



PrepScore Career Profile Optimizer

Minna Joby

Roll No: 35

Reg.No.: KTE24MCA-2038

Guided By : Prof. Shilpa M. Thomas

Project Coordinator : Dr. Reena Murali

Department of Computer Applications

Rajiv Gandhi Institute Of Technology, Kottayam

Introduction



- PrepScore is a smart web application that analyzes a student's professional profile—including skills, education, and experience—to generate a real-time "Career Readiness Score."
- It addresses the critical uncertainty students face by transforming a static resume into a dynamic tool, providing a clear, quantitative measure of their job market readiness.
- The system empowers users with actionable, data-driven recommendations, guiding them on how to strategically improve their profile and boost their career prospects.

Problem Statement

- **Career Uncertainty:** Students lack a clear, data-driven way to measure their job market readiness and identify specific areas for improvement.
- **Static Tools:** Existing tools like resumes and standard online profiles are passive and offer no quantitative analysis or actionable feedback.
- **The Gap:** There is a clear gap between having a list of qualifications and understanding their actual value in the competitive job market.

Objectives

- **Develop a Full-Stack Platform:** Build a secure Django web application with a PostgreSQL database, featuring full user authentication and CRUD profile management.
- **Integrate an Analysis Engine:** Implement a scikit-learn Machine Learning model for live scoring and a rule-based engine for personalized recommendations.
- **Design a Professional UI:** Create a responsive interface with a data visualization dashboard to clearly present the analysis to the user.

Scope

- A secure, locally-hosted Django web application for comprehensive user profile management (CRUD).
- An integrated analysis engine that uses a scikit-learn model to generate a live "PrepScore" and provides rule-based improvement suggestions.
- The scope is a proof-of-concept; it does not include cloud deployment or training on real user data.

Relevance

- Provides a data-driven tool for career self-assessment, offering clear benchmarks and actionable guidance to improve job readiness.
- Moves beyond static resumes to a dynamic profile analysis that better reflects current, skill-based hiring trends.
- Showcases a complete project lifecycle, demonstrating in-demand skills in full-stack web development and applied machine learning.

Existing Systems

- Tools like resume builders and LinkedIn are passive platforms for storing data, not for active analysis.
- They lack a data-driven feature to provide a real-time, quantitative score of a user's job market readiness.
- They do not offer automated, actionable recommendations on how a user can specifically improve their profile.

Proposed System

- An active system that analyzes a user's profile to generate a real-time "PrepScore," providing a clear benchmark of their job readiness.
- Provides personalized, data-driven recommendations to help users strategically address weaknesses in their profile.
- Moves beyond passive data storage to an active tool that empowers users to improve their career prospects through targeted guidance.

System Requirements

Software Requirements

- **Operating System:** Windows, Linux
- **Backend:** Python 3.+, Django 5.+
- **Frontend:** HTML5, CSS3

Hardware Requirements

- **Development:** Standard PC/Laptop
Minimum 4GB of RAM
- **End-User:** Any device with a modern browser

Development Tools

- **Version Control:** Git & GitHub
- **Code Editor:** Visual Studio Code
- **Database GUI:** pgAdmin

Development Methodology



- **Model** : An **Iterative and Incremental** model was used to build the application in distinct, manageable phases.
- **Process** : Each major feature (like Authentication or the ML Engine) was developed, tested, and refined as a complete unit before starting the next.
- **Benefits** : This approach allowed for early testing of core functionalities, better risk management, and the flexibility to adapt the design throughout the project lifecycle.

Development Methodology

- **Phase 1 - Foundation:** Project setup, database configuration, and a complete user authentication system.
- **Phase 2 - Core Functionality:** Implementation of a responsive UI and full CRUD (Create, Read, Update, Delete) for all profile modules.
- **Phase 3 - Intelligence:** Development of the end-to-end ML pipeline for live scoring and the rule-based recommendation engine.
- **Version Control:** Git and GitHub were used continuously to track all changes with descriptive commits and maintain a secure project backup.

