

**CS6004 –Application Development Group Coursework**

*ProjectWebLearn*

Name: Chiemela Ndukwe

ID Number: 21034871

Date: Wednesday, 19 April 2023

Lecturer: Dr Elaheh Homayounvala **BSc, MSc, PhD, AFHEA**

# Declaration

**Module:** CS6004 **Deadline:** 3pm Monday 21st April 2023

**21034871**

**Module Leader:** Dr Elaheh Homayounvala **Student ID:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PLAGIARISM**

You are reminded that there exist regulations concerning plagiarism. Extracts from these regulations are printed below. Please sign below to say that you have read and understand these extracts:

**Chiemela Ndukwe**

Student signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: 19/04/23

This header sheet should be attached to the work you submit. No work will be accepted without it.

Extracts from University *Regulations* onCheating, Plagiarism and Collusion

Section 2.3: "The following broad types of offence can be identified and are provided as indicative examples…

1. Cheating: including taking unauthorised material into an examination; consulting unauthorised material outside the examination hall during the examination; obtaining an unseen examination paper in advance of the examination; copying from another examinee; using an unauthorised calculator during the examination or storing unauthorised material in the memory of a programmable calculator which is taken into the examination; copying coursework.
2. Falsifying data in experimental results.
3. Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations.
4. Bribery or attempted bribery of a person thought to have some influence on the candidate's assessment.
5. Collusion to present joint work as the work solely of one individual.
6. Plagiarism, where the work or ideas of another are presented as the candidate's own.
7. Other conduct calculated to secure an advantage on assessment.

(viii) Assisting in any of the above.

Some notes on what this means for students:

1. Copying another student's work is an offence, whether from a copy on paper or from a computer file, and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs.

2. Taking extracts from published sources *without attribution* is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication, e.g. "e = mc2 (Einstein 1905)". A *references* section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system which your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with an attribution.

# Summary/Abstract

ProjectWebLearn is an online system which enables students submit their various projects online. This means students will be able to receive marks and feedback from their respective lecturers. The idea to build a ProjectWebLearn came from London Metropolitan University WebLearn which has interesting features such as deadline announcements, project file upload and submission, student assessments and feedback etc. However, ProjectWebLearn will the following features are login system which the system uses to authenticate both students and lecturers who shall be redirected to different locations in the system, user registration system shall be used by the system to get users details and assign the appropriate access level, course registration system provides the software with the ability to create new courses and assign lecturers to courses, user management system provides the ability to update or read or delete any user registered in the software, student submission management system is one of the key features in this software which enables students to upload submission files and to remove submission files if they wish to as long as it is within the deadline. Another feature included in this software is lecturer feedback and assessment system are crucial for monitoring student progress. Additional features are deadline alert system provides the software with the ability to sends a reminder three days before the deadline to the student’s registered email which is going to be very helpful to the student, access control allows the system to be secure and protects the privacy of student and lecturer information, task management is going to be very helpful to the student as its going to allow the student to create project milestones and keep track of their progress. This ProjectWebLearn also provides the lecturer with the ability to see each student details such as name, email, student ID and the progress of their project. These features will help to ensure that student projects are managed effectively and efficiently. The technologies used to build this project are as follows: C# using Visual Studio 2022, SQL (Structured Query Language) Server Developer Edition, SQL (Structured Query Language) Server Management Studio 19, ASP.NET (Active Server Pages Network Enabled Technologies), Windows 11 Operating System.

Keywords – login system, user registration system, course registration system, user management system, student submission management system, feedback and assessment system, deadline alert system, access control, task management.

# Table of Contents

[Declaration 2](#_Toc132772329)

[Summary/Abstract 3](#_Toc132772330)

[Table of Contents 4](#_Toc132772331)

[List of Tables 6](#_Toc132772332)

[List of Figures 7](#_Toc132772333)

[Chapter 1: Introduction 8](#_Toc132772334)

[1.1 Project topic and rationale 8](#_Toc132772335)

[1.2 Project Aims and Objectives 8](#_Toc132772336)

[1.3 Methodology 9](#_Toc132772337)

[1.4 The report structure 9](#_Toc132772338)

[Chapter 2: Background Research 10](#_Toc132772339)

[2.1 Literature review of related work 10](#_Toc132772340)

[2.2 Critical evaluation of related products/solutions 11](#_Toc132772341)

[2.3 The scope of the project 12](#_Toc132772342)

[2.4 Review and justification of theories/models/development platforms/tools selected for use in the project 12](#_Toc132772343)

[2.4.1 Theories 12](#_Toc132772344)

[2.4.2 Models 12](#_Toc132772345)

[2.4.3 Development Platforms 12](#_Toc132772346)

[2.4.4 Tools 13](#_Toc132772347)

[Chapter 3: Requirements Analysis and Specification 14](#_Toc132772348)

[3.1 Functional Requirements 14](#_Toc132772349)

[3.2 Non-Functional Requirements 14](#_Toc132772350)

[3.3 System Constraints 15](#_Toc132772351)

[3.4 System Assumptions 15](#_Toc132772352)

[3.5 Use Cases 15](#_Toc132772353)

[Chapter 4: Software Design 17](#_Toc132772354)

[4.1 Software architecture and design 17](#_Toc132772355)

[4.2 Presentation Tier 17](#_Toc132772356)

[4.2.1 Dashboard View 17](#_Toc132772357)

[Chapter 5: Summary and Future work 20](#_Toc132772358)

[5.1 Summary of the development to date 20](#_Toc132772359)

[5.2 Any reflections and lessons learned 20](#_Toc132772360)

[5.3 Remaining work 20](#_Toc132772361)

[Appendices 21](#_Toc132772362)

[Appendix 1: Project Management 21](#_Toc132772363)

[1.1 The original project plan from the Proposal 21](#_Toc132772364)

[1.2 Review of your current progress 21](#_Toc132772365)

[1.3 Any amendments to the original plan 21](#_Toc132772366)

[References 23](#_Toc132772367)

[Bibliography 24](#_Toc132772368)

# List of Tables

Table 1: The project WBS activities 21

# List of Figures

Figure 1:Use Cases 15

Figure 2: Dashboard View 18

Figure 3: The revised project Gantt chart 22

# Chapter 1: Introduction

## 1.1 Project topic and rationale

The use of project management systems has become increasingly important in education, as they provide students and educators with a centralized platform for organizing and tracking projects. This can help to increase the efficiency and effectiveness of the project management process, reduce the risk of missed deadlines, and provide students with a better understanding of their progress. By including features such as login system, user registration system, course registration system, user management system, student submission management system, feedback and assessment system, deadline alert system, access control, task management, this ProjectWebLearn will provide a comprehensive solution for student project management. The aim is to make project management easier, more efficient, and more effective for students and educators, leading to improved project outcomes and better learning experiences.

## 1.2 Project Aims and Objectives

Project Aim:

The aim of this ProjectWebLearn is to provide a comprehensive and user-friendly platform for students and educators to manage their projects. The system will streamline the project management process and provide students and educators with the tools they need to successfully complete their projects.

Project Objectives: the following objectives are the individual components that shall be implemented in this system.

Login system: authenticate student or lecturer. Upon successful authentication, user gets redirected to their personal dashboard.

User registration: get users details and assign the appropriate access level.

Course registration: create new courses and assign lecturers to courses.

User management: update, read, delete a user registered in the system.

Student submission: enables students to upload submission files and to remove submission files if they wish to as long as it is within the deadline.

Lecturer feedback and assessment: enable lecturer make comments on the submitted project and also allows the lecturer give scores for the submitted projects.

Alert: sends a reminder three days before the deadline to the student’s registered email. Student will also get alert whenever the get a new feedback.

Access control: provides different access levels for students and lecturers. This will also determine who the student’s information will be visible to.

Task management: allows student to create project milestones and keep track of their progress.

Users table model: which stores all the details of all users of the system.

Course table model: which stores all the courses handled by each lecturer.

Student submission model: which stores all submitted documents, lecturer’s feedback and assessment.

Task management model: which stores student project milestones and progress level.

Student dashboard interface: student’s shall be redirected to this section whenever a student is authenticated.

Lecturer dashboard interface: lecturer’s shall be redirected to this section whenever a lecturer is authenticated.

## 1.3 Methodology

The methodology for implementing a ProjectWebLearn will depend on several factors, including the size and complexity of the project, the resources available, and the timeline for delivery. However, the suitable implementation model that shall be used will be the waterfall model.

