

Title: Student-Directed Project Proposal and Grading Platform for Database Courses

Xinqiong Yu, Ling Zhang, Teng Zhao

Team Spirit

CS6460

December 10th, 2016

We plan to submit this paper to *Journal of Open Research Software* (<http://openresearchsoftware.metajnl.com/>), a journal that publish research papers that cover different aspects of creating, maintaining and evaluating open source research software. By publishing on this open access journal, we aim to inform educators for database-related courses about our software, and encourage their interest in applying our application to facilitate their teaching process. Also, the software and the paper will be citable, and reuse will be tracked.

This page is for instructor/TA/peer feedback reference only, not part of the submission to the journal

Student-Directed Project Proposal and Grading Platform for Database Courses

Xinqiong Yu¹, Ling Zhang¹, Teng Zhao¹

1. Department of Computer Science, Georgia Institute of Technology, Atlanta, GA 30332

Abstract: The student-directed project proposal and grading platform is aimed to help instructors in database courses to monitor projects designed and carried out by students. The platform also acts a convenient tool to allow students to form teams on projects they are interested in. The autograder in this platform allows for customized, automated and fast evaluation of project submissions.

Keywords: Database course, Educational Technology, Student-directed platform, project proposal, autograder

1. Introduction

In most learning processes, there are usually two sets of knowledge involved: one set of knowledge is what students grasped in the past - their existing expertise; the other one is what they are going to learn in this learning process, which is the new knowledge. How well students can integrate the new knowledge into their existing system will largely depend on how well they can apply the new knowledge practically to solve real-world problems, especially problems they faced in the past [1]. Currently, although a lot of courses start to apply the concept of Problem-Based Learning and Project-Based Learning [2], there is still the drawback that the project/problem is fixed, and students are forced to work on the same project topic in order to learn the new knowledge, which could possibly result in consequences that students to some extent lose interests and initiative to further explore the project.

This application is inspired by challenges faced by the current educational courses, specifically how to formulate student-initiated project proposals and effectively evaluate project results.

(1) Background

The CS 6460-Education Technology course offered by Georgia Institute of Technology [3] is doing a great job providing customized learning experiences. In this course, students propose their own projects and are able to implement ideas they are interested in. However, it demands tremendous amount of teaching resources, which also makes it harder to standardize the evaluation criteria among different graders. On the other hand, courses such as database systems mandate a pre-defined project for all students, despite their background or interests, which makes the evaluation easier, but fails to provide tailored educational experiences.

We've seen tools helpful in certain aspects of the customized project learning experience. For instance, Github is a source management tool which also provides team collaboration features, and it encourages people to participate in different open source projects [4]. However, it's not for educational purpose, and there is no function provided for instructor feedback or proposal evaluation.

Software such as Slack or Flowdock are team communication tools, which allow live feedbacks and instant communication between any party [5, 6]. We can certainly apply them for educational purposes to facilitate proposal communication between students and instructors. But it's not much useful in terms of evaluation of the work, which requires tremendous manual efforts.

(2) Our solution

Our team designed a solution that combines all the advantages of existing tools, that is, to develop a platform for students so they can formulate their own ideas/problems related to their expertise, which can also allow them to apply the new knowledge they've gained from the course, to gather a group of students who have similar interests and required expertise, and to solve the problems or implement their ideas with the instructors' guidance. The ultimate goal is to allow students to learn the new knowledge in an efficient way with the help of each other's expertise, and at the same time, to expand or further improve their competence during the process of utilizing the platform to better understand the new knowledge.

This project is designed to be applied to a general database course. Database is currently one of the hottest topics in Computer Science, and a lot of people are eager to learn Database knowledge. Database courses are included in Computer Science education in most if not all schools, and are also a major component in MOOCs [7]. The problem with most database courses is that the topics of course projects are too narrow or even fixed, of which the most common one is library management system. Students with different expertise may want to create a different kind of system that allows them to apply their database knowledge and sharpen their domain expertise at the same time. The system could actually be helpful to the students in their everyday life, work or study, which gives the students more motivation to work on the project. For example, students with biology background may want to create a biology lab management system; students with economy background may want to create a simple trading management system, etc..

2. Availability

Operating system

This platform can be set up using IntelliJ or Eclipse on MAC or Windows Operating systems by the course instructor, while on student side, it is platform-independent and is compatible with web browsers (IE, Google Chrome, Firefox, Safari, etc.).

Programming languages

Java, MySQL, HTML, CSS, JavaScript

Additional system requirements

None

Dependencies

The platform uses Spring framework

List of contributors

Teng Zhao, Xinqiong Yu, Ling Zhang

Code repository

<https://github.com/zt7584/cs6460-education-tech-project.git>

Language

English

3. Implementation and Architecture

(1) Technical Details

This is a detailed list of programming languages and resources used to build this platform.

