[E-Concordia Drive]

1. Actor-Goal List

Actor	Goal
Student	- Log in - learning from lessons / attending video lessons - practice lessons - attempt quiz
Trainer	- Log in - create lessons - update lessons - delete lessons - create quiz - update quiz - delete quiz
Admin	- Log in - review trainer's lessons content - feedback on lessons - publish lessons

2. Use Case Model

Id: UC- 01
Use Case: Review Trainer's lessons content

Description
The use case depends on the trainer's create/edit lessons and then after admin review the content of the lesson before publishing.

Level: User-goal

Primary Actor
Admin, Trainer

Stakeholders and Interests
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Student - Access the lessons on the application to learn about driving

Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions
Admin should be logged into the system

Post Conditions
Success end condition
Review on the lesson is approved based on the guidelines by the admin and published on the system

Failure end condition:
The lesson content is not up to the mark and admin adds feedback to trainers to change it before the admin publishes.

Main Success Scenario

- Trainer create the lesson
- Trainer add content to the lesson
- Admin get notification to review this lesson
- Admin reviews the content of the lesson
- Admin publishes the lesson if all content is appropriate based on guidelines
- Trainer and Students will be notified about the publishment

Extensions
The notification should be sent, otherwise trainer will unable to know without checking the status of the lesson publishment into the system

Special Requirements
The application should be consistent. ie. data should be consistent before and after each operation

---

Id: UC- 02
Use Case: Preview lesson

Description
The use case depends on the student, when the student wants to take the lesson from the application and start watching video content.

Level: User-goal

Primary Actor

Student

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions

- Students should be logged into the system and have an active subscription. ie. payment should be valid
- The lesson is already added by the trainer/admin

Post Conditions
Success end condition
The Student should be able to access the lesson and it's content

Failure end condition:
The student may not be able to access the lesson

Main Success Scenario

- Student logs in to the system
- Student look for the specific lesson that (s)he wants to preview
- Student go to the lesson
- Student is able to preview lesson content and related material

Extensions

- The student can not able to log in to the system
- The student is unauthorized due to inactive payment subscription
- The lesson is deleted by the admin/trainer

Special Requirements
The student's login credentials are stored securely

Id: UC- 03
Use Case: Attempt quiz

Description
The use case depends on the student, when the student wants to attempt the quiz to check his knowledge or do some practice

Level: User-goal

Primary Actor
Student

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions

- Students should be logged into the system and have an active subscription. ie. payment should be valid
- The student should able to access the specific lesson

Post Conditions
Success end condition
The Student should be able to attempt the quiz

Failure end condition:
The student can not be able to attempt the quiz

Main Success Scenario

- Student logs in to the system
- Student look for the specific lesson that (s)he wants to preview
- Student go to the lesson
- Student is able to preview lesson content
- Student can move to attempt the quiz
- Student successfully attempted the quiz

Extensions

- The student can not able to log in to the system
- The student is unauthorized due to inactive payment subscription
- The quiz is not yet posted for the specific lesson

Special Requirements
The student’s login credentials are stored securely

Id: UC- 04
Use Case: Create lessons

Description
The use case depends on the trainer, when the trainer wants to create a new driving lesson

Level: User-goal

Primary Actor
Trainer, Admin

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions
The admin/trainer should be logged into the system

Post Conditions
Success end condition
The Trainer should be able to create a driving lesson

Failure end condition:
The Trainer unable to create a driving lesson

Main Success Scenario

- Admin/Trainer logs in to the system
- Admin/Trainer move to lesson management/ all lessons section
- Admin/Trainer fills up the information required in-order to create lesson
- Trainer submits the lesson to the admin for review

Extensions

- The Admin/trainer can not able to log in to the system
- The Admin/trainer unable to create the lesson due to internal error

Special Requirements

- The Admin/trainer's login credentials are stored securely
- The create lesson interface is a useable enough by the Admin/trainer

Id: UC- 05
Use Case: Edit lesson

Description
The use case depends on the trainer, when the trainer wants to edit an existing driving lesson

Level: User-goal

Primary Actor
Admin, Trainer

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions
The Admin/trainer should be logged into the system

Post Conditions
Success end condition
The Admin/Trainer should be able to edit a driving lesson