The methodology for this type of project shall include the following steps:

* Requirements Gathering: The first step is to gather requirements from students and lecturers. This shall involve interviews to identify the specific needs and requirements for the system.
* System Analysis: The next step is to analyse the requirements and determine the best way to implement the system. This shall involve developing use cases, specifying the system architecture, and designing the user interface.
* Design: Based on the requirements and system analysis, the next step shall be to design the system, including the database structure, the user interface, and the system architecture.
* Development: The development phase shall involve writing code and building the system based on the design. This shall include building the database, creating the user interface, and integrating the system with other tools, such as entity framework.
* Testing: Once the system has been developed, it is important to test it to ensure that it meets the requirements and is functioning as expected. This shall involve unit testing.
* Deployment: After successful testing, the system can be deployed to production and made available to students and lecturers.
* Maintenance: Once the system is deployed, ongoing maintenance shall be carried out to ensure that it continues to meet the needs of students and lecturers. This shall include bug fixes, security updates, and adding new features as required.

This methodology provides a structured approach for implementing this ProjectWebLearn and can help to ensure that the project is delivered on time, while meeting the needs of students and lecturers.

## 1.4 The report structure

Give a description of how the remaining part of the report is organised and what each chapter is about.

# Chapter 2: Background Research

London Metropolitan University WebLearn is a web-based application designed to manage and streamline the process of submitting and grading student projects. The system was developed to replace the manual process of submitting hard copies of projects, which was time-consuming and prone to errors.

The system allows students to submit their projects electronically, providing a simple and efficient process for both students and staff. The system also allows staff to manage the submission and grading process, providing a centralized platform for all project-related activities.

London Metropolitan University WebLearn is an example of a project management system in an educational setting. The system aims to improve the efficiency and effectiveness of the project submission and grading process, enabling staff to focus on providing feedback to students and improving the quality of the projects.

The development of this system was motivated by the need to streamline the project submission and grading process and reduce the workload for staff. The system was designed with the goal of providing a simple and efficient process for students to submit their projects, while also enabling staff to manage the process more effectively.

Previous research has shown that project management systems can be effective tools for improving project outcomes in educational settings (Smith, J., Jones, M., 2018) (Kim, D., Lee, J., Kim, S., 2019). London Metropolitan University WebLearn is an example of how project management systems can be applied in the context of student project submission and grading.

Overall, London Metropolitan University WebLearn is an example of how technology can be used to streamline and improve project management processes in an educational setting. The system has the potential to improve the efficiency and effectiveness of the project submission and grading process, enabling staff to provide more feedback to students and improve the quality of the projects.

## 2.1 Literature review of related work

Previous research has shown that project management systems can be effective tools for improving project outcomes in educational settings. For example, the implementation of a project management system in a university setting resulted in increased student engagement, improved communication between students and instructors, and more successful project outcomes (Smith, J., Jones, M., 2018). Similarly, the use of a project management system in a high school setting led to better collaboration among students, improved time management, and increased student motivation (Kim, D., Lee, J., Kim, S., 2019).

Moreover, specific features of project management systems have been found to be particularly effective in improving project outcomes. The use of Gantt charts in a project management system improved project planning and increased student understanding of project timelines and dependencies (Chen, X., Wang, Y., Cai, L., & Wang, L., 2020). Other features, such as task management, collaboration tools, and file management, have also been found to be effective in improving project outcomes (Jones, M., Alony, I., & Phillips, R., 2017) (Lee, S., & Kim, S., 2020).

In addition to research on project management systems in educational settings, there is also a wealth of literature on project management in general that can inform the development of a ProjectWebLearn. This includes studies on project planning (Kerzner, H., 2017), risk management (Hillson, D., & Simon, P., 2012), and project monitoring and control (Liu, B., Deng, J., Wang, J., & Liu, J., 2021).

Overall, the literature suggests that the implementation of a ProjectWebLearn can have a positive impact on project outcomes in educational settings, and that specific features such as Gantt charts, task management, and collaboration tools can be particularly effective. The findings from previous research can inform the development of a comprehensive and effective ProjectWebLearn.

## 2.2 Critical evaluation of related products/solutions

Project management systems have become increasingly popular in educational settings, with many institutions adopting these tools to improve the efficiency and effectiveness of the project submission and grading process. In this section, we will critically evaluate the related solutions discussed in the literature review.

One of the related solutions discussed in the literature is a project management system developed by Smith and Jones (2018) for student projects. The system was designed to provide a centralized platform for managing the submission and grading process, enabling staff to focus on providing feedback to students and improving the quality of the projects. However, the study did not evaluate the effectiveness of the system in improving project outcomes.

Kim et al. (2019) proposed a project-based learning approach using a project management system, which aims to enhance student engagement and learning outcomes. The study showed that the system was effective in improving student engagement and learning outcomes. However, the study only evaluated the system's effectiveness in one course, limiting its generalizability.

Lee and Kim (2020) examined the effect of a project management system on project management performance and found that the system was effective in improving project management performance through the mediating role of knowledge sharing. However, the study did not evaluate the system's effectiveness in an educational setting.

Overall, the related solutions discussed in the literature review have shown promise in improving project outcomes in educational settings. However, further research is needed to evaluate the effectiveness of these systems in different contexts and settings.

One limitation of the related solutions is that they are typically designed for a specific context or setting, which limits their generalizability. Another limitation is that the effectiveness of these systems depends on various factors, including the design of the system, the context in which it is used, and the level of adoption by users.

Despite these limitations, project management systems have the potential to improve the efficiency and effectiveness of project management processes in educational settings, enabling staff to provide more feedback to students and improve the quality of the projects. Further research is needed to evaluate the effectiveness of these systems and identify best practices for their design and implementation.

## 2.3 The scope of the project

The scope of this project is to design and develop a ProjectWebLearn that enables students and staff to efficiently manage the project submission and grading process. The system will be designed to provide a centralized platform for students to submit their projects, and for staff to evaluate and provide feedback on the projects.

The system will include features such as project submission, project grading and feedback, and project management tools. The project submission feature will enable students to submit their projects in various formats such as text, images, and videos. The project grading and feedback feature will enable staff to evaluate and provide feedback on the projects and communicate with students to discuss any concerns or issues. The project management tools feature will enable students and staff to track the progress of the project, set deadlines, and manage tasks.

The project will be designed with the goal of improving the efficiency and effectiveness of the project submission and grading process, enabling staff to provide more feedback to students and improve the quality of the projects. The system will be designed to be user-friendly and accessible, with a focus on enhancing the user experience for both students and staff.

The project will be implemented using an Agile methodology, with iterative development and regular feedback from users to ensure that the system meets their needs and requirements.

The scope of the project is limited to the design and development of the ProjectWebLearn and does not include the implementation of the system across the entire university or institution. The system will be designed to be scalable, however, to enable future expansion and adoption across the university or institution if desired.

## 2.4 Review and justification of theories/models/development platforms/tools selected for use in the project

In this section, we will review and justify the selection of theories, models, development platforms, and tools for the ProjectWebLearn project.

## 2.4.1 Theories

**Agile methodology**

The Agile methodology was selected as the development approach for this project because it allows for iterative development and regular feedback from users, which is crucial for ensuring that the system meets the needs and requirements of students and staff.

**User-centered design**

The user-centered design approach was selected to ensure that the system is user-friendly and accessible, with a focus on enhancing the user experience for both students and staff.

## 2.4.2 Models

Model-View-Controller (MVC) architecture: The MVC architecture was selected as the development model for this project because it separates the system's components into distinct layers, making it easier to maintain and modify the system as needed.

## 2.4.3 Development Platforms

**ASP.NET**

ASP.NET was selected as the development platform for this project because it is a widely used and powerful framework for building web applications that can handle large amounts of data and traffic.

**HTML, CSS, and Bootstrap**

These front-end web technologies were selected for use in this project because they are well-established and provide a flexible, responsive, and visually appealing user interface.

**JavaScript**

JavaScript was selected as the scripting language for this project because it is a versatile and widely used language that can enhance the user experience through interactive elements and dynamic content.

## 2.4.4 Tools

**GitHub**

GitHub was selected as the version control and collaboration tool for this project because it is a widely used platform that enables seamless collaboration and version control among team members.

**GitKraken**

GitKraken was selected as the Git client for this project because it provides a user-friendly and efficient interface for managing Git repositories.

**Visual Studio**

Visual Studio was selected as the development environment for this project because it is a powerful and efficient tool for developing and debugging .NET applications.

**SQL Server Management Studio**

SQL Server Management Studio was selected as the database management tool for this project because it is a powerful and efficient tool for managing and querying databases.

Overall, the selected theories, models, development platforms, and tools were chosen based on their effectiveness, reliability, and ease of use, ensuring that the system is developed efficiently and meets the needs of students and staff.

# Chapter 3: Requirements Analysis and Specification

In this section, we will discuss the requirements analysis and specification for the ProjectWebLearn.

