# Developing an eLearning web app

### Use Cases Identified

1. Create Accounts (Register Students / New Users)
2. Manage accounts
3. Create Course
4. ~~Manage Courses~~
5. Enroll students into courses
6. Create training record
7. View training records
8. Search functionality (firstname / lastname)

### Information given:

* Account is nothing but a registered user

### Assumptions (if any):

* Course once created need not be managed (updated / deleted).
* Once a training record is created, it cannot be deleted.
* Active state of an account is either active / inactive
* Address is not a mandatory field and we are storing only one address per account
* Included active\_ind (active status/indicator) in course to turn on / off a course
* Training start date and completion date are only date and not date timestamp
* Many strings lengths will not be greater than 60. Assumed appropriate length when necessary.
* All tables in order to support optimistic locking, can have version column
* Correspondingly, tables could also have additional information such as last\_modified\_by, last\_modified\_timestamp to identified who and when the record was modified.
* The tables could also include tenant\_id when we plan to support SAAS based model.

### Implementation Notes:

* Developed using the latest STS tool
* I’ve purposefully used JPA annotations wherever possible, to refrain from heavily dependent on hibernate annotations. Wanted to be generic as much as possible.

### Learnings:

* Started the project with maven and spring mvc configuration. Took a long time to get the spring configurations working since it is always tricky with maven and configuring spring beans. The delay started right there.

### Improvements:

* Need to write documentation for javadocs
* Can enhance exception handling
* Can localize the messages

### ER Diagram Snapshot

