A shield with a mermaid and lion

AI-generated content may be incorrect.

**Rookies**

**Leveraging Sentiment Analysis in web application for career guidance and roadmap visualization.**

**Qin Tao Zeng  
Student ID: 2459811**

School of Computer Science

College of Engineering and Physical Sciences

University of Birmingham

2024-2025

Abstract

Acknowledgement

I would like to thank my supervisor, colleagues, and family for their support and guidance throughout this project. Special thanks go to the developers and communities behind Next.js, TailwindCSS, and the various tools used in this research.

Contents

Table of Contents

1. Introduction
   1. Background
   2. Objectives
   3. Significance
2. Literature Review
   1. Career Guidance Systems
   2. Sentiment Analysis in Decision Support
   3. Full-Stack Web Development for Educational Tools
3. System Architecture and Technologies
   1. Architecture Overview
   2. Technologies Used
   3. Data Flow
4. System Design and Implementation
   1. User Interface Design
   2. Database Design and Schema
   3. Implementation Details
   4. Human Centred Design
5. Sentiment Analysis and Recommendation
   1. Sentiment Analysis Approach
   2. Recommendation Algorithm
   3. Discussion
6. Evaluation and Testing
   1. Functional Requirements
   2. Non-Functional Requirements
7. Discussion and Future Work
8. Conclusion
9. Bibliography
10. Appendices

**Chapter 1: Introduction**

**1.1 Background**

First-year computer science students often face significant uncertainty when selecting a career path. Traditional career counseling may not fully address individual interests or leverage the potential of data-driven insights. *Rookies* is developed to bridge this gap by integrating modern web development techniques with advanced sentiment analysis to provide personalized career guidance and a structured roadmap.

**1.2 Objectives**

The main objectives of this project are:

* To develop a full-stack web application that collects user data through an interactive quiz.
* To design and implement a recommendation algorithm that integrates sentiment analysis for ranking career options.
* To visualize career roadmaps with detailed milestones, guiding users in a step-by-step manner.
* To ensure the application is user-friendly by applying Human-Computer Interaction (HCI) principles.

**1.3 Significance**

By providing personalized recommendations and visual roadmaps, *Rookies* aims to empower students to make informed career decisions. The project demonstrates the practical application of sentiment analysis and full-stack web development in the field of career guidance, addressing a critical need among early-stage computer science students.

**Chapter 2: Literature Review**

**2.1 Career Guidance Systems**

This section reviews existing career guidance systems, highlighting their approaches, strengths, and limitations. While many systems offer generic advice, few tailor recommendations to the specific skills and interests of computer science students.

**2.2 Sentiment Analysis in Decision Support**

Sentiment analysis has been successfully applied in domains such as product reviews and social media monitoring. This thesis explores its novel application in career guidance, where sentiment scores—derived from market data and user feedback—can enhance the personalization of recommendations.

**2.3 Full-Stack Web Development for Educational Tools**

A review of modern web development frameworks and tools (Next.js, React, TailwindCSS, etc.) reveals trends in building responsive, scalable applications for education. This review supports the choice of technologies used in *Rookies*.

**Chapter 3: System Architecture and Technologies**

**3.1 Architecture Overview**

*Rookies* follows a modern full-stack architecture:

* **Frontend:** Developed using Next.js (React) with TailwindCSS for styling.
* **Backend:** Serverless functions deployed on Vercel, with a Neon cloud database managed via Drizzle ORM.
* **Authentication:** Managed by Auth.js for secure user sessions.
* **Data Flow:** User inputs are collected through a quiz, processed by serverless functions, and stored in the database. The recommendation algorithm runs on the server and returns personalized career data.

**3.2 Technologies Used**

* **Next.js & React:** For building a dynamic and responsive UI.
* **TailwindCSS:** For rapid UI development and styling.
* **Neon & Drizzle ORM:** For cloud-based database management and ORM interactions.
* **Auth.js:** For handling user authentication.
* **Vercel:** For deployment and scaling of serverless functions.

**3.3 Data Flow**

User interactions initiate data flow:

1. Users complete the quiz.
2. Responses are saved locally (e.g., in localStorage) and submitted to the backend.
3. The recommendation API processes these responses, applies the sentiment analysis, and retrieves matching career data.
4. The selected career and its roadmap (milestones) are stored and displayed on the homepage.

**Chapter 4: System Design and Implementation**

**4.1 User Interface Design**

The application includes several key UI components:

* **Quiz Page:** An interactive quiz with Next.js and TailwindCSS, capturing responses and storing them locally.
* **Recommendation Page:** Displays a ranked list of career options based on the recommendation algorithm. Users can select a career here.
* **User Profile and Homepage:** Shows the user’s chosen career and its associated roadmap, with interactive elements (toggles, modals, or accordions) for milestone details.