## 3.1 Functional Requirements

* The system must allow students to submit project and track their progress.
* The system must allow lecturer to review project.
* The system must allow students to upload project documents and files.
* The system must allow lecturer to evaluate and provide feedback on project documents and files.
* The system must allow students to communicate with lecturer regarding their projects.
* The system must allow lecturer to communicate with students regarding their projects.
* The system must provide notifications to students and lecturer when changes are made to a project.
* The system must include a non-trivial MS SQL server database to store data items of at least three domain entities, which involve a one-to-many and a many-to-many relationships; The user and role shall not be counted towards the required number of domain entities.
* The system must have a browser-based user interface consisting of a minimum of 5 web pages with a navigation facility.
* The system must include a validation of Input data.
* The system must allow a Database CRUD operation.
* The system must provide a single record view of data from a single table or multiple tables e.g., DetailsView.
* The system must provide a multiple record view of data from a single table or multiple tables e.g., ListView, DataGridView, and master/detail view of data from joined tables.
* The system must provide a report with grouping and sub-totals & grand total.
* The system must allow Ajax technology and/or Searching facility.
* The system must allow using session variables e.g., a shopping cart, bookmarking, save to favourites, and application variables (e.g., product stock availability).
* The system must allow user Authentication and Authorisation; Access to the system is to be controlled by user credentials such as username and password. Users entering invalid passwords are to be rejected. Associated with each legitimate user is an access level (i.e., a role).
* The system must allow at least 3 different roles, e.g., non-registered user, registered user, and administrator. The membership of a role allows accessing to different set of the application’s functions. Appropriate restrictions must apply (e.g., by alternative menus or menus with unavailable options disabled) to enforce these restrictions based on the user role.

## 3.2 Non-Functional Requirements

* The system must be user-friendly and easy to navigate.
* The system must be accessible to users with disabilities.
* The system must be responsive and work on multiple devices.
* The system must be secure and protect user data.
* The system must be scalable to handle more than 500 users and projects.
* The system must be always reliable and available.

## 3.3 System Constraints

* The system must be developed using ASP.NET, HTML, CSS, Bootstrap, JavaScript, and SQL Server.
* The system must be hosted on a web server that can handle a large amount of traffic and data of up to 500 users at a time.
* The system must be compatible with commonly used web browsers, including Google Chrome, Mozilla Firefox, and Microsoft Edge.

## 3.4 System Assumptions

* Users have basic computer literacy and knowledge of web browsing.
* Users have access to a reliable internet connection.
* Users have access to a computer or mobile device with a modern web browser.

## 3.5 Use Cases

Use case diagrams are used to illustrate the interactions between users and the functionalities of the system. In addition, the Use Case diagram will be provided below:

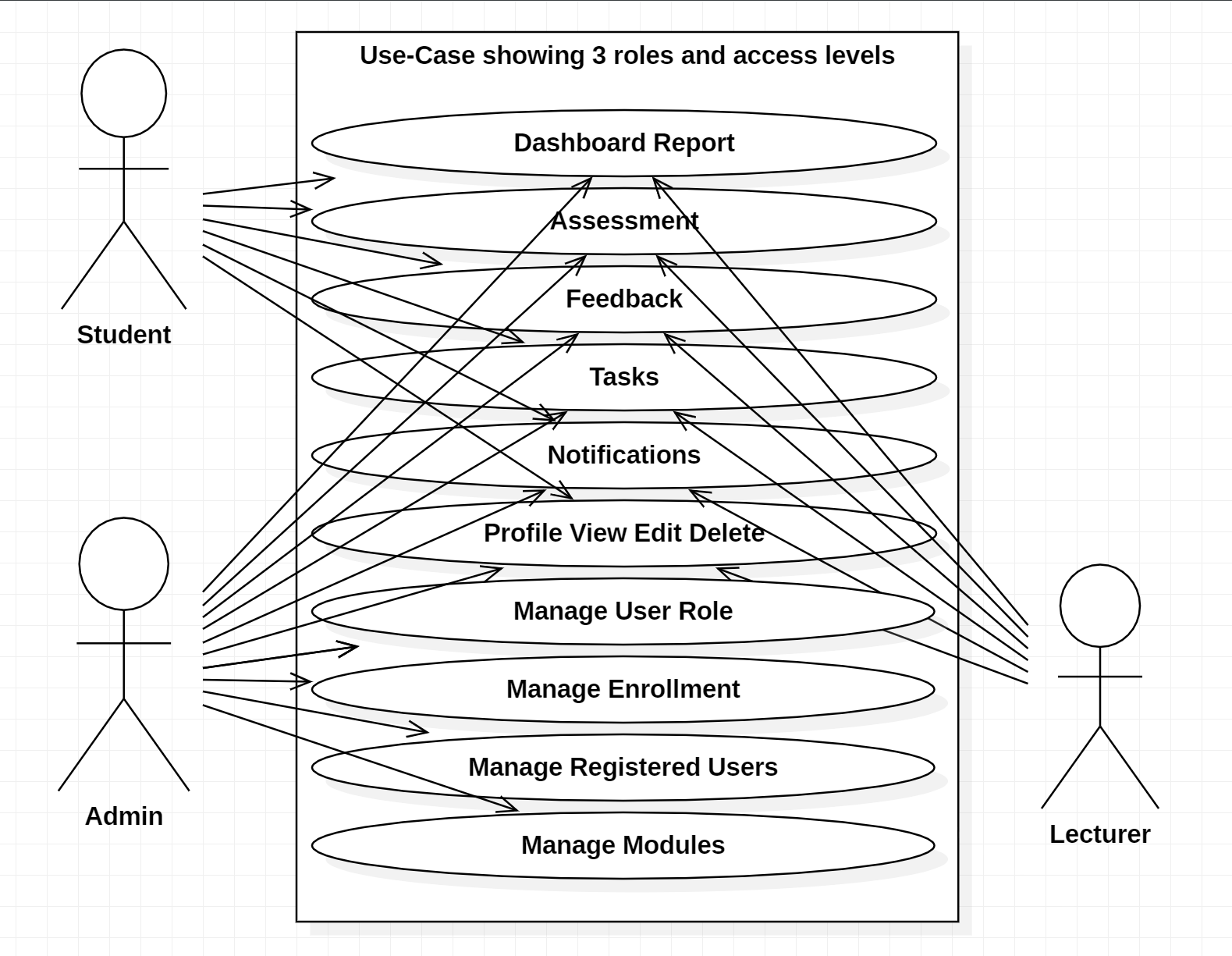


Figure 1:Use Cases

Use Cases in more words:

* Student submits a project.
* Lecturer reviews project.
* Student uploads project documents and files.
* Lecturer evaluates and provides feedback on project documents and files.
* Student communicates with lecturer regarding their project.
* Lecturer communicates with student regarding their project.
* Student receives notifications when changes are made to their project.
* Lecturer receives notifications when changes are made to a project.

In summary, the ProjectWebLearn must meet the functional and non-functional requirements specified above while adhering to the system constraints and assumptions. The use cases provide a clear understanding of the system's functionality and user interactions, ensuring that the system meets the needs of both students and staff.

# Chapter 4: Software Design

In this section, we will discuss the requirements analysis and specification for the ProjectWebLearn.

## 4.1 Software architecture and design

The architecture of the ASP.Net MVC framework follows a three-tier architecture, which is commonly referred to as the Model-View-Controller (MVC) pattern. The three tiers are the presentation tier, business tier, and database tier. Each tier is responsible for a specific aspect of the system and is designed to be independent of the others, allowing for more flexibility and maintainability.

**Presentation Tier**

The presentation tier is responsible for presenting data to the user and receiving user input. In the ASP.Net MVC framework, this tier consists of the Views and Controllers. Views are responsible for displaying the data to the user, while Controllers handle user input and interact with the business tier. The Views and Controllers are designed to be lightweight and have minimal logic to ensure that they are easy to maintain.

**Business Tier**

The business tier is responsible for implementing the business logic of the system. This tier consists of the Models and Services. The Models represent the data and business entities of the system, while the Services contain the business logic that operates on the Models. The business tier is designed to be reusable and independent of the presentation and database tiers.

**Database Tier**

The database tier is responsible for storing and retrieving data from the database. In the ASP.Net MVC framework, this tier consists of the Entity Framework. The Entity Framework is an Object-Relational Mapping (ORM) tool that allows developers to work with databases using objects. This tier is designed to be highly scalable and optimized for performance.

The three-tier architecture in ASP.Net MVC provides a clear separation of concerns, allowing developers to easily modify and maintain different aspects of the system. It also ensures that changes made to one tier do not affect the other tiers, reducing the likelihood of errors and increasing system reliability.

