user-friendly interaction. The process begins with data collection where raw job datasets are obtained from Kaggle, including Naukri job listings focused on India-based positions and global Data Science salary and job datasets. These datasets are then processed through a data preprocessing and cleaning pipeline that handles missing values, removes duplicates, and standardizes data formats across different sources. Feature standardization is performed to ensure all datasets share common attributes such as job title, company name, location, experience level, and role category. The cleaned and standardized data is merged into a single master file called jobs_master.csv containing 13,623 job records. Exploratory Data Analysis is then conducted to identify patterns, trends, and distributions in the data. The processed data is fed into an interactive Streamlit dashboard that serves as the user interface. Users interact with the system through various input methods including a search bar for job titles, dropdown filters for role category selection, location filtering, and experience level filtering. The system responds to user inputs by dynamically updating visualizations and displaying filtered results in tables and charts. This interactive flow allows users to explore the job market from multiple perspectives and gain insights relevant to their specific interests.

Program/Module Specification

The system is organized into several modular components for maintainability and scalability. The Data Collection Module handles downloading and organizing raw datasets from Kaggle sources. The Data Preprocessing Module performs cleaning operations including handling missing values using appropriate imputation techniques, removing duplicate entries, standardizing column names and formats, and normalizing categorical variables like experience levels and locations. The Data Integration Module merges multiple datasets ensuring consistency of schema, resolving conflicts in overlapping data, and creating a unified master dataset with all required features. The Exploratory Data Analysis Module generates statistical summaries, creates distribution plots, identifies correlations between variables, and produces insights about skill demand and location trends. The Streamlit Dashboard Module forms the core user interface with a home page providing an overview and key statistics, search functionality allowing users to find specific job titles, filter components for role category, location, and experience level, visualization components displaying interactive charts and graphs, and a results display showing filtered job listings in tabular format. The Data Visualization Module creates various chart types including bar charts for role distribution, pie charts for location analysis, line charts for experience level trends, and heatmaps for skill demand patterns. Each module is designed to be independent yet interconnected, allowing for easy testing, debugging, and future enhancements. The modular architecture also facilitates the addition of new features such as machine learning models for salary prediction or recommendation systems without requiring major restructuring of existing code.

Time Line Chart

Month 1 - Project Planning and Data Collection: Week 1-2: Requirement analysis, literature review, and technology selection. Week 3-4: Dataset identification, collection from Kaggle, and initial data exploration.

Month 2 - Data Preprocessing and Integration: Week 1-2: Data cleaning, handling missing values, and removing duplicates. Week 3-4: Feature standardization, data integration, and creation of master dataset.

Month 3 - Exploratory Data Analysis: Week 1-2: Statistical analysis, trend identification, and pattern recognition. Week 3-4: Visualization creation, insight extraction, and documentation.

Month 4 - Dashboard Development: Week 1-2: Streamlit interface design, implementation of search and filter features. Week 3-4: Integration of visualizations, user testing, and refinement.

Month 5 - Testing and Deployment: Week 1-2: System testing, bug fixes, and performance optimization. Week 3-4: Documentation completion, final presentation preparation, and project submission.

Expected Outcome

The project is expected to deliver a fully functional job market analysis dashboard that provides clear insights into job trends and skill demand patterns across Data Science, AI, ML, and Cloud domains. Users will have access to an easy-to-use tool that allows them to search for specific job titles and filter results based on role category, location, and experience level. The system will present data through interactive charts and tables that make complex job market information accessible and understandable. Students and job seekers will gain practical exposure to real-world data analysis and deployment processes. The application will serve as a valuable resource for career planning and skill development decisions. The dashboard will feature a clean and intuitive user interface that requires no technical expertise to navigate. By the end of the project, users will be able to identify which skills are most in demand, understand geographic variations in job availability, and recognize how experience levels affect career opportunities in their chosen field.

Future Enhancements

The current system provides a strong foundation for further development and advanced features. Future enhancements could include salary prediction using machine learning models that analyze factors such as skills, experience, location, and company size to estimate expected compensation ranges. A skill recommendation system could be developed to suggest which skills users should learn based on their current profile and desired job roles. Time-based job trend forecasting using historical data analysis and predictive modeling could help users anticipate future market demands. Deployment on Streamlit Cloud or other cloud platforms would make the application accessible to a wider audience without requiring local installation. User authentication and saved preferences would allow individuals to create accounts, save their favorite searches, and receive personalized job alerts. Natural language processing could be integrated to analyze job descriptions and extract detailed skill requirements automatically. A comparison feature could enable users to compare multiple job roles or locations side by side. Integration with real-time job APIs would ensure the data remains current and reflects the latest market conditions. A feedback mechanism could be implemented to collect user input and continuously improve the system's accuracy and relevance. Mobile responsiveness could be enhanced to provide a seamless experience across all devices. These enhancements would transform the current analytical tool into a comprehensive career guidance platform.

Conclusion

This project demonstrates how real-world job market data can be transformed into meaningful insights using data analysis and visualization techniques. By collecting and processing job listings from multiple sources, the system provides a comprehensive view of opportunities in Data Science, Artificial Intelligence, Machine Learning, and Cloud computing domains. The interactive dashboard bridges the gap between raw data and actionable intelligence, making complex job market information accessible to users regardless of their technical background. For students, this project offers valuable hands-on experience with data preprocessing, exploratory data analysis, web development using Streamlit, and working with real-world datasets. For job seekers, it provides data-driven insights that can guide career decisions, skill development priorities, and job search strategies. The modular architecture ensures the system can be easily maintained and extended with additional features in the future. Through this project, we have created a practical tool that addresses a real need in the job market while simultaneously serving as an excellent learning resource. The combination of technical skills, practical application, and user-focused design makes this project a valuable contribution to both educational and professional domains. As the job market continues to evolve, tools like this will become increasingly important in helping individuals navigate their career paths with confidence and clarity.