# e3Learning - Programming Test

Create a simple web application that illustrates the following simple business case.

The fictious client here is a training organization that provides training to customers and maintains their qualifications on their system. They would like to have a system where they can create accounts and courses they offer and manage their account’s training using web based system. Basic terminology client prefers is as follows. An “account” is a registered user in the system. All accounts have basic attributes like firstname, lastname, email address, active\_state, address(which contains streetname,suburb,state,country). A “course” is a training that organization can offer to customers. Every course has unique id and a title. Accounts can “enrol” into the courses. When the users finish training and achieve a qualification by completing the enrolled course a training record is created. A training record is a snapshot of any user’s training. A training record illustrates what course they did, when they started and when they completed, and the grade they achieved.

Imagine you are building a system for an administrator for the above system. You need various forms to manage the students registration, enrolment and managing their training records.

1. A form where you can create a data store of courses that the training organization can offer.
2. Account creation form. In this form you also enrol the account into pre-existing courses.
3. Create a search form where you can search for account using first name or last name and display results in a table.
4. When admin selects account in the table, it displays a form to be able to add/view training records.

## Technology stack.

* Angularjs (**Not Mandatory**)
* Spring MVC
* JPA with Hibernate provider.
* Mysql

## Artifacts

* **Code and instructions on how to run. Prefer code committed to Git HUB.**
* UML class diagrams(any tool, but can use draw.io) or ER diagrams
* Junit test cases
* Integration test cases

Please remember the emphasis is the below.

*Working and usable quality product is more preferable than having lots of features with bugs.*

Create forms that are simple and intuitive. Use ajax and lazy fetch data where possible.