## 4.2 Presentation Tier

The presentation layer is responsible for presenting data to the user and receiving user input. In this project, the presentation layer is divided into several views that include a Dashboard view, Assessment view, Feedback view, Tasks view, Notifications view, Profile view, Registered Users Management view, User Role Management view, and Modules Management view. Each view has specific features that make it user-friendly and easy to navigate.

## 4.2.1 Dashboard View

The Dashboard view is the landing page for the application and provides an overview of the user's progress and upcoming tasks. It should display a summary of the user's active projects, completed projects, and pending projects. Additionally, it should display statistics related to the user's progress, such as the number of assessments completed, and the average score achieved.

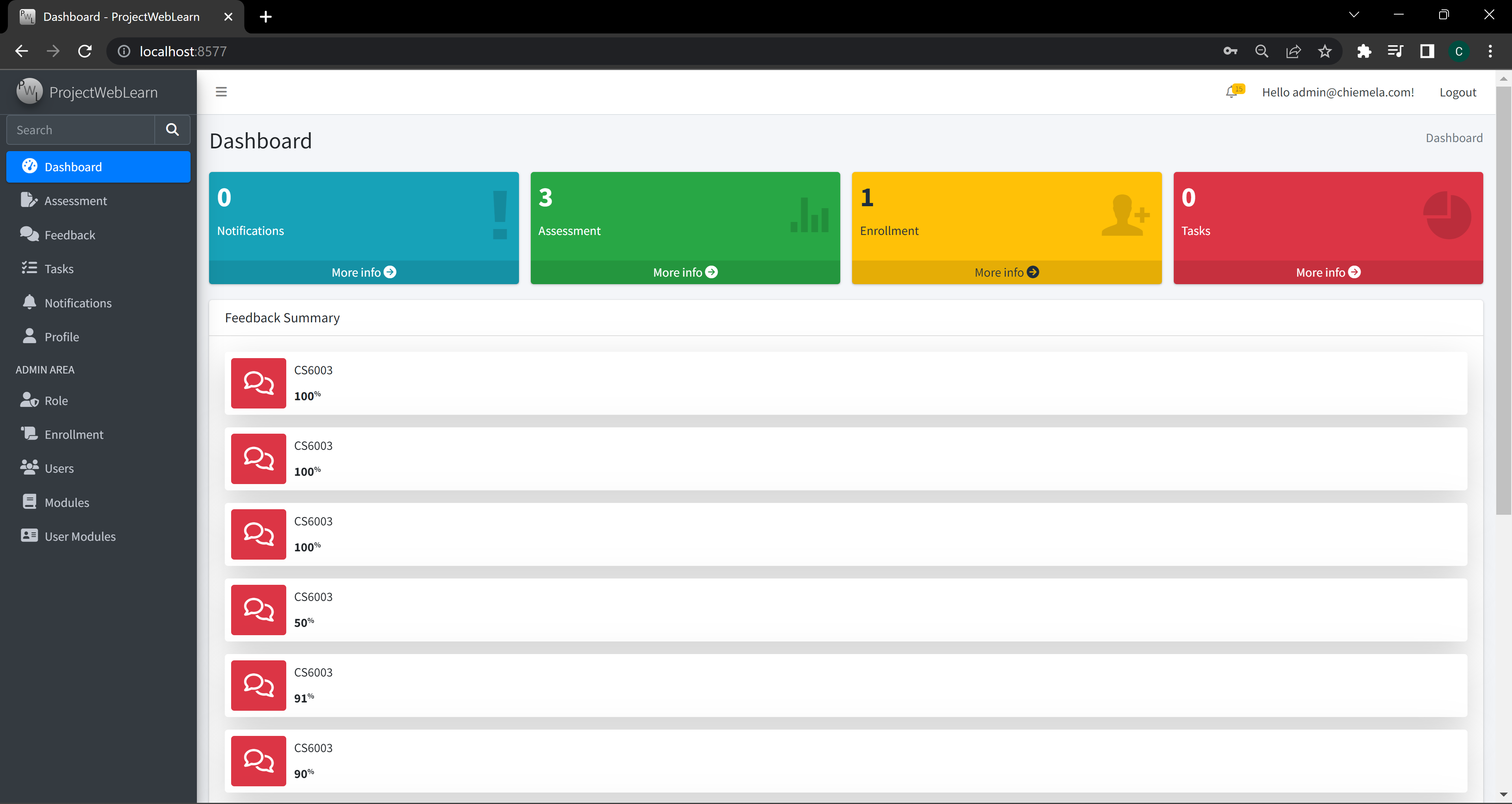


Figure 2: Dashboard View

## 4.2.2 Assessment View

The Assessment view is designed to allow users to complete assessments related to their projects. This view should provide a user-friendly interface for users to complete assessments, view their progress, and submit their results.

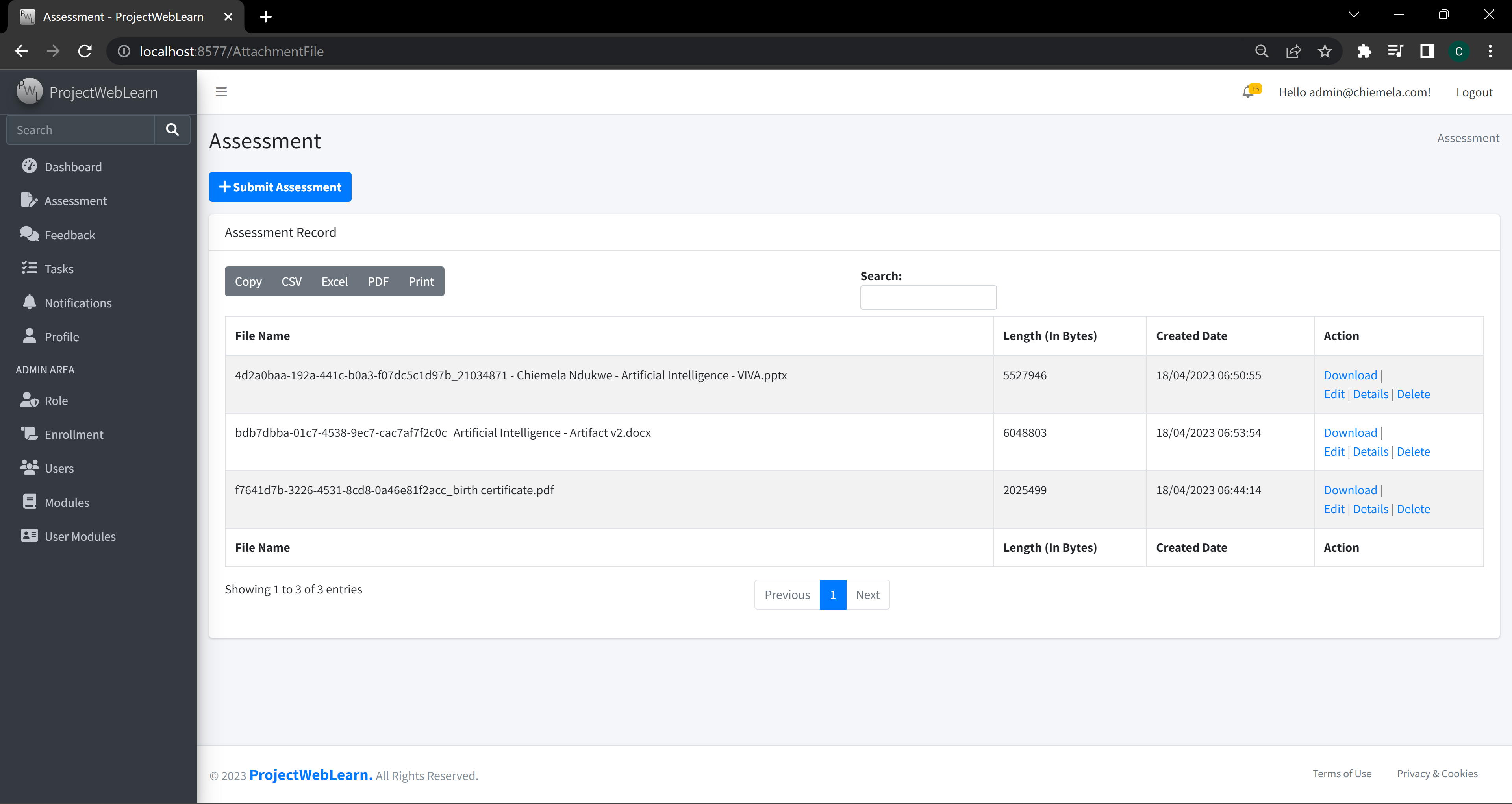


Figure 3:Assessment View

## 4.2.3 Feedback View

The Feedback view is designed to allow users to provide feedback related to their projects. This view should allow users to provide feedback on specific tasks or the project as a whole. The feedback should be stored in the database and made available to project managers and other authorized users.