Failure end condition:
The Admin/Trainer unable to edit a driving lesson

Main Success Scenario

- Admin/Trainer logs in to the system
- Admin/Trainer move to lesson management/ all lessons section
- Admin/Trainer opens the lesson (s)he wants to edit
- Admin/Trainer edit the content which requires the update
- Trainer submit the lesson again to admin for review

Extensions

- The Admin/trainer can not able to log in to the system
- The Admin/trainer unable to edit the lesson due to internal error as if another trainer may currently editing the same lesson

Special Requirements

- The Admin/trainer's login credentials are stored securely
- The edit lesson interface is a useable enough by the trainer

Id: UC- 06
Use Case: Delete lesson

Description
The use case depends on the trainer, when the trainer wants to delete an existing driving lesson

Level: User-goal

Primary Actor
Admin, Trainer

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions
The Admin/trainer should be logged into the system

Post Conditions
Success end condition
The Admin/Trainer should be able to delete a driving lesson

Failure end condition:
The Admin/Trainer unable to delete a driving lesson

Main Success Scenario

- Admin/Trainer logs in to the system
- Admin/Trainer move to lesson management/ all lessons section
- Admin/Trainer opens the lesson (s)he wants to delete
- Admin/Trainer deletes the lesson

Extensions

- The Admin/trainer can not able to log in to the system

Special Requirements

- The Admin/trainer’s login credentials are stored securely
- The delete lesson interface is a useable enough by the trainer

Id: UC- 07
Use Case: Add quiz

Description
The use case depends on the trainer, when the trainer wants to add quiz for the specific lesson

Level: User-goal

Primary Actor
Admin, Trainer

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions

- The Admin/trainer should be logged into the system
- The lesson is already added in the application which Admin/trainer needs to create a quiz for

Post Conditions
<u>Success end condition</u>
The Admin/Trainer should be able to add a quiz for the lesson

<u>Failure end condition</u>:
The Admin/Trainer unable to add a quiz under a lesson

Main Success Scenario

- Admin/Trainer logs in to the system
- Admin/Trainer move to lesson management/ all lessons section
- Admin/Trainer opens the lesson (s)he wants to add a quiz under
- Admin/Trainer fill up the create quiz form with the necessary information
- Admin/Trainer adds the options for each quiz question

- Quiz addition is completed for that lesson

Extensions

- The Admin/trainer can not able to log in to the system

Special Requirements

- The Admin/trainer's login credentials are stored securely
- The add quiz with multiple options under a given lesson interface is a useable enough by the Admin/trainer
- The Admin/trainer can able to re-order the options for quiz question

---

Id: UC- 08
Use Case: Edit quiz

Description

The use case depends on the trainer, when the trainer wants to edit quiz for the specific lesson

Level: User-goal

Primary Actor

Admin, Trainer

Stakeholders and Interests

Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions

- The trainer should be logged into the system
- The lesson is already added in the application which trainer needs to create a quiz for
- The quiz should already be present in the application under that lesson

Post Conditions

Success end condition

The Admin/Trainer should be able to edit a quiz for the lesson

Failure end condition:

The Admin/Trainer unable to edit a quiz under a lesson

Main Success Scenario

- Admin/Trainer logs in to the system
- Admin/Trainer move to lesson management/ all lessons section
- Admin/Trainer opens the lesson (s)he wants to add a quiz under
- Admin/Trainer fill up the edit quiz form with the necessary information
- Admin/Trainer edit the options for each quiz question
- Quiz edition is completed for that lesson

Extensions

- The Admin/trainer can not able to log in to the system

Special Requirements

- The Admin/trainer's login credentials are stored securely
- The edit quiz with multiple options under a given lesson interface is a useable enough by the Admin/trainer
- The Admin/trainer can able to re-order the options for quiz question

---

Id: UC- 09
Use Case: Delete quiz

Description
The use case depends on the trainer, when the trainer wants to delete quiz for the specific lesson

Level: User-goal

Primary Actor
Admin, Trainer

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions

- The Admin/trainer should be logged into the system
- The lesson is already added in the application which Admin/trainer needs to create a quiz for
- The quiz should already be present in the application under that lesson

Post Conditions

Success end condition
The Admin/Trainer should be able to delete a quiz for the lesson