Design

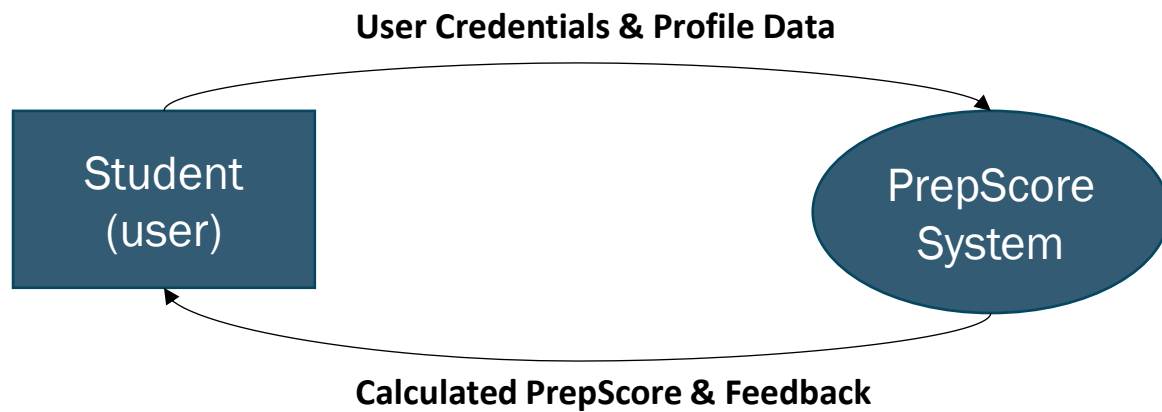


FIG 1 : DFD LEVEL 0

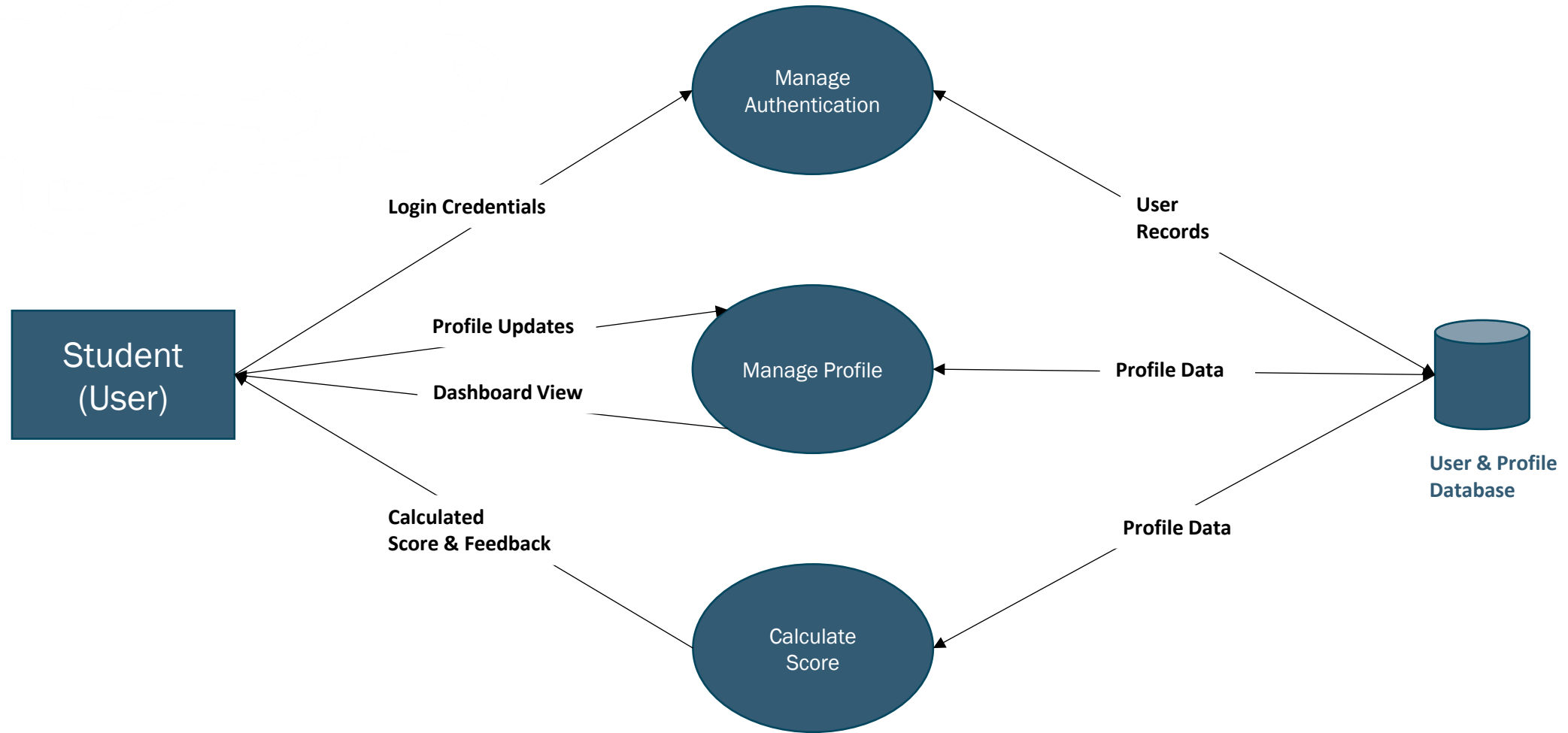


FIG 2 : DFD LEVEL 1

Implementation Details

- **Backend:** A **Django** application using a **PostgreSQL** database, with secure CRUD views (@login_required) and a custom authentication backend.
- **Machine Learning Pipeline:**
 - ✓ A RandomForestRegressor was trained with scikit-learn on a synthetic dataset.
 - ✓ The trained model is loaded by Django for live, real-time predictions.
- **Scoring Engine:** A hybrid scoring system was implemented, using a rule-based "gate" for new users and the ML model for established profiles.

Implementation Details

- **Frontend** : A clean and maintainable frontend was achieved using Django's template inheritance with a main `base.html` and a separate `dashboard_base.html` for the logged-in "app shell."
- **Data Visualization** : The dashboard provides a rich user experience with an animated **donut chart** (CSS), a **pie chart** (Chart.js), and **bar charts** (Bootstrap Progress Bars).

Current Status of Work

- **Status:** The project is **100% feature-complete** and functions as a robust, locally-hosted prototype.
- **Key Achievements:** All major features are implemented, including secure user authentication, full CRUD profile management, and a live, integrated machine learning scoring engine.
- **Readiness:** The application is stable, tested, and fully prepared for final evaluation and demonstration.