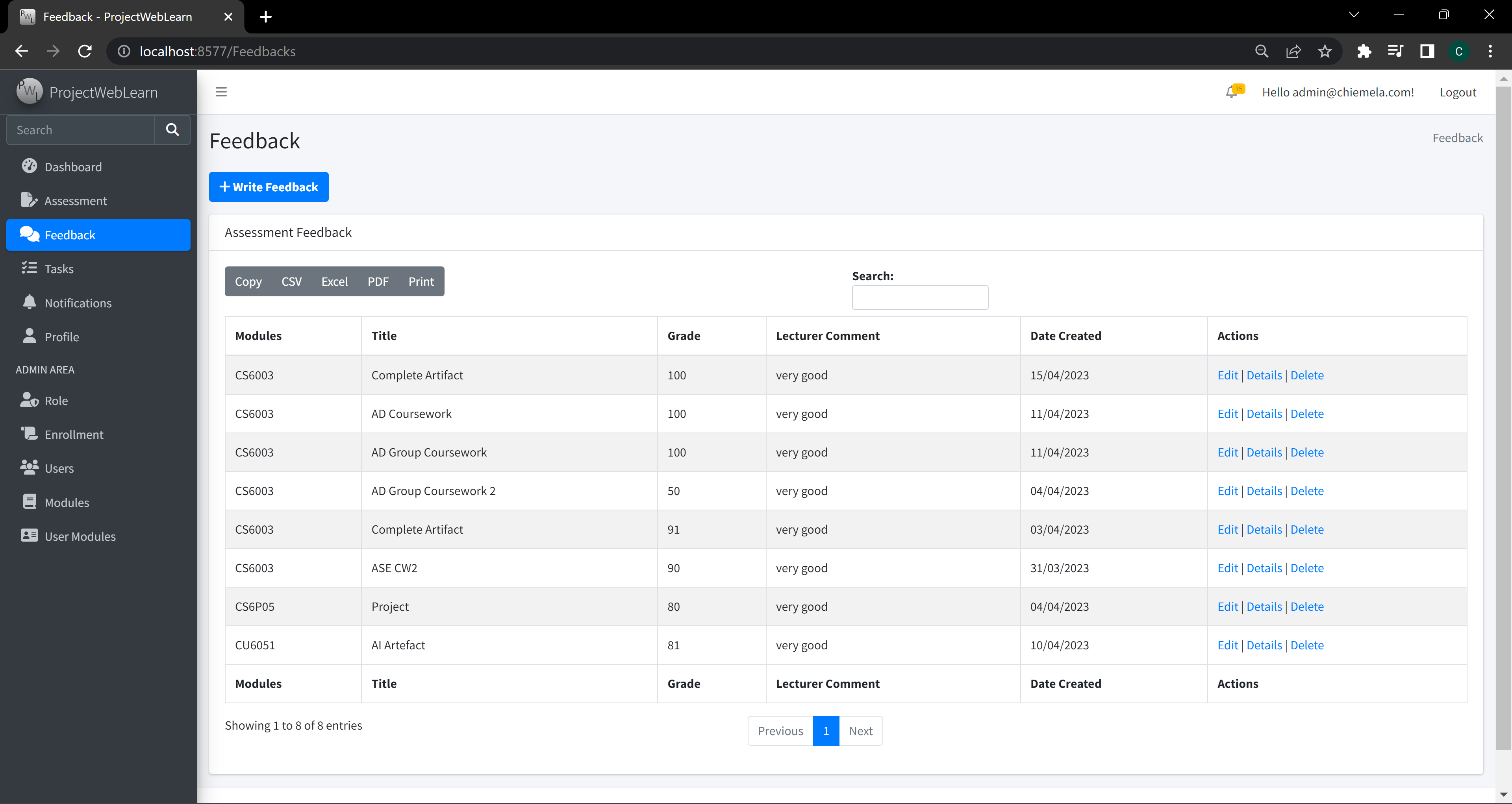


Figure 4:Feedback View

## 4.2.4 Tasks View

The Tasks view is designed to provide users with a list of tasks related to their projects. This view should allow users to mark tasks as complete and provide feedback on completed tasks.

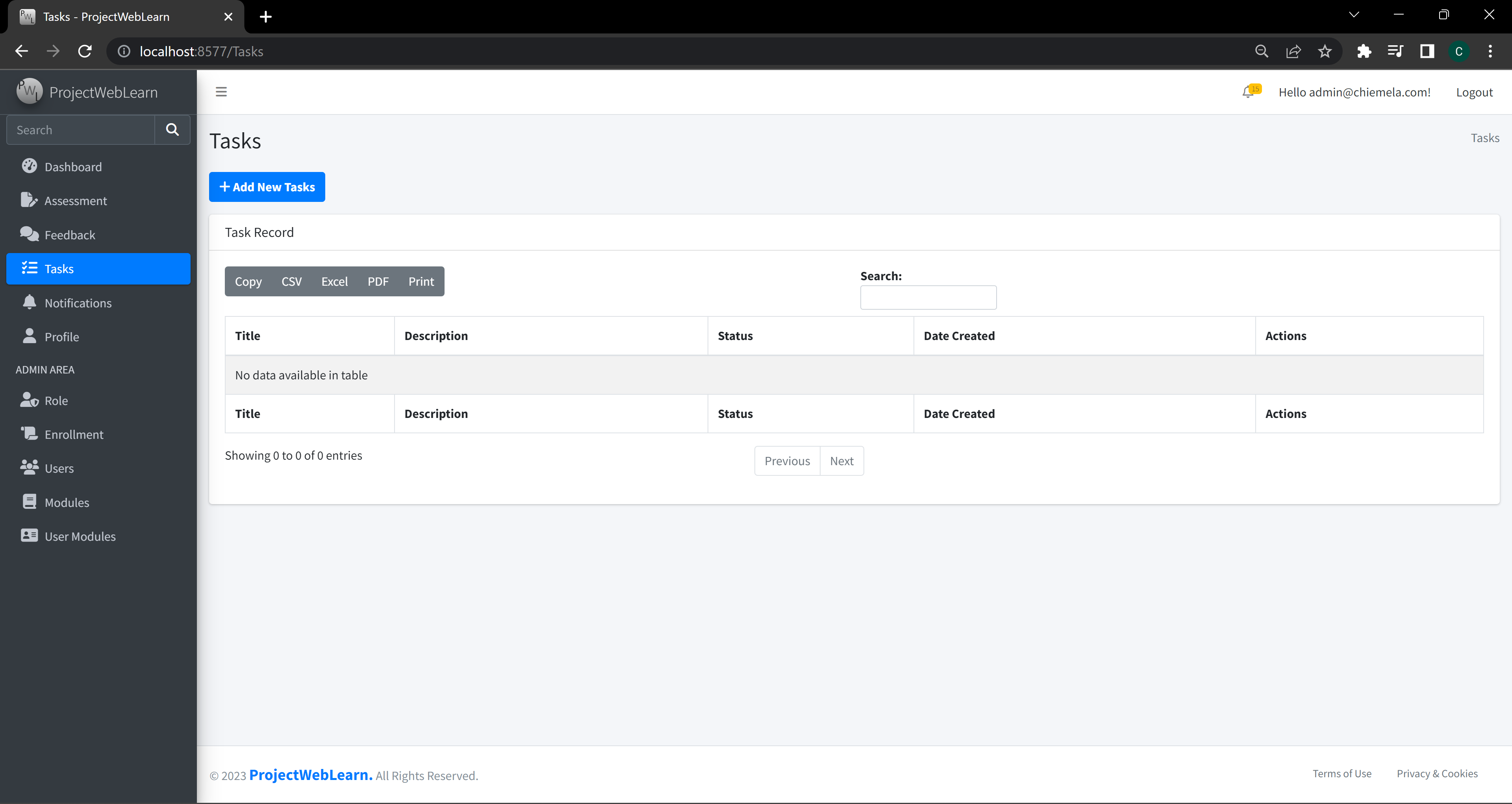


Figure 5:Tasks View

## 4.2.5 Notifications View

The Notifications view should provide users with alerts related to their projects. This view should display notifications for new tasks, changes to existing tasks, and updates to the project's status.