Application Type:

- Web Application
- REST APIs

Programming Languages:

- Java (Core language for backend development)
- HTML, CSS, JavaScript (Core languages for frontend development)
- Material-UI (UI library)

Frameworks:

- Spring Boot (For fast and simple configuration)
- Spring MVC (For developing web application)

Databases:

- MySQL (Selected as an example of traditional relational database to run SQL statements)
- MongoDB (Selected as an example of new NoSQL database to run NoSQL scripts)

Other resources:

- SQL statements static analysis algorithm implementation [8]
- SQL runtime performance analysis algorithm implementation [9]
- NoSQL scripts static analysis algorithm implementation [10]
- NoSQL runtime performance analysis algorithm implementation [10]

(2) Modules

The platform allows students to formulate their own business logic for the database course project, to form a group of students to work out the business logic part and database part, to allow instructor to define grading rubrics, and finally to grade the project automatically according to instructor's grading rubrics. The core modules of the platform include:

User Management Module:

1. User can register as a Student or as an Instructor role
2. User can login or logout from the platform, each user is identified with a username (Figure)

Student Proposal Module (Figure 1):

1. Student can make a proposal and submit to Instructor (Figure 1A)
2. Instructor can review the proposal (Figure 1B)
3. Instructor can approve the proposal
4. Instructor can decline the proposal with suggestions so that Student can revise and submit again

A

Welcome Teng Zhao!

Actions

Create Proposal

Online Judge

Proposals

| Proposal Id | Proposal Title | Proposal Description | Proposal Status | Created At | Last Updated At | Action |
|-------------|----------------|----------------------|-----------------|-----------------------|-----------------------|-------------------------|
| 1 | Proposal-0 | This ia... | Approved | 2016-10-26 20:20:47.0 | 2016-10-26 20:20:47.0 | <div>View Details</div> |
| 2 | Proposal-1 | This is... | Pending | 2016-10-26 22:45:55.0 | 2016-10-26 22:45:55.0 | <div>View Details</div> |

B

Proposals

Pending Proposals

| Proposal Id | User Id | Proposal Title | Proposal Description | Proposal Status | Created At | Last Updated At | Action |
|-------------|---------|----------------|----------------------|-----------------|-----------------------|-----------------------|--|
| 2 | 2 | Proposal-1 | This is... | Pending | 2016-10-26 22:45:55.0 | 2016-10-26 22:45:55.0 | <div>View Details</div> <div>Approve</div> |
| 3 | 3 | Proposal-2 | This is... | Pending | 2016-10-26 23:05:14.0 | 2016-10-26 23:05:14.0 | <div>View Details</div> <div>Approve</div> |

Approved Proposals

| Proposal Id | User Id | Proposal Title | Proposal Description | Proposal Status | Created At | Last Updated At | Action |
|-------------|---------|----------------|----------------------|-----------------|-----------------------|-----------------------|-------------------------|
| 1 | 2 | Proposal-0 | This ia... | Approved | 2016-10-26 20:20:47.0 | 2016-10-26 20:20:47.0 | <div>View Details</div> |

Figure 1. Student Proposal Module. (A) Student interface. (B) Instructor interface.

Project Team Management Module:

1. Once a proposal is approved, other Students can make request to join the team for that proposal
2. Student can accept the joining team request
3. Student can decline the joining team request
4. Instructor can see and edit all the proposals and teams in the class

Database REST API Module:

1. Student can make REST API calls with input SQL query commands and receive response of execution results
2. Different types of database are provided (MySQL is used as an example of relational database; MongoDB is used an example of NoSQL database)
3. Each REST API call is marked with Student's username and authorized with Student's credentials

Online Judge Module (Figure 2):

1. SQL Online Judge Module (Figure 2A) : Student can type their SQL query in a page, execute it, and get response and SQL execution performance statistics (very similar to a coding online judge)
2. MongoDB Online Judge Module (Figure 2.2): Student can type their MongoDB query in a page, execute it, and get response and MongoDB execution performance statistics

A

The screenshot shows the 'Online Judge' interface for SQL. At the top, there's a blue header with the text 'Online Judge'. Below it, a dropdown menu is set to 'MySQL'. A text input field is labeled 'Enter Your Query Here'. A green 'Execute' button is positioned below the input field. The results section, titled 'Statistics', shows 'Execution Time: 8 ms' and 'Number of Join: 0'. Below this, the 'Execution Result' is displayed as a JSON array of three user objects.

```
Database: MySQL
Enter Your Query Here
Execute

Statistics:
• Execution Time: 8 ms
• Number of Join: 0

Execution Result:
[
  {
    "email": "instructor@gatech.edu",
    "id": 1,
    "name": "Instructor Inspiration",
    "password": "password",
    "role": 0
  },
  {
    "email": "tzhao@spsu.edu",
    "id": 2,
    "name": "Teng Zhao",
    "password": "1234",
    "role": 1
  },
  {
    "email": "zt7584@gmail.com",
    "id": 3,
    "name": "Test Zhao",
  }
```