Results

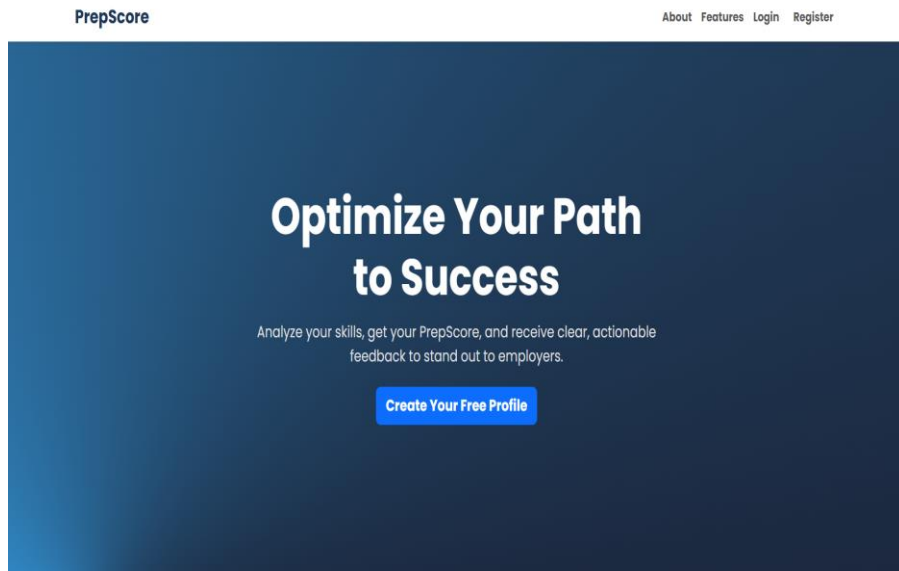


FIG 3 : HOMEPAGE

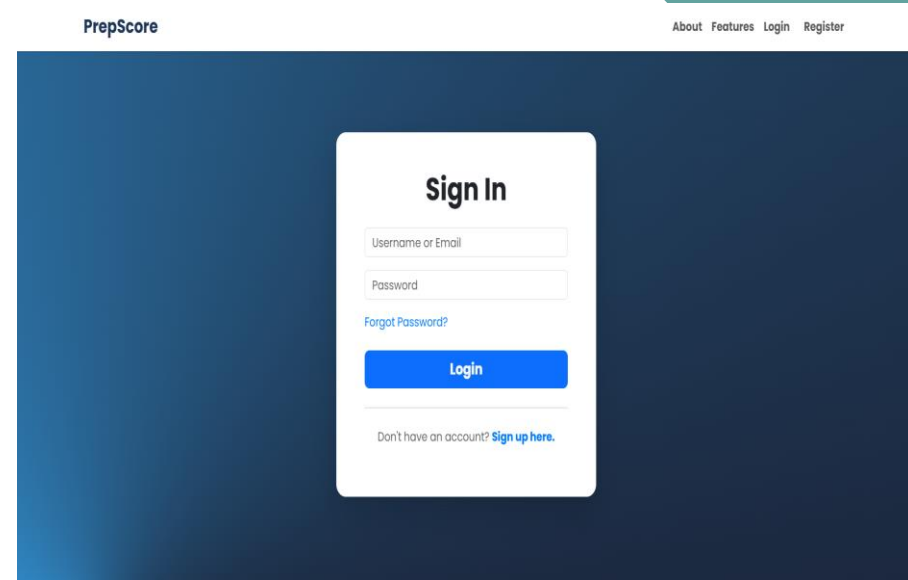


FIG 4 : LOGIN

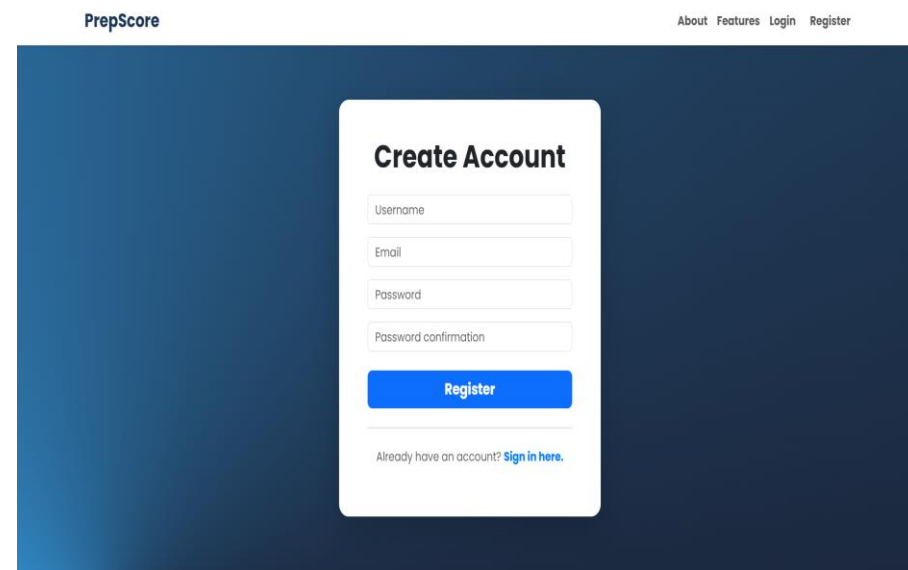


FIG 5 : REGISTER

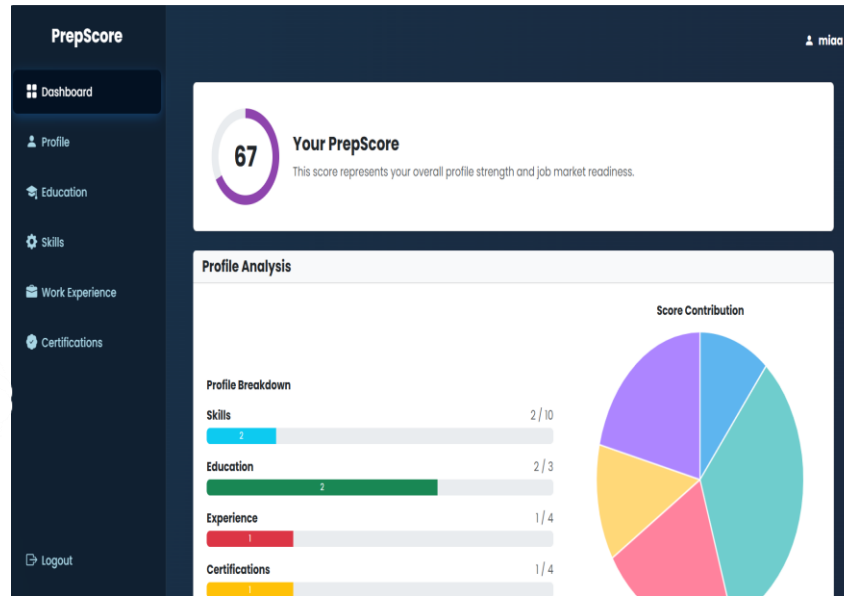


FIG 6 : DASHBOARD & LIVE SCORING

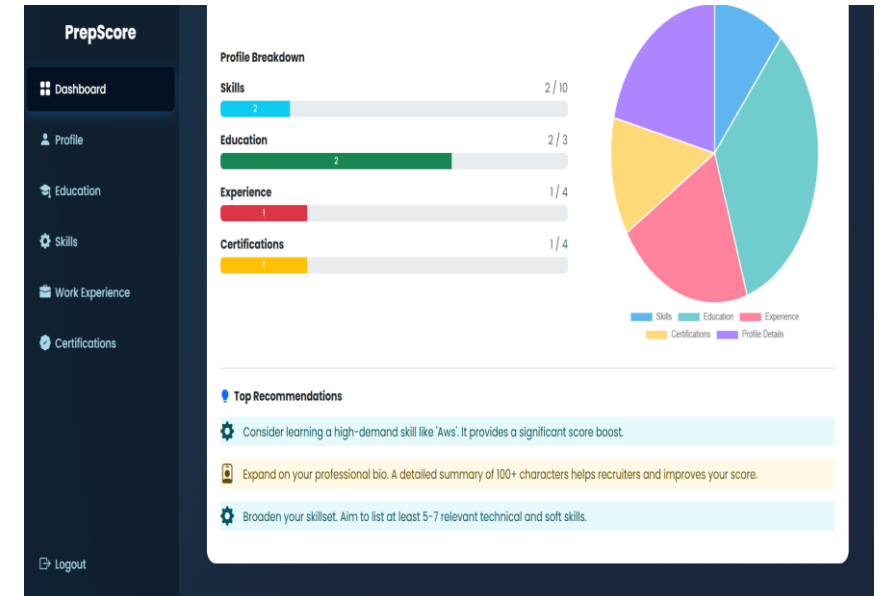


FIG 7 : DYNAMIC FEEDBACK

The PrepScore profile management interface allows users to update their professional headline, location, and bio. It includes fields for the following information:

- Headline: Developer
- Location: Kerala
- Bio: Hello Guys
- LinkedIn url: <https://www.linkedin.com/in/minnajiboy>
- Github url: <https://www.github.com/in/minnajiboy>

A 'Save Changes' button is located at the bottom of the form.

FIG 8 : PROFILE MANAGEMENT

The PrepScore CRUD functionality interface allows users to add, edit, or delete their academic qualifications. It includes a 'Manage Your Education' section with a 'Add New Qualification' form and a 'Your Current Qualifications' table.