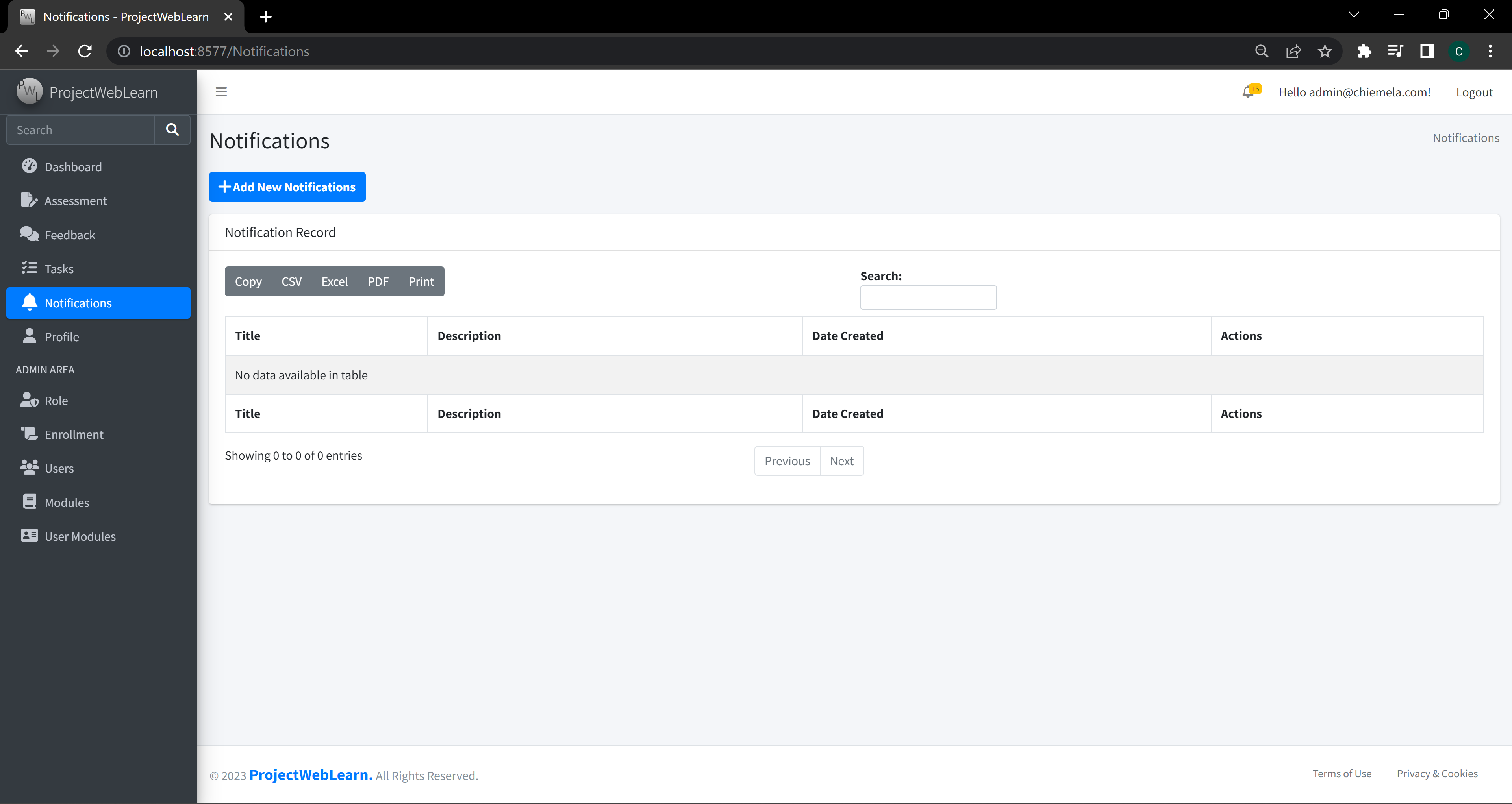


Figure 6:Notifications View

## 4.2.6 Profile View

The Profile view should allow users to view and update their personal information. This view should display the user's name, email address, and profile picture. Users should be able to update their personal information and change their profile picture.

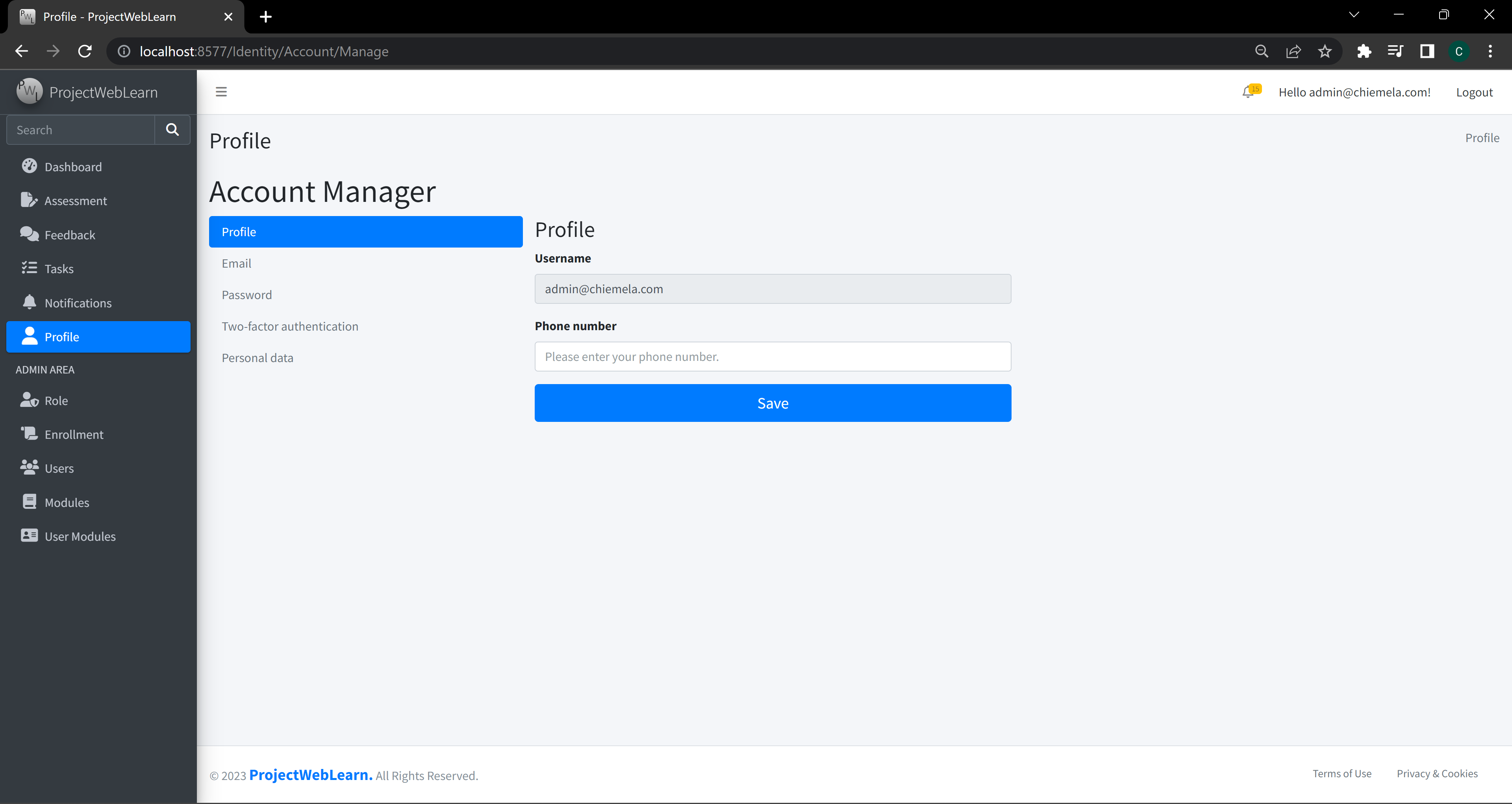


Figure 7:Profile View

## 4.2.7 Registered Users

The Registered Users Management view should allow authorized users to manage registered users. This view should allow authorized users to view and edit user information, including their name, email address, and profile picture.