Failure end condition:
The Admin/Trainer unable to delete a quiz under a lesson

Main Success Scenario

- Admin/Trainer logs in to the system
- Admin/Trainer move to lesson management/ all lessons section
- Admin/Trainer opens the lesson (s)he wants to delete a quiz under
- Admin/Trainer deletes the quiz
- Quiz is deleted for the selected lesson

Extensions

- The Admin/trainer can not able to log in to the system

Special Requirements

- The Admin/trainer's login credentials are stored securely
- The delete quiz with multiple options under a given lesson interface is a useable enough by the Admin/trainer

Id: UC- 10
Use Case: Comment summary

Description
The use case depends on the admin, when the admin wants to review notification section

Level: User-goal

Primary Actor
Admin

Stakeholders and Interests
Student - Access the lessons on the application to learn about driving
Admin - Review lessons content, publish it on the application.
Trainer - Manage lessons and quiz
Developer - application implementation team who implements the features
Tester - application testing team who tests the features before releasing

Pre-Conditions

- The admin should be logged into the system
- The admin should be on the notification sub-tab

Post Conditions

Success end condition
The Admin should be able to review all the comments for lessons in the notification section

Failure end condition:
The Admin is unable to view the notification section

Main Success Scenario

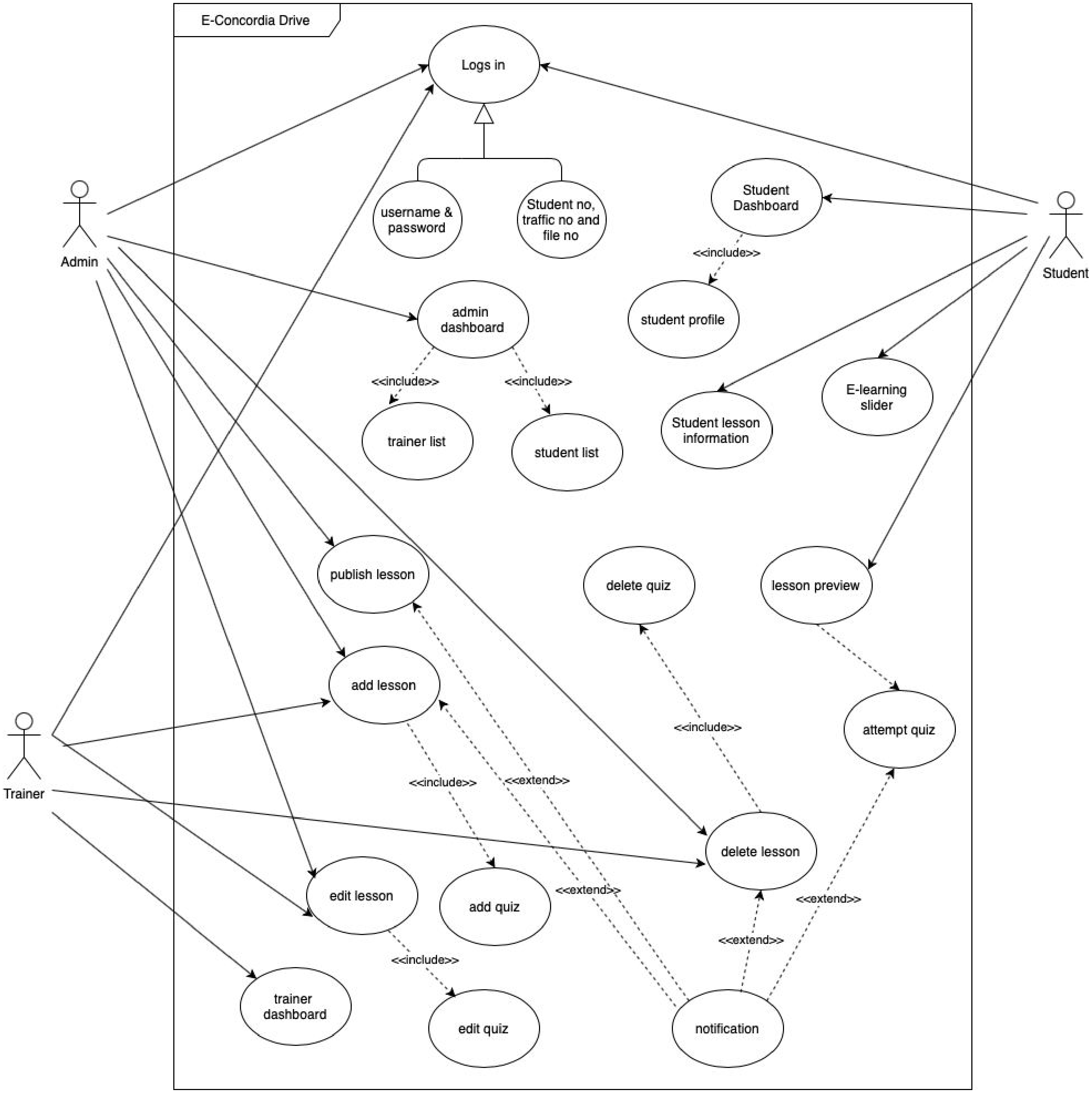
- Admin logs into the application
- Admin moves to Notification tab section
- Admin can able to see the list of all notification and the comment with status and lessons