Add New Qualification Form:

- Degree:
- Institution:
- Year:
- Save button

Your Current Qualifications Table:




Qualification	Year	Edit	Delete
Master of Computer Applications	2020	Edit	Delete
Bachelor of Science in Physics	Vimala College - 2024	Edit	Delete

FIG 9 : CRUD FUNCTIONALITY

Analysis Of Results

Application Functional Validation

All core application features were tested and found to be fully functional, secure, and responsive.

- **User Management:**  **Success.**
- **CRUD Operations:**  **Success.**
- **UI/UX:**  **Success.**



Quantitative Model Performance

The integrated RandomForestRegressor model was evaluated and demonstrated a high degree of accuracy.

Metric	Result	Meaning
MAE	1.00	The score is, on average, only off by 1.00 points. (Very Accurate)
R ² Score	0.99	The model explains 99% of the score's logic. (Very Reliable)

Hybrid Scoring Engine Validation

The live scoring engine was tested for logical consistency and real-time responsiveness.

- "Zero Score" Test:  Passed.
- "Dynamic Update" Test:  Passed.

Feature Importance

An analysis of the model's feature importances confirms it learned logical patterns from the training data.

```
Top 10 most important features:
      feature  importance
2      num_experiences  0.394571
0      num_skills      0.208573
4      has_bio         0.172176
3      num_certifications 0.152655
1      num_educations   0.041876
6      has_linkedin     0.007236
5      has_headline     0.005050
17 has_skill_machine_learning 0.002608
20 has_skill_python     0.002285
9      has_skill_azure   0.002199
```

- **Key Insight:** The model accurately identified **work experience** as the most important feature for a high score, followed by skills and bio.
- **Recommendation Synergy:** The recommendation engine is aligned with this finding, prioritizing suggestions for gaining practical experience to provide the most impactful advice.

Conclusion

- **Project Success:** Successfully developed a feature-complete, full-stack Django application that meets all project objectives, from secure user management to a responsive UI.
- **ML Integration:** The core goal was achieved by successfully integrating a scikit-learn model for live, data-driven scoring and a rule-based engine for actionable user recommendations.
- **Final Outcome:** The final prototype is a robust and effective tool that successfully validates the concept of an intelligent career profile optimizer.

Git History

Commits

main	All users	All time
Commits on Oct 1, 2025		
Feat: Implement final data-visualization dashboard UI minnajorby committed yesterday	c79ee1a	
Commits on Sep 14, 2025		
Feat: Implement suggestion engine and enhance scoring algorithm minnajorby committed 3 weeks ago	3b88bc5	
Commits on Sep 1, 2025		
Feature: Build complete CRUD functionality and dashboard UI minnajorby committed on Sep 1	f44222e	
Commits on Aug 30, 2025		
Build professional About and Features pages with Bootstrap minnajorby committed on Aug 30	8571e63	
Commits on Aug 19, 2025		
Feature: Add Skill editing and refactor Dashboard UI minnajorby committed on Aug 19	8847c5a	
Commits on Aug 5, 2025		
Design: Implement homepage layout with header and footer minnajorby committed on Aug 5	6f82cea	
Commits on Jul 29, 2025		
Feature: Implement user authentication with login, logout, and dashboard minnajorby committed on Jul 29	5842eed	
Commits on Jul 22, 2025		
Secure secrets using .env file minnajorby committed on Jul 22	9a570f7	
Feature: Create and display basic home page minnajorby committed on Jul 22	8b37335	
Commits on Jul 20, 2025		
Initial Commit - prepscore project minnajorby committed on Jul 20	b4e1392	

Bibliography

- **Django Documentation:** docs.djangoproject.com
 - Primary resource for backend development, ORM, security, and template logic.
- **Scikit-learn User Guide:** scikit-learn.org/stable/user_guide.html
 - Used for RandomForestRegressor implementation, train_test_split, and model evaluation metrics.
- **PostgreSQL Documentation:** postgresql.org/docs/
 - Reference for the relational database setup and data management.
- **Pandas User Guide:** pandas.pydata.org/docs/user_guide/index.html
 - Utilized for creating and manipulating the training dataset from the generated data.

Thank you

