BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2026
http://excel.fit.vutbr.cz

Web Application for Teacher Evaluations

Anastasiia Chumachenko

Abstract
This work presents a web application for structured evaluation of teachers and courses. The system enables students to provide ratings, write reviews, and evaluate existing feedback. A ranking model is introduced to prioritize relevant reviews using multiple factors such as reputation and time decay. The platform improves decision-making in course and supervisor selection.

xchuma00@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Choosing a suitable teacher or thesis supervisor is a difficult task, especially for first-year students who lack prior experience and reliable information.

In practice, students share feedback through Discord servers or shared spreadsheets (e.g., Microsoft Excel). These solutions are usable, but not convenient. Spreadsheets can be easily modified or overwritten, which affects reliability. Discord preserves information, but makes it difficult to navigate and compare feedback. Similar issues have been observed in systems based on user-generated reviews [1].

This problem is closely related to reputation and feedback systems, which are widely used in online platforms [2]. However, in academic environments, similar solutions are typically missing or implemented only in an informal way.

The goal of this project is to provide a structured platform for collecting and presenting student feedback. The platform allows users to evaluate teachers and courses, and explore aggregated results.

The main contribution of this work is the combination of structured reviews, user interaction, and a ranking model that prioritizes relevant feedback.

2. System Overview

The platform provides a single place where students can explore teachers, read feedback, and contribute their own experiences.

The main entry point of the system is the overview page (Fig. 2). It displays a set of teachers bookmarked by the user, allowing quick access to relevant profiles. The page also shows the most recent reviews across all teachers, providing a simple overview of current activity.

Each teacher has a dedicated detail page (Fig. 1). This page contains the overall rating, a distribution of ratings, and reviews linked to specific contexts such as courses or thesis supervision. New reviews can be created directly on this page.

The rating visualization is inspired by platforms such as IMDb [3], where both the average score and its distribution are shown.

Interaction with reviews is an essential part of the system. Users can evaluate the usefulness of existing reviews through voting, allowing the system to distinguish between more and less relevant feedback.

User-related functionality is represented by the profile page (Fig. 3), where users can manage their account and access information about their activity.

3. Core Features

The platform supports structured feedback that can be both created and explored in a consistent way. Reviews combine a numerical rating with a short text and are always tied to a specific context, such as a course or thesis supervision. This prevents overly general feedback and makes individual entries easier to interpret.

User interaction is incorporated through voting on reviews. This provides a lightweight mechanism for identifying which feedback is considered useful and helps surface relevant reviews as the amount of data grows [4].

In addition to individual reviews, the platform focuses on aggregated information. Teachers can be ordered by rating or by the number of reviews, which highlights the most relevant or most discussed cases without introducing a separate comparison interface.

Together, these mechanisms connect individual feedback with its aggregated representation and allow users to navigate the data efficiently.

4. Comment Weighting Model

A key component of the platform is the ranking of reviews. Instead of displaying reviews in chronological order, the platform prioritizes them based on their relevance.

The ranking is based on multiple factors, including author reputation, user votes, and the age of the review. These factors are combined into a single score that determines the order in which reviews are displayed (Equation 1).

The use of logarithmic scaling prevents vote counts from dominating the score, while the decay factor gradually reduces the influence of older reviews. This helps balance popularity with recency. This approach is conceptually related to ranking methods used in recommender systems [5].

5. Architecture and Technologies

The platform follows a client–server architecture.

The frontend is built using React, which allows the interface to remain responsive and interactive while working with dynamic data. The backend is provided by Supabase, which exposes database functionality through a REST-like API and handles authentication.

PostgreSQL is used as the underlying database system. It stores all core entities, including teachers, reviews, and user interactions, and enables efficient aggregation of ratings and statistics.

This setup makes it possible to implement the platform without a custom backend server while still maintaining a structured data model and consistent access control.

6. Conclusion

The project shows that structuring student feedback makes it easier to use and interpret in practice.

The ranking model further improves usability by prioritizing relevant reviews instead of displaying them in chronological order.

Future work may focus on refining the ranking model and extending the system based on real usage.

Acknowledgements

I would like to thank my supervisor, prof. Ing. Adam Herout, Ph.D., for his guidance, feedback, and valuable suggestions throughout this work.

References

[1] Nan Hu, Paul A Pavlou, and Jie Zhang. What we rate when we rate online reviews. In *Information Systems Research*, 2008.

[2] Paul Resnick and Richard Zeckhauser. Reputation systems. In *Communications of the ACM*, 2000.

[3] Imdb: Internet movie database. https://www.imdb.com/. Accessed: 2026-04.

[4] Kristina Lerman. Social voting and information filtering. In *Proceedings of the International Conference on Weblogs and Social Media*, 2010.

[5] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems. In *IEEE Transactions on Knowledge and Data Engineering*, 2005.