Extensions

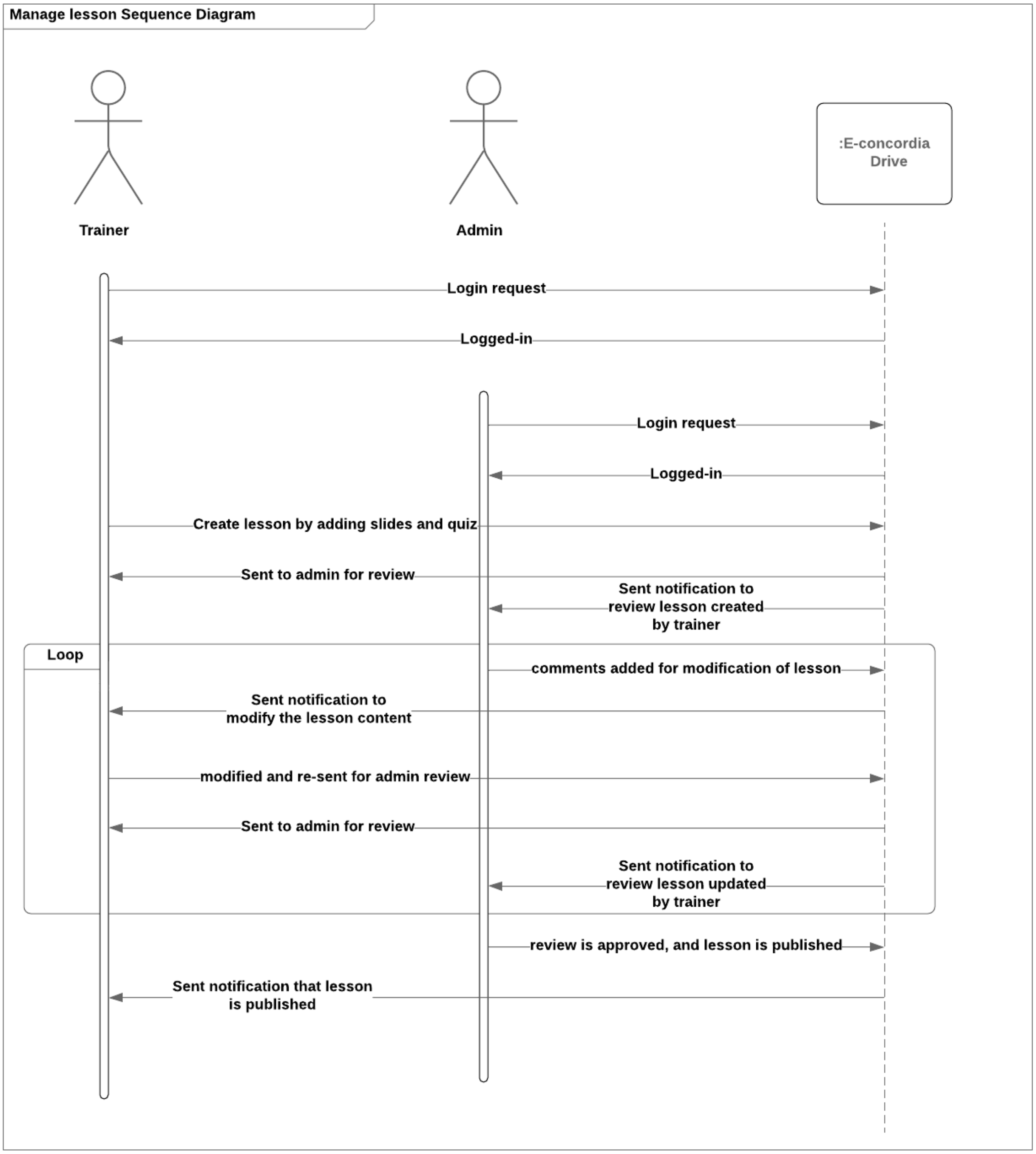
- The admin can not able to log in to the system
- There will be no notifications

