Tribhuvan University

Institute of Engineering

PULCHOWK CAMPUS

Pulchowk, Nepal

A

PROJECT REPORT

on

Lesson Plan Management System
[Software Engineering]

Submitted By:

Aadarsh Khatri (076BCT002)

Anish Sapkota (076BCT008)

Kushal Subedi (076BCT031)

Nabin Khanal (076BCT036)

Submitted To:

Department of
Electronics and Computer
Engineering

Submitted On:

August 25, 2022

Acknowledgements

Abstract

The "Lesson Plan Management System" project on Software Engineering, which is developed using JavaScript is a tool for teachers to create and maintain their teaching plan. It makes it easier for teachers to manage subjects, chapters, topics along with the assignments and resources associated with them. The project meets the objective to implement software engineering principles in real world project.

Contents

1   Introduction

Lesson Plan Management System (LPMS) is designed for teachers to manage and maintain their teaching plan throughout the semester. It is tedious work for most of the teachers to keep track of all the teaching materials for every subject they teach. LPMS aims to solve this problem by providing a systematic way of managing the syllabus. It also helps in managing all the assignments and resources associated with each chapter of every subject. Also, after the plan is designed once, it can be used with little to no modification for subsequent semesters.

2   Objectives

The primary goal of the project is to acquire comprehensive understanding of software engineering principles and their applications. Particularly, the project has the following sub-objectives:

1. To implement the web app logic in JavaScript

2. To be familiarized with software engineering principles

3. To put libraries into good use without completely compromising the learning process.

4. To maintain and update the legacy code

5. To work in the spirit of team within a time constraint.

3   System Architecture

The system can be mainly categorized into two components:

1. Backend System

   It is implemented in Node.js (Javascript Runtime Environment) and contains all the data processing logic. Express.js is used as a webserver and ejs is used as a templating engine. MVC (Model-View-Controller) pattern is being used. Mongoose is used as a mongodb driver.

2. Frontend System

   It is implemented with HTML, CSS and JavaScript. HTML is generated by templating engine (ejs), "Tailwind" framework is used for CSS. "Script" tag

is used within the ejs file to implement Javascript logic.

In the system, there is only one user role: Teacher, who can create an account and maintain their teaching plan.

4 Methodology

4.1 Agile Model

The Lesson Plan Management System is often difficult or impossible to predict how it will evolve as time passes. Teacher's demands changes rapidly, existing functionality may become obsolete and additional functionality becomes necessity. Students and upcoming batches must be quick on their feet if they are to accommodate the rapid changes. Agility can be applied to any software process. However, to accomplish this, it is essential that the process be designed in a way that allows the project team to adapt tasks and to streamline them, conduct planning in a way that understands the fluidity of an agile development approach, eliminate all but the most essential work products and keep them lean, and emphasize an incremental delivery strategy that gets working software to the customer as rapidly as feasible for the product type and operational environment.