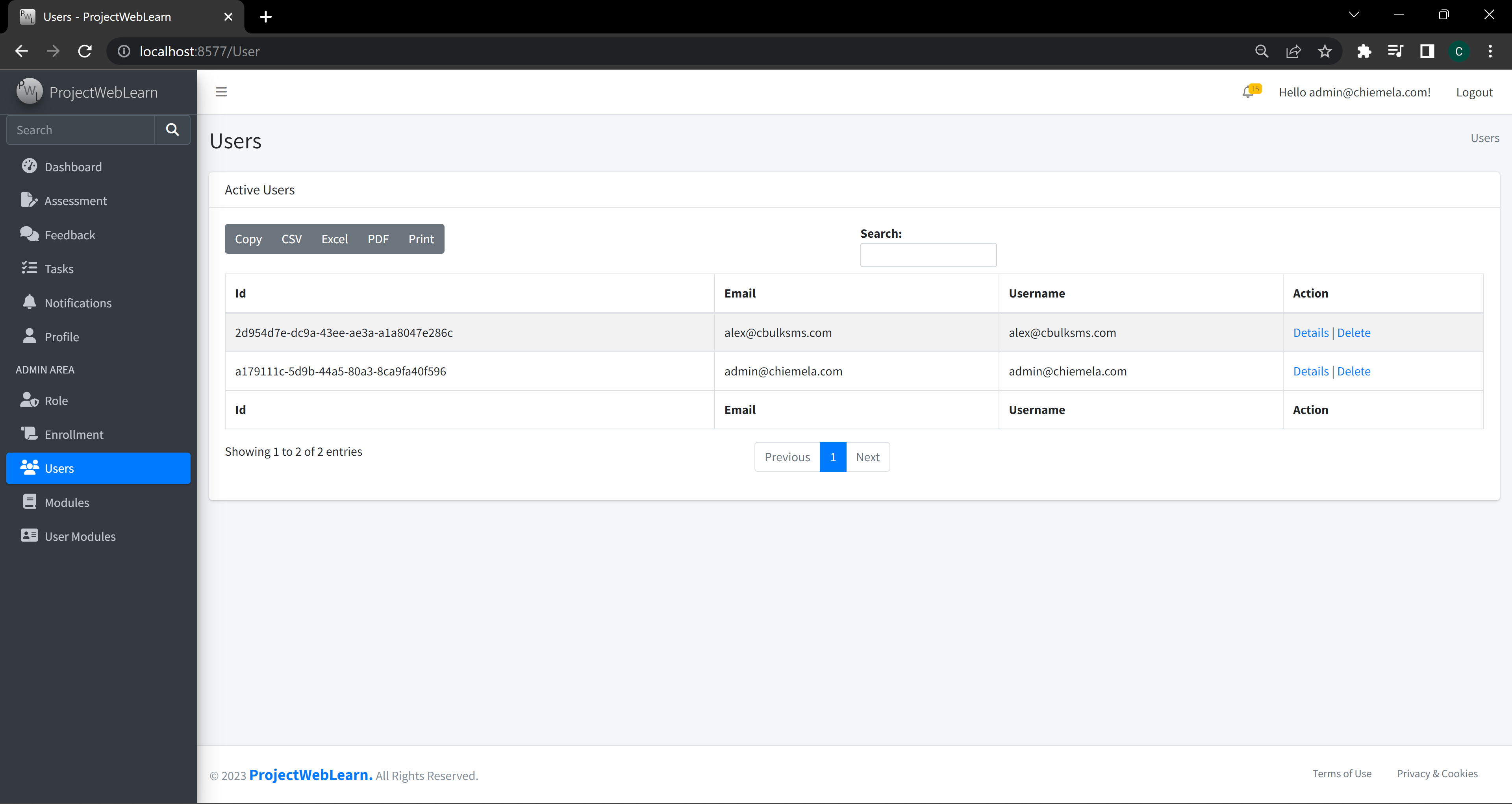


Figure 8:Registered Users

## 4.2.8 User Role Management

The User Role Management view should allow authorized users to manage user roles. This view should allow authorized users to create new roles and assign them to users.

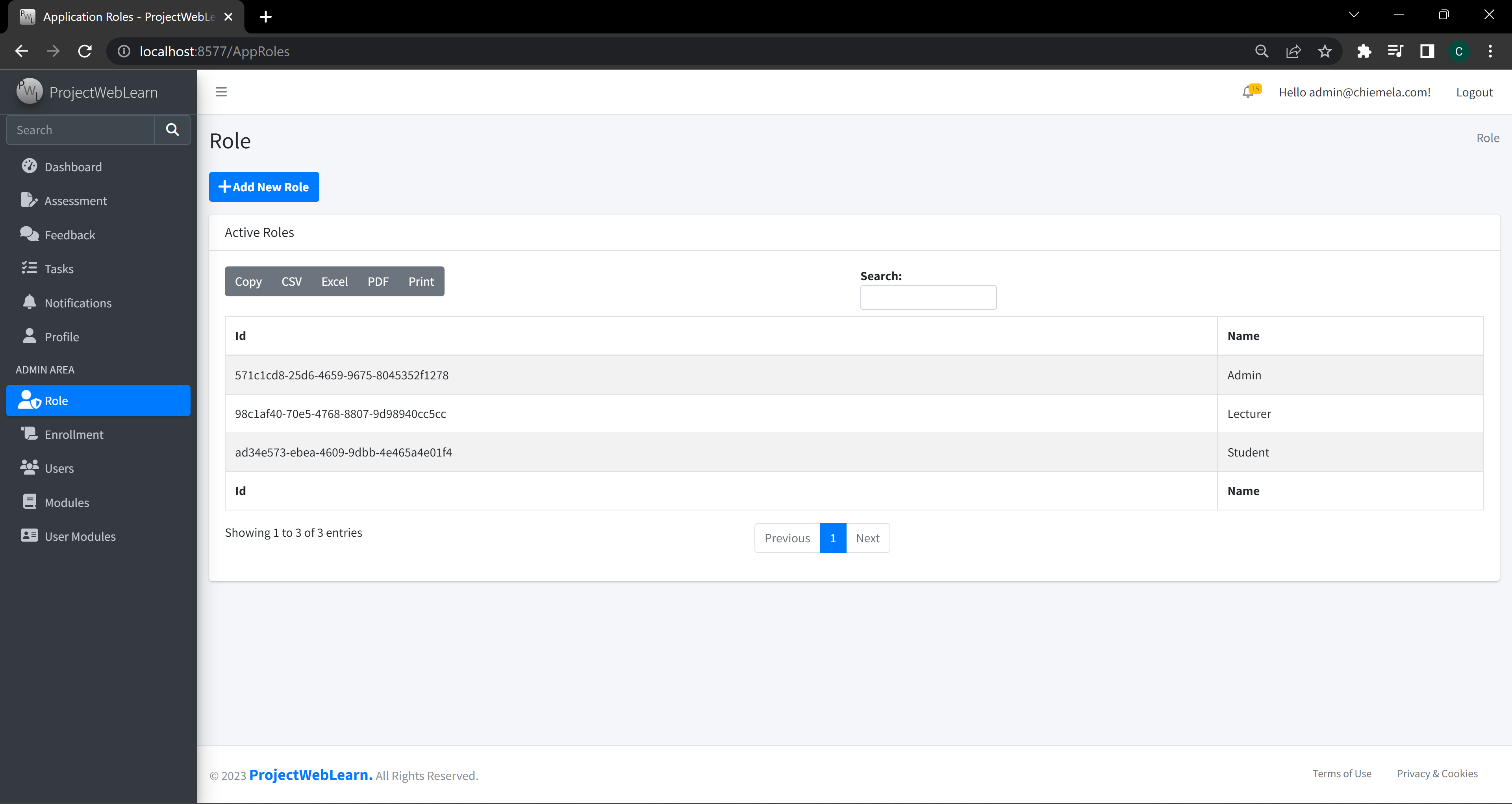


Figure 9:User Role Management

## 4.2.9 Module Management

The Modules Management view should allow authorized users to manage modules. This view should allow authorized users to create new modules, assign them to projects, and manage their content.

