

Web Framework Development

Individual Project

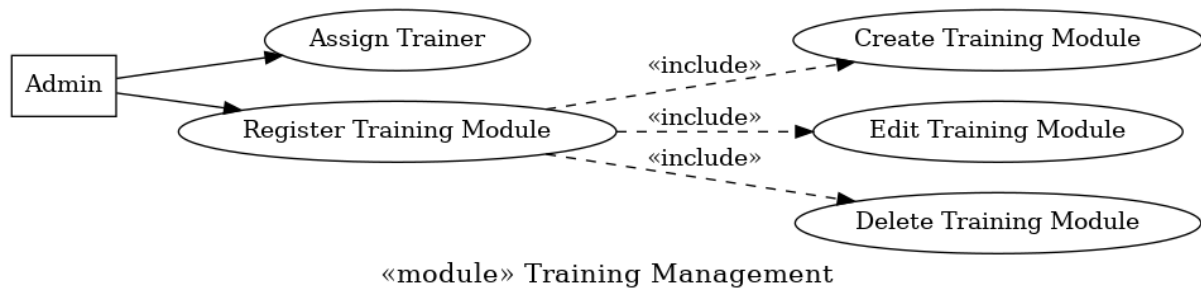
Deliverable 2

Design 1 – Use Cases

B00138007

Burair Moosavi

Admin: Training Management



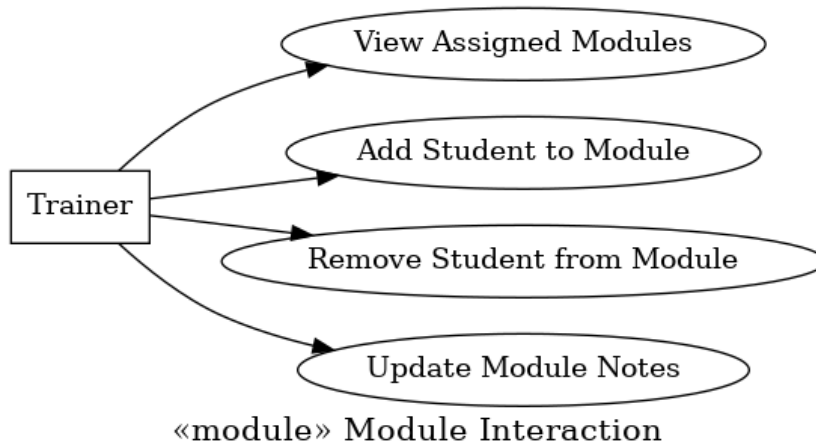
This diagram shows the Admin's main actions when managing training modules in the system

The Admin is responsible for managing the training modules within the system. The main action is to register a training module, which includes the ability to:

- Create a new training module
- Edit an existing training module
- Delete a training module

In addition to module management, the Admin will be able to assign trainers to each module. The «include» arrows show that creating, editing, and deleting a module are all part of registering it.

Trainer – Manage Enrolments



This diagram shows the Trainer's use cases for managing module enrollments and content

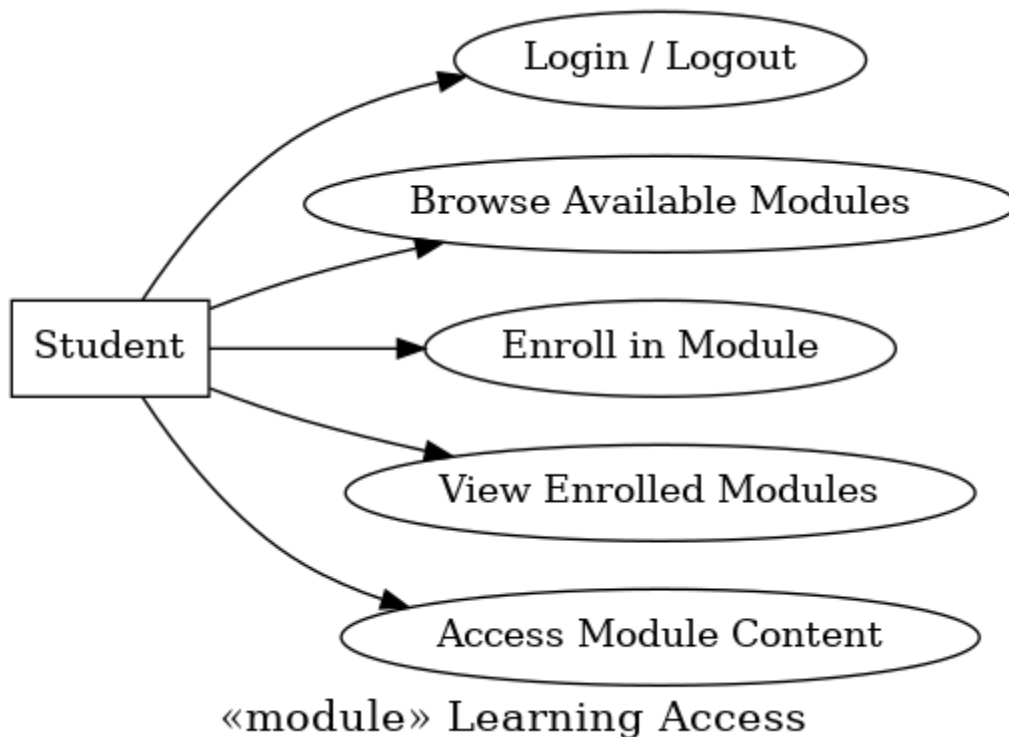
The Trainer is responsible for handling actions related to the students and content within their assigned training modules. The system allows the Trainer to:

- View the list of modules they are assigned to
- Add students to specific modules

- Remove students from modules if needed
- Update notes related to a module

These actions are carried out independently and do not rely on one another. There are no «include» or «extend» relationships in this diagram because each task is a standalone use case within my module interaction process.

Student: Learning Access



This diagram shows the Student's use cases for enrolling in and accessing module content

The Student interacts with the system primarily to manage their learning . The diagram outlines the core actions a student can take to view and engage with modules that are available to them:

- Login / Logout of the system to access features securely
- Browse available modules offered in the system
- Enroll in a module based on interest or requirement
- View enrolled modules to track what they are currently taking
- Access module content such as resources

While each use case is presented separately, most actions rely on the student being logged in first. For example, a student must log in before they can browse available modules or enroll in

one. Once logged in, these actions can be performed independently depending on the student's needs.

There are no «include» or «extend» relationships in this diagram, as each task is a direct and standalone action within the learning access process