Special Requirements

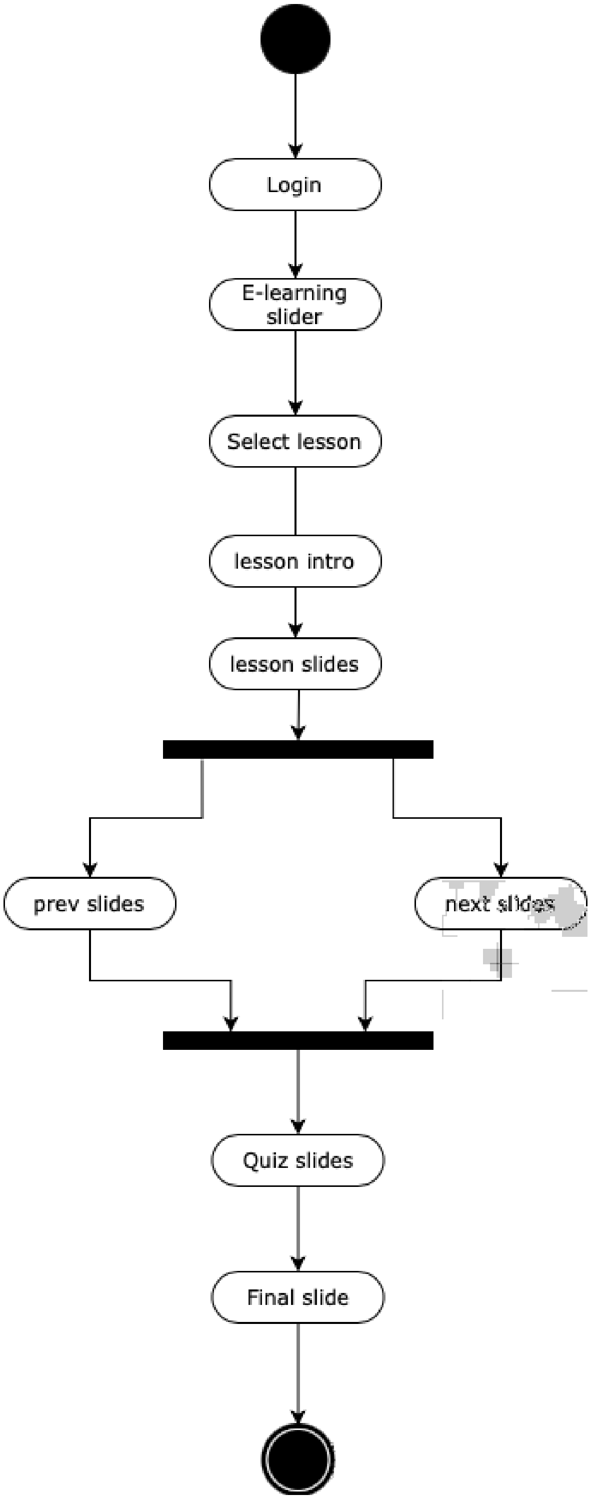
- The admin's login credentials are stored securely
- The notification section having action column, which should redirect the admin to the specific lesson



[UML USE CASE DIAGRAM]


Manage lesson Sequence Diagram
Trainer
Admin
:E-concordia Drive
Login request
Logged-in
Login request
Logged-in
Create lesson by adding slides and quiz
Sent to admin for review
Sent notification to review lesson created by trainer
Loop
comments added for modification of lesson
Sent notification to modify the lesson content
modified and re-sent for admin review
Sent to admin for review
Sent notification to review lesson updated by trainer
review is approved, and lesson is published
Sent notification that lesson is published