B

The screenshot shows the 'Online Judge' interface for MongoDB. At the top, there's a blue header with the text 'Online Judge'. Below it, a dropdown menu is set to 'MongoDB'. A text input field contains the query '{first:'mickey',last:'mouse'}'. A green 'Execute' button is positioned below the input field. The results section, titled 'Console', shows 'Statistics' with 'Execution Time: 39 ms' and 'Number of Fields Queried: 2'. Below this, the 'Execution Result' is displayed as a JSON array containing one document.

```
Database: MongoDB
{first:'mickey',last:'mouse'}
Execute

Console
Statistics
• Execution Time: 39 ms
• Number of Fields Queried: 2

Execution Result
[
  {
    "id": "583e74890626680ea8c4b02b",
    "first": "mickey",
    "last": "mouse"
  }
]
```

Figure 2. Online judge module. (A) SQL online judge module. (B) MongoDB online judge module.

Project Grading Module (Figure 3):

1. Instructor can define grading rubrics, such as minimum number of type of SQL statements, minimum number of different SQL statements, minimum number of table join, maximum execution time of SQL statements, etc.
2. Instructor can see all the execution logs from students across the semester
3. Instructor can start grading monitor which will grade all the SQL queries as a whole for a period of time, a grade report will be generated in the end

CS6460 Education Technology Project

Grading Rubrics

| Id | Name | Threshold | Operator |
|----|--|-----------|----------|
| 1 | Execution Time Upper Limit | 100 | <= |
| 2 | There must be one query that has more than or equal to 1 table joining | 1 | >= |

Grading ...

Student Id:

Stop Grading

Grading Progress

Grading starts at Sat Nov 19 2016 22:06:45 GMT-0500 (EST)
Query: [select * from user] is requested to execute at 2016-11-19 22:07:00.0

Figure 3. Project grading module.

4. Reuse Potential

This is an open source project, with the goal of supporting instructors in database courses. Database has been offered in most of institutes for Computer Science and many other relevant majors. Instructors in database courses could set up a website to use this platform for the database course they teach. This software serves small- to large- size course settings. For small courses with <20 students, the instructor could have students perform projects individually instead of forming teams. For bigger courses, team formation can be optional, to encourage students to develop more complicated database management systems.

5. Concluding Thoughts

This software uses Project-Based Learning as the pedagogical strategy for students to learn new knowledge in a non-traditional style. We believe that the platform we are building will not only help students to practice their Computer Science knowledge by proposing and executing their own projects, but also help students to gain teamwork experience, which is a major quality employers look for nowadays, while only requiring minimum amount of teaching resources.

Acknowledgements

We thank Dr. David Joyner and Ray Reese for their insightful instruction and suggestions on completing this project, and for peers in CS6460-Educational Technology course from Georgia Institute of Technology for their comments and discussion on our software.

Competing Interests

The authors have no competing interests to declare.

References

1. Jan W. and Bruce W.. (2009). Enhancing Learning by Integrating Theory and Practice. International Journal of Teaching and Learning in Higher Education 21(2), 258-265.
2. "Dr. Dave" M.. (2011, May 02). Problem-Based Learning and Project-Based Learning. Retrieved from <http://pages.uoregon.edu/moursund/Math/pbl.htm>
3. Georgia Institute of Technology College of Computing. Retrieved October 09, 2016, from <https://www.omscs.gatech.edu/cs-6460-educational-technology>
4. GitHub. (2016, October 8). In Wikipedia, The Free Encyclopedia. Retrieved 17:12, October 8, 2016, from <https://en.wikipedia.org/w/index.php?title=GitHub&oldid=743230862>
5. Slack (software). (2016, October 9). In Wikipedia, The Free Encyclopedia. Retrieved 10:24, October 9, 2016, from [https://en.wikipedia.org/w/index.php?title=Slack_\(software\)&oldid=743369692](https://en.wikipedia.org/w/index.php?title=Slack_(software)&oldid=743369692)
6. Comparison of continuous integration software. (2016, September 29). In Wikipedia, The Free Encyclopedia. Retrieved 12:53, September 29, 2016, from https://en.wikipedia.org/w/index.php?title=Comparison_of_continuous_integration_software&oldid=741752490
7. Build software better, together. Retrieved October 09, 2016, from <https://github.com/>
8. Carl G., Zhendong S., and Premkumar D.. (2004). JDBC Checker: A Static Analysis Tool for SQL/JDBC Applications. [Proceedings - International Conference on Software Engineering](#).
9. Omed H.. Top 6 Database Performance Metrics to Monitor in Enterprise Applications - Application Performance Monitoring Blog | AppDynamics. Retrieved October 09, 2016, from <https://blog.appdynamics.com/database/top-6-database-performance-metrics-to-monitor-in-enterprise-applications/>
10. Mongo Team. Analyze Query Performance. Retrieved October 09, 2016, from <https://docs.mongodb.com/manual/tutorial/analyze-query-plan/>
11. Georgia Institute of Technology College of Computing. Retrieved October 09, 2016, from <https://www.omscs.gatech.edu/cs-6400-database-systems-concepts-and-design>