Database Systems Project: Learning Management System

Project Submission Report

M. Alzhori and M. Kalaoun

Abstract—This report explores the implementation of a Learning Management System (LMS) for elementary schools. As a volunteer project for an NGO school, our LMS seamlessly integrates into existing frameworks, benefiting students, teachers, and administrators. The report covers background, implementation, experimental evaluation, and emphasizes practical contributions.

Index Terms— Education, Information Storage, Instructor interfaces, Learning management systems.

Contents

1 Introduction

THIS report presents a thorough exploration of our practical Learning Management System (LMS) project, carefully crafted to address the needs of elementary schools. Notably, this initiative is undertaken as a volunteer project for an NGO school, underscoring our commitment to supporting educational endeavors in underserved communities. In contrast to a standalone overhaul, our LMS functions as a valuable assisting tool, seamlessly integrated into existing educational frameworks to enhance daily operations. Serving as a centralized hub, students benefit from an intuitive platform that enables them to effortlessly navigate and manage their academic journey, encompassing subjects, lessons, homework assignments, and marks—all consolidated for ease of access. Teachers find streamlined processes facilitated by the LMS, allowing them to upload instructional materials and efficiently manage student marks. Simultaneously, administrators gain access to robust oversight tools that contribute to the well-coordinated functioning of the educational environment. By harmonizing website development and database functionality, our LMS underscores practicality and functionality in equal measure. This report provides a detailed and insightful examination of our volunteer project, shedding light on its practical contributions to the seamless operation and efficiency of elementary schools associated with the NGO.

2 Background

2.1 Context and Problem Analysis

The project requirements were gathered in a meeting with a school representative. After collecting these requirements, we analyzed them and used them to design the project's structure. We split the user needs into different types: students, teachers, and managers, each having specific functionalities.

For Managers:

- Add or remove students and teachers from the school database.
- Assign or remove teachers to grades and subjects.

Students can:

- Access lessons, homework, and marks for each subject.
- Change their password.

Teachers can:

- Assign or remove homework for their students in each subject.
- Add or remove lessons for their students.
- For each lesson, assign or remove documents for their students.
- Update their personal information.
- Add marks for their students.
- Add success stories for their students.

These functionalities will be explained further in the evaluation section.

2.2 Information Needs

The project requires data related to students, teachers, and managers, encompassing details such as names, unique usernames, passwords, and phone numbers. During the development phase, we utilized random data that aligns with the school's operational requirements, ensuring effective system testing.