**4.2 Database Design and Schema**

The system uses a relational database with the following tables:

* **users:** Stores user credentials and profile information.
* **careers:** Contains career details including title, description, and sentiment score.
* **user\_career:** Links users to their selected career.
* **milestones:** Details key milestones for each career.
* **career\_milestone:** Establishes a many-to-many relationship between careers and milestones, with an additional step\_order field to indicate the sequence.

*(Include an ER diagram if available.)*

**4.3 Implementation Details**

* **Server Actions & API Endpoints:**  
  API endpoints (e.g., /api/recommend, /api/select-career, /api/user-career-details) are implemented to handle quiz response processing, career selection, and retrieval of career details.
* **Frontend Components:**  
  Components are built with React and styled using TailwindCSS. The use of shadcn UI components ensures a polished, professional interface.

**4.4 Human Centred Design**

**Chapter 5: Sentiment Analysis and Recommendation**

**5.1 Sentiment Analysis Approach**

Sentiment analysis in this project evaluates career-related data from multiple sources (such as market trends and user feedback) to generate a sentiment score for each career. This score represents the perceived positivity or viability of a career path and is integrated into the recommendation algorithm.

**5.2 Recommendation Algorithm**

The algorithm, implemented in lib/recommendation.ts, uses quiz responses to assign weight adjustments to various careers. Each answer influences the weights based on predefined mappings. The algorithm then sorts the careers by their cumulative weight and integrates the sentiment score to ensure that recommended careers not only match user preferences but also reflect a positive industry outlook.

**5.3 Discussion**

The combination of quiz responses and sentiment analysis allows for a nuanced recommendation. Instead of merely displaying a sentiment score, the system leverages these scores internally to rank careers, ensuring that the top recommendations align well with both the user’s preferences and broader career trends.

**Chapter 6: Evaluation and Testing**

**6.1 Functional Requirements**

The system is evaluated on the following core functionalities:

* **Quiz Submission:** Users can complete and submit a quiz; all responses are stored and processed.
* **Career Recommendation:** The recommendation API returns personalized career options based on user inputs.
* **Career Selection:** Users can select and update their chosen career.
* **Roadmap Visualization:** The selected career’s milestones are accurately displayed in a sequential, interactive format.

**6.2 Non-Functional Requirements**

The evaluation also covers:

* **Usability:** The interface is intuitive, responsive, and accessible.
* **Performance:** Serverless functions and API endpoints return responses quickly, ensuring a smooth user experience.
* **Reliability:** The system handles edge cases (such as incomplete quiz responses) gracefully, with proper error messages and redirection.
* **Scalability:** The architecture supports an increasing number of users without degradation in performance.
* **Security:** User data is protected through secure authentication (using Auth.js) and safe API interactions.

**Chapter 7: Discussion and Future Work**

**7.1 Discussion**

*Rookies* demonstrates that a full-stack web application can effectively combine sentiment analysis with personalized career guidance. The integration of user quiz responses with data-driven sentiment scores provides a unique perspective on career suitability, offering more than just static recommendations.

**7.2 Limitations**

Key limitations include:

* **Data Quality:** The accuracy of sentiment analysis depends on the quality and relevance of the input data.
* **User Feedback:** Initial recommendations may require refinement based on user feedback and further data collection.
* **Scalability of Analysis:** As more career data becomes available, the algorithm may need to be adjusted to handle larger datasets.

**7.3 Future Work**

Future research could focus on:

* **Enhanced Machine Learning Models:** Integrating more sophisticated ML techniques to improve recommendation accuracy.
* **Real-Time Data Integration:** Incorporating real-time industry data and user feedback to dynamically update sentiment scores.
* **User Personalization:** Expanding the personalization features to include user profiles, historical data, and adaptive learning pathways.
* **Extensive Usability Testing:** Further evaluating the HCI aspects to improve accessibility and user satisfaction.

**Chapter 8: Conclusion**

**8.1 Summary of Contributions**

This project has successfully developed a full-stack web application—*Rookies*—that provides personalized career recommendations and roadmap visualizations for first-year computer science students. Key contributions include:

* The design and implementation of an interactive quiz interface.
* A novel recommendation algorithm that integrates sentiment analysis.
* A comprehensive system architecture using modern web technologies.
* An emphasis on HCI principles, ensuring the system is user-friendly and accessible.

**8.2 Final Remarks**

*Rookies* bridges the gap between academic guidance and industry trends by delivering data-driven, personalized career recommendations. This project demonstrates the potential of combining sentiment analysis with web technologies to empower students in making informed career decisions.

**Bibliography**

*Include a comprehensive list of all references, journals, websites, and books cited throughout the thesis.*

**Appendices**

* **Appendix A:** Detailed Database Schema and ER Diagram
* **Appendix B:** Key Code Listings and API Documentation
* **Appendix C:** User Survey Results and Usability Testing Data