4.2 Use Cases

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case itself might drill into a lot of detail about every possibility, and can help provide a higher-level view of the system. Use case diagrams are the blueprints for the system:
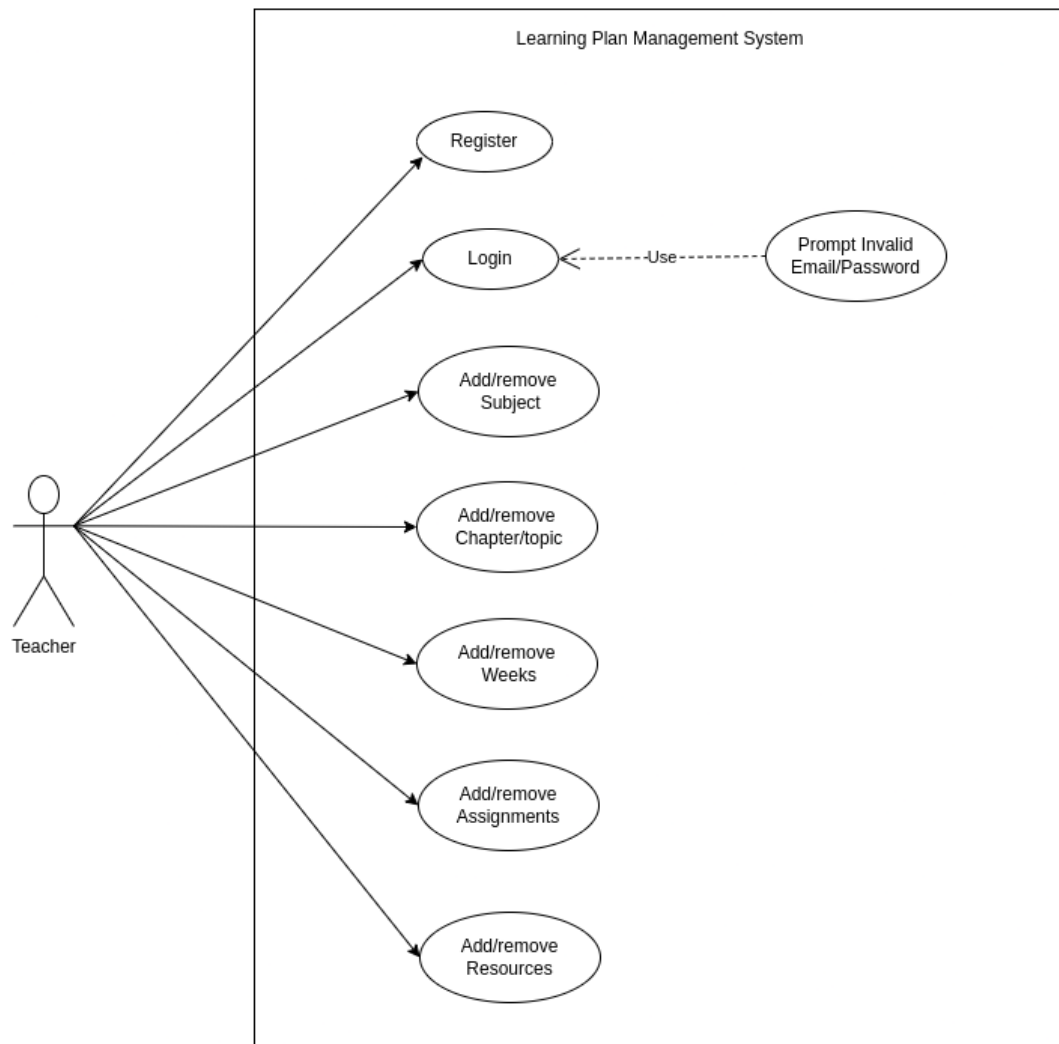
Figure 1: Use case diagram to meet major requirements

4.3   Context diagram

A system context diagram (SCD) defines the boundary between the system, or part of a system, and its environment, showing the entities that interact with it. This diagram is a high level view of a system. It is similar to a block diagram.
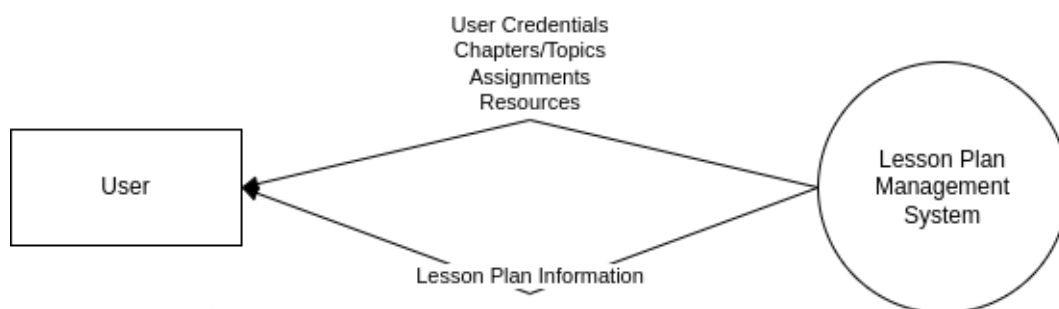