3 IMPLEMENTATION

3.1 Conceptual Data Model (CDM)

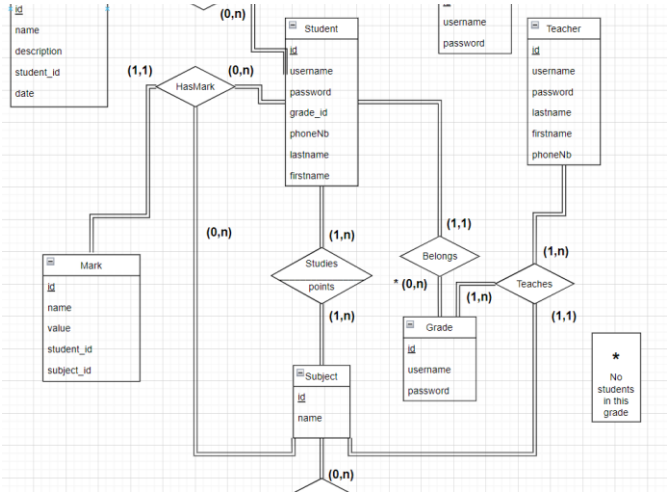

Figure 1: CDM - 1

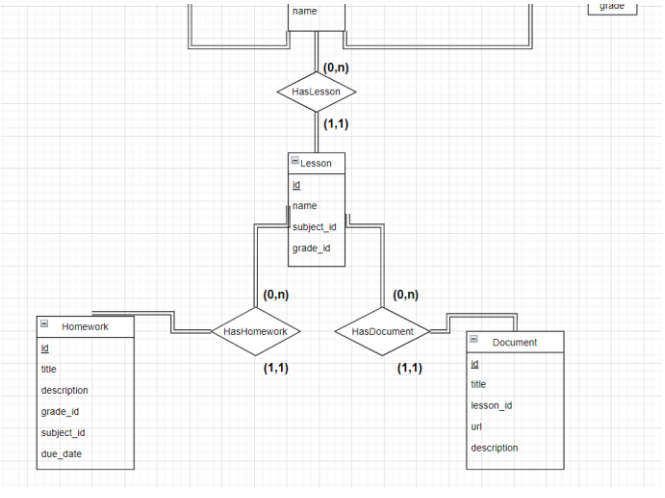

Figure 2: CDM - 2

3.2 Logical Data Model (LDM)

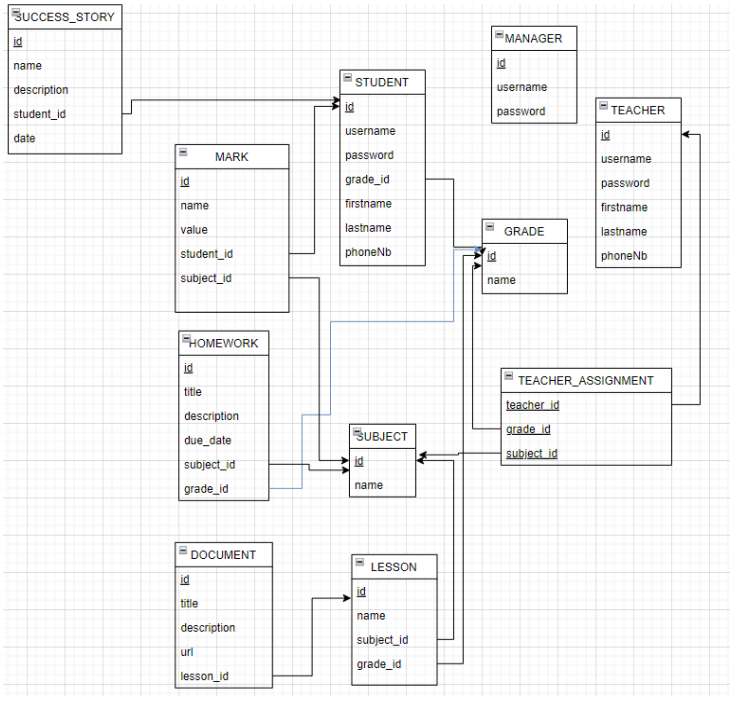

Figure 3: LDM

3.3 Software Application and Database Design and Implementation

The tools that were used in this project are the following:

- React v 18.2.0
- Node.js v 18.12.1
- Express.js v 4.18.2
- MySQL Community Server v 8.2.0
- Postman v10

4 Experimental Evaluation

This section will show the non-expert and expert uses of all the functionalities of the website. Non-expert users are bound to the software interface, providing for example keywords, through textboxes or drop-down menus. Expert users can interact through SQL statements.
In the following SQL statements, the symbol "?" represents inputs from the user interface.
For Managers:

- Add or remove students and teachers from the school database.

Non-Expert: page called all students/ all teachers which diplays students for each grade in a table, with a button to delete each, and a form to add a student

Expert:

➢ To add/ remove a student:

```
INSERT INTO STUDENT (username, password, grade_id, firstname, lastname, phoneNb) VALUES (?, ?, ?,?,?,?);
DELETE FROM STUDENT WHERE username = ?;
```

➢ To add/ remove a teacher:

```
INSERT INTO TEACHER (username, password, firstname, lastname, phoneNb) VALUES (?, ?,?,?,?);
DELETE FROM TEACHER WHERE id = ?;
```

- Assign or remove teachers to grades and subjects.

Non-Expert: Table inside all teachers page, with each row a button to delete the assignment and a form to add one

Expert:

➢ To add/ remove teacher assignmet:

```
INSERT INTO TEACHER_ASSIGNMENT (teacher_id, grade_id, subject_id) VALUES (?, ?, ?);
DELETE FROM TEACHER_ASSIGNMENT WHERE teacher_id = ? AND grade_id = ? AND subject_id = ?;
```

Students can:

- Access lessons, homework, and marks for each subject.

Non-Expert: Students choose a subject card, then press on homeworks, lessons, or marks to display a table with the information.

Expert:

- Lessons:

```
SELECT id, name FROM LESSON WHERE subject_id = ? AND grade_id = ?;
```

- Homeworks:

```
SELECT id, title, description, due_date FROM HOMEWORK WHERE subject_id = ? AND grade_id = ?;
```

- Marks:

```
SELECT id, name, value FROM MARK WHERE subject_id = ? AND student_id = ?;
```

- Change their password.

Non-Expert: Page called student profile where a form and a table are straightforward to update password.

Expert:

```
UPDATE STUDENT SET ? WHERE username = ?;
```

Teachers can:

- Assign or remove homework for their students in each subject.

Non-Expert: Table of homeworks with button to delete each and a form to add ones

Expert:

```
INSERT INTO HOMEWORK (title, description, due_date, subject_id, grade_id) VALUES (?, ?, ?, ?, ?);
DELETE FROM HOMEWORK WHERE id = ?;
```

- Add or remove lessons for their students.

Non-Expert: Table of lessons with button to delete each and a form to add ones

Expert:

```
INSERT INTO lesson (name, subject_id, grade_id) VALUES (?, ?, ?);
DELETE FROM DOCUMENT WHERE lesson_id = ?;
```

- For each lesson, assign or remove documents for their students.

Non-Expert: Besides each lesson there's a show documents page where documents are in a table with button to delete each and a form to add ones.

Expert:

```
INSERT INTO document (title, description, URL, lesson_id) VALUES (?, ?, ?, ?);
DELETE FROM DOCUMENT WHERE id = ?;
```

- Update their personal information.

Non-Expert: Page called teacher profile where a form and a table are straightforward to update information.

Expert:

```
UPDATE TEACHER SET ? WHERE username = ?;
```

- Add marks for their students.

Non-Expert: A form to add marks for their students

Expert:

```
INSERT INTO MARK (name, value, student_id, subject_id) VALUES (?, ?, ?, ?);
```

- Add success stories for their students.

Non-Expert:

Expert:

```
INSERT INTO SUCCESS_STORY (month, name, description, student_id, grade, subject_id) VALUES (?, ?, ?, ?, ?, ?);
```

5 Conclusion

In wrapping up our Learning Management System (LMS) project, tailored for an NGO school, we successfully fused technology into elementary education. This collaborative endeavor, backed by a robust database structure and user-friendly interfaces, ensures practicality for students, teachers, and administrators. The project stands as a testament to the impactful integration o f technology in education, streamlining operations and contributing to the efficiency of el-

ementary schools.

REFERENCES

[1] "Node.js v21.5.0 documentation," [Online]. Available: https://nodejs.org/docs/latest/api/.
[2] "React Reference," [Online]. Available: https://react.dev/reference/react.
[3] "MySQL Workbench," [Online]. Available: https://dev.mysql.com/doc/workbench/en/.
[4] "Postman documentation," [Online]. Available: https://learning.postman.com/docs/introduction/overview/.
[5] "Resources," [Online]. Available: https://expressjs.com/en/resources/glossary.html.