[UML SEQUENCE DIAGRAM]



[UML ACTIVITY DIAGRAM]

APPENDIX

Activity	Duration
Actor goal List	15 minutes
Use Case Model	2 hour 30 minutes
Use Case Diagram	1 hour
Sequence Diagram	1 hour
Activity Diagram	30 minutes

[E-Concordia Drive]
Supplementary Specification

1. Introduction

The document defines the supplementary requirements for E-concordia drive, a web based application for learning and practicing driving lessons to obtain a licence. The supplementary specification together with the use case model called as Software Requirement Specification (SRS), which is used to determine the functional and nonfunctional requirements for the E-Concordia drive.

1.1. Purpose

The main objective of this document is to elaborate requirements which are not easily derivable from use case models. It focuses on nonfunctional requirements, application standards and rules alongside with quality attributes of the system which are already defined under the performance requirement of the vision document and hardware requirement constraint.

1.2. Scope

The solution offers flexibility to students to learn lessons self-paced with updated content, provides trainers to reduce quantitative efforts so that more qualitative content can be generated and Admin can inspect content quality before publishing to students. And, all these activities can be done through smart devices from any place

1.3. Definitions, Acronyms and Abbreviations

CDN - Content distribution network

OS - Operating system

IE - Internet explorer

MTTR - Mean time to repair

MTTF - Mean time to failure

1.4. References

- Project description provided by the professor
- E-concordia drive vision document
- UCM-Template for E-concordia drive

2. Functionality

2.1. Error and logs

- application should store the errors in the separate section to identify the system risk and later used to fix the loopholes.
- application must log accessibility information in-order to verify the action performed by the user and the timestamp.

2.2. Security

- The security constraint must be applied in order to prevent unauthorized access.

2.3. Connectivity check

- The application should check the internet connectivity and speed after the login and based on the connection speed it loads the content from cache or CDN.

2.4. Training

- The application may contain the user manual and/or the contact-us in order to get the training on how the application works. So, that all admin, trainers and students are aware about the new solution application.

3. Usability

3.1. Supported OS

- The application can be accessible from the macOs, Windows and Linux operating system.

3.2. Supported browsers

- The application can be accessible from Google chrome, Mozilla Firefox, Safari and IE.

4. Reliability

4.1. Availability

- The application can be accessible to the end user at any given time, 100% available unless maintenance is scheduled during a specified time window.

4.2. MTTF

- The application failure can be within two days.

4.3. MTTR

- The users are unable to access the application for 8 hours. Ie. The application can remain a maximum of 8 hours of unoperational.

4.4. Accuracy

- The application must be 100% accurate as it stores the quiz related question answer which is the most important part. if the accuracy is compromised then it will impact the result.

5. Performance

5.1. Capacity

- It would be good to have multiple sessions supported by the application for the same user. ie. one user can access the application from chrome and from the safari at the same time.

5.2. Response time

- The application would have a maximum of 3 seconds of response time. It would be good to have a caching mechanism added in-order to reduce the latency.

5.3. Resource utilization

- The application can use the scalable database using any cloud techniques in-order to achieve scalability.

6. Supportability

6.1. Coding standard

- The application should follow the coding standards of the PHP framework. It would be good to have an object oriented approach followed in the application implementation.

6.2. Naming conventions

- The naming should be properly used so that one can easily understand from its name what it means.

6.3. Maintenance utilities

- The application must have periodic maintenance even if there is no new feature addition. This way engineers can clear the system cache and related files and storage in-order to improve the data consistency and faster response time.

7. Design Constraints

7.1. Software and technical methodology

- The application should follow the agile process for implementation. The application will be implemented using the PHP framework.

7.2. Hardware and Software requirements

- The application can be run on the above mentioned OS on the laptop/computers by any company.

8. Online User Documentation and Help System Requirements

- The manual and training can be provided along with the application in-order to understand the application operations by the end users.
- The contact-us option will also be given to the users for any real time inquiry.

9. Glossary

CDN - Content distribution network

OS - Operating system

IE - Internet explorer

MTTR - Mean time to repair

MTTF - Mean time to failure

APPENDIX

Activity	Duration
Supplementary specification document fill up	1 hour 30 minutes