Figure 2: Context diagram

4.4   Level one- DFD

In 1-level DFD, the context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses.
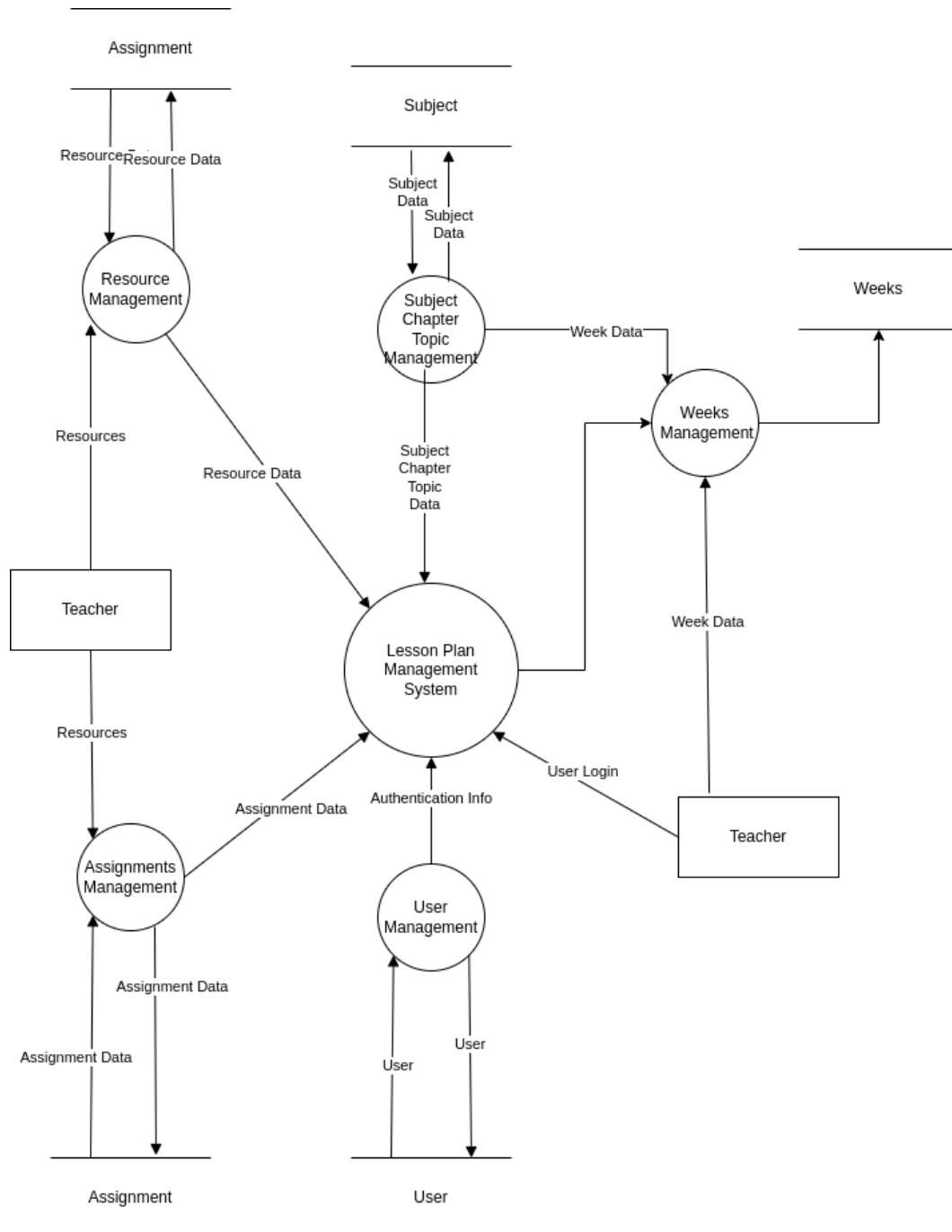
Figure 3: 1-level DFD