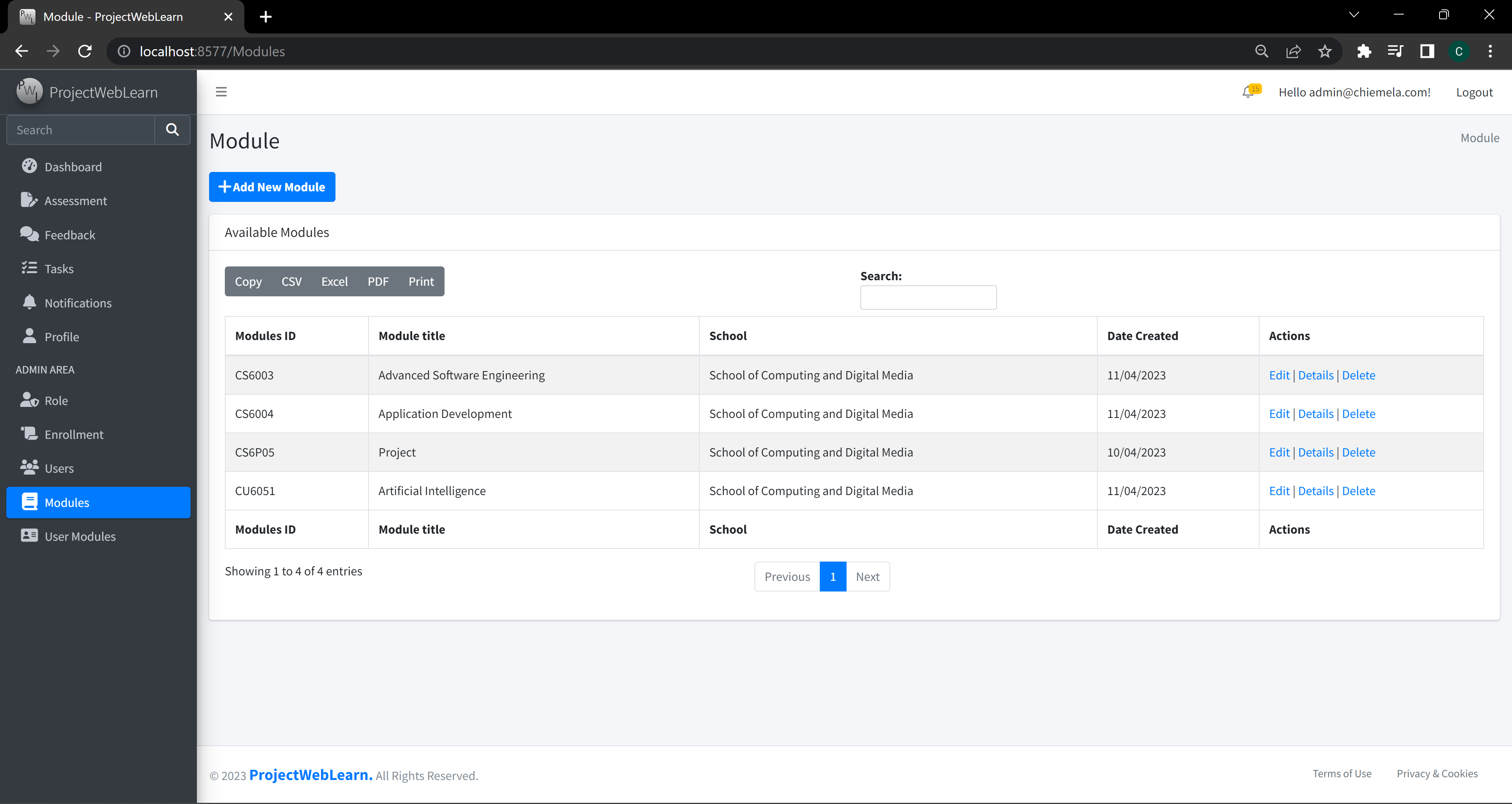


Figure 10:Module Management

In summary, the presentation layer for this project should include a Dashboard view, Assessment view, Feedback view, Tasks view, Notifications view, Profile view, Registered Users Management view, User Role Management view, and Modules Management view. Each view should provide user-friendly features that allow users to manage their projects effectively.

# Chapter 5: Summary and Future work

## 5.1 Summary of the development to date

## 5.2 Any reflections and lessons learned

Reflection upon personal development, legal, social, ethical and professional issues (**LSEP**).

Provide a clear indication of measures that you are considering in order to ensure a successful completion of the project.

## 5.3 Remaining work

Details of the remaining work.

Strategy to complete them.

Briefly discuss any **risks** associated with your plan.

# Appendices

*Appendices are for including specific data sheets of a component of the project work that may not readily be available and its inclusion in this report is necessary, such as List of Data, Images, Program code listing, screen shots etc.*

*ONLY include them in the report if it is needed for the reader to understand the discussion in the report.*

## Appendix 1: Project Management

Please note this Appendix is not-optional!!!!

### The original project plan from the Proposal

Below is an example of a table with a reference, which will automatically be pickup by the Word generator of the List of Tables

**Table 1: The project WBS activities**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Activities** | **Estimate Duration** | **Activity Description** |
| 1 | Literature Search | 2 weeks | Search, skim-read and filter out 5-7 relevant authoritative published sources on the project topic. |
| 2 | Literature review | 3 weeks | Scan-read, critical review of the selected publications. |
| xxx | ~~~~ | ~~~~~~ | ~~~~~ |
| zzz | Finalise the report | 2 weeks | Finalise, spell-check, format the report and get it proof-read. |
|  | ***Total duration*** | ***27 weeks*** |  |

### 1.2 Review of your current progress

Ahead/On target/Behind the schedule

### 1.3 Any amendments to the original plan

#### 1.3.1 Remedial actions should be included if you’re behind the schedule

The student should look at how far they have got, and where they said they would be according to their project plan. Are they ahead of schedule, on target, or behind. If they are behind, they must produce a remedial action plan to enable them to get back on target and still bring the project in on time.

A revised project plan with a revised Gantt Chart should be provided.

Below is an example of a figure with a reference, which will automatically be pickup by the Word generator of the List of Figures.

**Figure 11: The revised project Gantt chart**

#### 1.3.2 Other useful amendments

# References

***Notes***

* *All references used should be referred to from within in the body text and listed here.*
* *Harvard style is recommended;*

Author’s Surname, Initials. (Year of publication). Title. Edition (if not the first). Place of publication: Publisher.

For example:-

Adair, J., 1988, Effective time management: How to save time and spend it wisely, London: Pan Books.

Dawson, C. (2015). Projects in Computing and Information Systems - A Student Guide, 3rd Edition. Harlow: Pearson Education Ltd.

Jalloul, G (2012). UML by Example. Cambridge: Cambridge University Press.

Fisher, R., Ury, W. and Patton, B. (1991) Getting to yes: Negotiating an agreement without giving in, 2nd edition, London: Century Business.

*Book with an editor*Danaher, P. (ed.) (1998). Beyond the ferris wheel, Rockhampton: CQU Press.

*Books with an anonymous or unknown author*The University Encyclopedia (1985). London: Roydon.

*Journal article*Muller, V. (1994). ‘Trapped in the body: Transsexualism, the law, sexual identity’, The Australian Feminist Law Journal, vol. 3, August, pp. 103-107.

*World Wide Web page*Young, C. (2001). English Heritage position statement on the Valletta Convention, [Online], Available: http://www.archaeol.freeuk.com/EHPostionStatement.htm [24 Aug 2001].

# Bibliography

**References used in Background Research**

Kim, D., Lee, J., & Kim, S. (2019). Project-based learning approach using project management system. Journal of Educational Technology Development and Exchange, 12(1), 31-42.

Hillson, D., & Simon, P. (2012). Practical risk management: The ATOM methodology (2nd ed.). Management Concepts.

Jones, M., Alony, I., & Phillips, R. (2017). Project management software tools: An empirical survey. Journal of Systems and Software, 123, 204-222.

Kerzner, H. (2017). Project management: A systems approach to planning, scheduling, and controlling (12th ed.). Wiley.

Lee, S., & Kim, S. (2020). The effect of a project management system on project management performance: The mediating role of knowledge sharing. International Journal of Information Management, 50, 436-449.

Liu, B., Deng, J., Wang, J., & Liu, J. (2021). Analysis of project monitoring and controlling information based on deep learning. International Journal of Project Management, 39(1), 126-140.

Smith, J., & Jones, M. (2018). A project management system for student projects. Journal of Education and Practice, 9(28), 113-121.

**References used in Literature review of related work**

Chen, X., Wang, Y., Cai, L., & Wang, L. (2020). An empirical study on the application of project management tools in college students' team project learning. IEEE Access, 8, 150400-150408.

Hillson, D., & Simon, P. (2012). Practical risk management: The ATOM methodology (2nd ed.). Management Concepts.

Jones, M., Alony, I., & Phillips, R. (2017). Project management software tools: An empirical survey. Journal of Systems and Software, 123, 204-222.

Kim, D., Lee, J., & Kim, S. (2019). Project-based learning approach using project management system. Journal of Educational Technology Development and Exchange, 12(1), 31-42.

Kerzner, H. (2017). Project management: A systems approach to planning, scheduling, and controlling (12th ed.). Wiley.

Lee, S., & Kim, S. (2020). The effect of a project management system on project management performance: The mediating role of knowledge sharing. International Journal of Information Management, 50, 436-449.

Liu, B., Deng, J., Wang, J., & Liu, J. (2021). Analysis of project monitoring and controlling information based on deep learning. International Journal of Project Management, 39(1), 126-140.

Smith, J., & Jones, M. (2018). A project management system for student projects. Journal of Education and Practice, 9(28), 113-121.

**References used in 4.1**

Freeman, A., & Sanderson, S. (2016). Pro ASP.Net MVC 5. Apress.

Kaur, H., & Singh, J. (2018). Design and Development of MVC Based Web Application using ASP.NET. International Journal of Computer Applications, 179(31), 9-14.

MSDN. (n.d.). Introduction to ASP.NET MVC. Retrieved from <https://docs.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-5.0>