4.5 ER Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases.
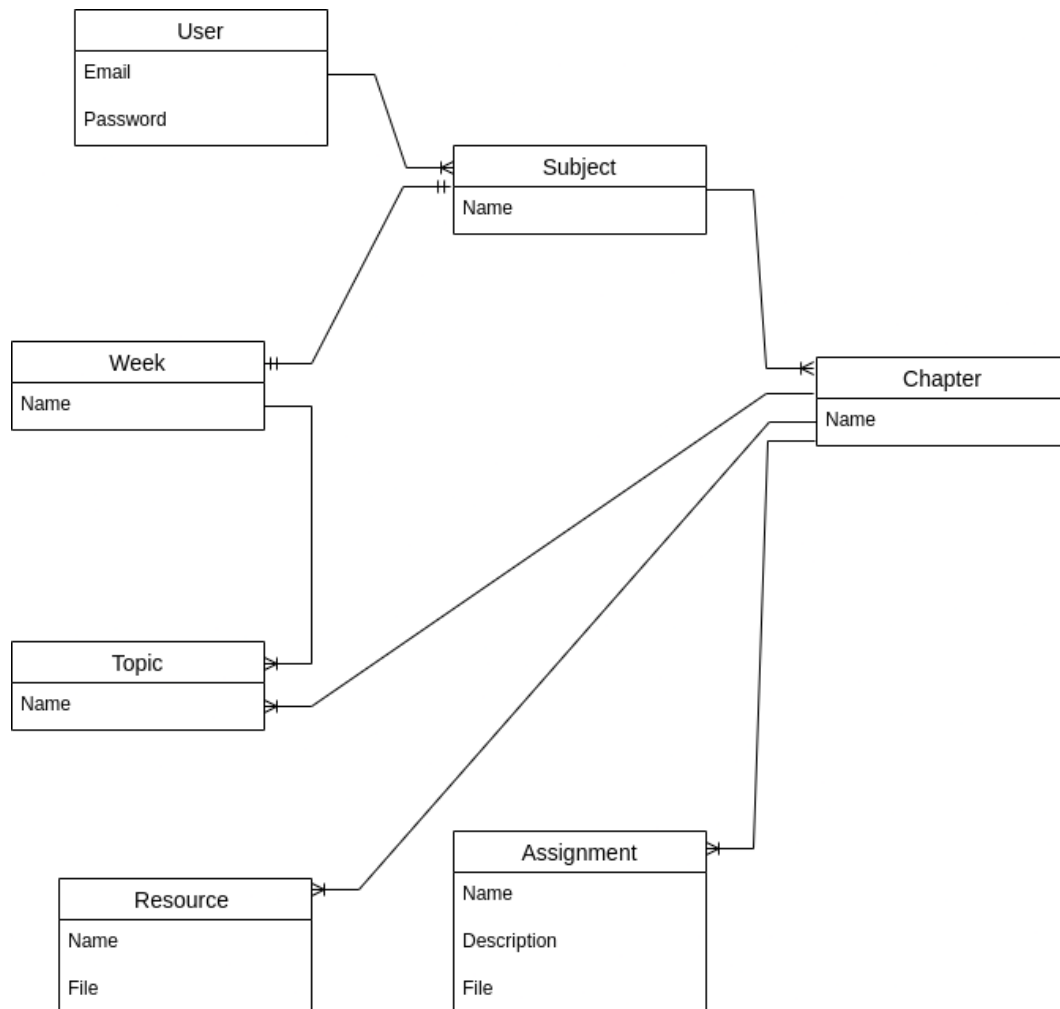
Figure 4: 1-level DFD

5 User Guide

5.1 Setup Guide

Prerequisits:

NodeJS ($>=$14)

npm ($>=$10)

Mongodb (>=4)

Clone the Repository:

```
git clone git@github.com:Aurshal/Lesson-Plan-Management
    ↪ .git
```

Install required npm packages:

```
npm install
```

Start the Project

```
npm start
```

If everything goes well, the application starts running at port 3000.

5.2 Walkthrough

A walkthrough of the Recommendation Letter Generator System:

5.2.1 Login and Register

Here are the login page and register page that is presented to user when they visit the website:
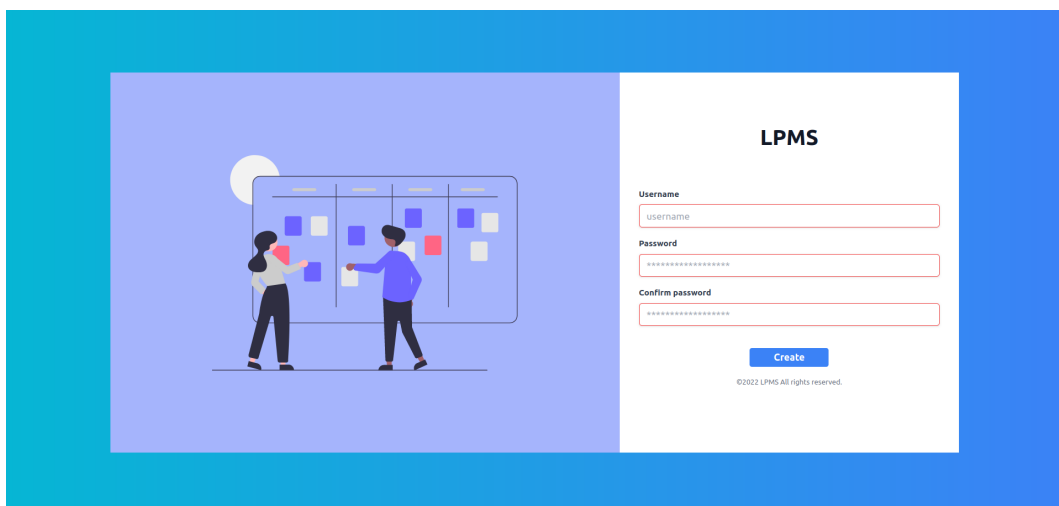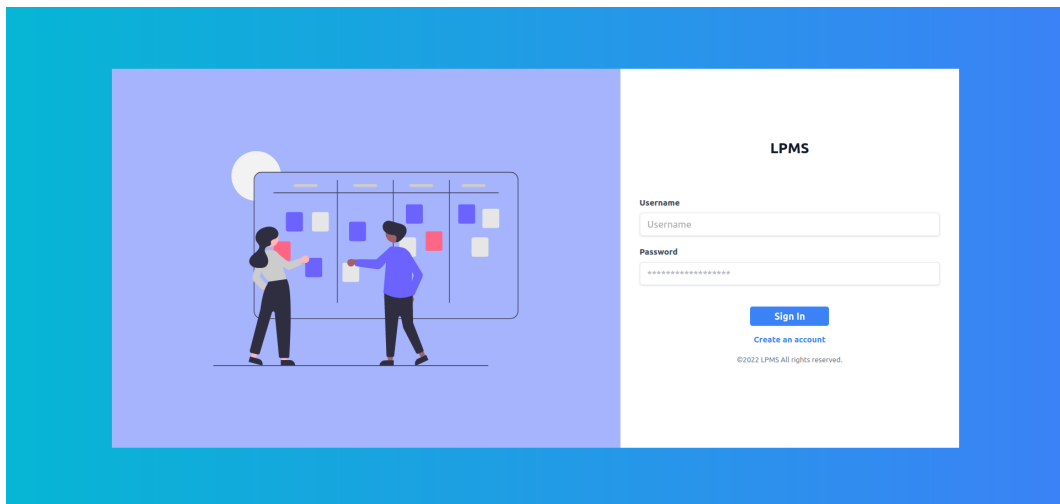
5.2.2 Subject

After Logging in, users get options to add and manage subjects they teach.

5.2.3 Weeks

Under Weekly Plan, users can add/remove weeks. Under each week, there are options to assign/remove chapters/topics to that week.

5.2.4 Chapters

Under Chapters, users get options to add/remove chapters. Similarly, Topics can also be added and removed.

Their associated teaching weeks are also shown alongside (if assigned).

5.2.5 Assignments

Under Assignments, the user gets the option to add assignments for each chapter under that subject. Each assignment has a "Title", a "Description" and a "File" field associated with it.

5.2.6 Resources

Under Resources, the user gets the option to add assignments for each chapter under that subject. Each assignment has a "Title", and a "File" field associated with it.

Users can also delete any previously created resources for any chapter.

6   Conclusion

The Lesson Plan Management System is a web app for teachers that helps in managing the lesson plan and associated resources assignments. Teachers can keep track of their current syllabus status for the subjects and reuse the same lesson plan in different semesters too.

6.1   Future Enhancements

Following features can be added to the software in future under the same agile incremental model.

1. To import and export the lesson plan in different formats

2. To add central management system that can be used by departments to monitor teacher's plans