CENG3004: Software Engineering

**VISUALEARN**

**Design Document**

**05/06/2024**

**Team Members:**

GÜLSİNEM KAZANCI 200709056

NAZLI ÇIPLAK 200709061

SUDE EBRAR ÇAT 200709007

ZEYNEP HUMAÇ 200709026

**DISCLAIMER**

This document is produced based on Software Requirements Specification (SRS) that was delivered to and agreed by the customer. For the success of the project, it needs to be developed carefully tracing back to requirements as applicable, set design goals and address design goals thoroughly.

This page can be removed once read.

Contents

[1 Overview 3](#_Toc513410287)

[2 Design Goals 3](#_Toc513410288)

[3 System Models 3](#_Toc513410289)

[3.1 Class Diagrams 3](#_Toc513410290)

[3.2 Sequence Diagrams 3](#_Toc513410291)

[3.3 Actvitiy Diagrams 3](#_Toc513410292)

[3.4 Statechart Diagrams 3](#_Toc513410293)

[4 Subsystem Decomposition 4](#_Toc513410294)

[5 Hardware / Software mapping 4](#_Toc513410295)

[6 Other Design Concerns (use relevant subsections) 4](#_Toc513410296)

[6.1 Concurrency 4](#_Toc513410297)

[6.2 Data Management 4](#_Toc513410298)

[6.3 Global Resource Handling 4](#_Toc513410299)

[6.4 Boundary Conditions 4](#_Toc513410300)

[7 Glossary 5](#_Toc513410301)

[8 References 5](#_Toc513410302)

[9 Appendix 5](#_Toc513410303)

# Overview

Our project, VisuaLearn, revolutionizes traditional teaching methods by integrating artificial reality (AR) technology to provide students with a more immersive and effective learning experience. The platform offers a wide range of functionalities to enhance education through AR.

The ultimate goal of VisuaLearn is to revolutionize education by offering students a more effective and enjoyable learning experience through the integration of virtual reality technology. By providing immersive learning environments and interactive experiences, VisuaLearn aims to enhance student engagement, comprehension, and retention of course materials.

User Registration: Users can easily register on the platform to access virtual classes and resources.

Virtual Classes: Interactive virtual classrooms allow students to attend lectures, participate in discussions, and collaborate with classmates and instructors.

Laboratory Experiments: Through AR simulations, students can conduct experiments and practical activities in a virtual laboratory environment.

Virtual Field Trips: VisuaLearn offers virtual field trips to various locations, allowing students to explore and learn from real-world environments without leaving the classroom.

# Design Goals

|  |  |  |
| --- | --- | --- |
| **Goal’s Concern[[1]](#footnote-1)** | **Related Requirement Identifier**[[2]](#footnote-2) | **Description** |
| Performance | NR-1 | User inputs, clicks or gestures, receive feedback within 1 second. Complex interactions, loading new virtual environments or rendering high-detail AR objects, complete within 3 seconds. |
| Usability | NR-2 | We optimized usability for all users on the VisuaLearn platform. We designed an intuitive interface with high-contrast and color-blind friendly palettes, adjustable color settings, all elements are clearly labeled with icons and text, and regular user testing to guarantee accessibility and ease of use. |
| Security | NR-3 | All user data, including personal information and interaction data within AR environments, is encrypted both in transit and at rest using industry-standard encryption protocols (AES-256). Also multi-factor authentication for accessing sensitive features, user identities are verified through at least two independent methods. |
| Reliability | NR-4 | The system is operating without errors continuously over a defined period. System errors is only acceptable for a specific duration per year, a 99.9% uptime should be guaranteed. |
| Scability | NR-5 | For scalability, the platform uses elastic cloud resources to automatically scale up for up to 10,000 concurrent users during peak times and scale down when demand decreases, ensuring optimal performance and cost efficiency. |
|  |  |  |
|  |  |  |

# System Models

## Class Diagrams

metin, ekran görüntüsü, diyagram, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu

## Sequence Diagrams

metin, ekran görüntüsü, çizgi, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

## Actvitiy Diagrams

ekran görüntüsü, diyagram, metin, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

ekran görüntüsü, diyagram, metin, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, diyagram, tasarım içeren bir resim

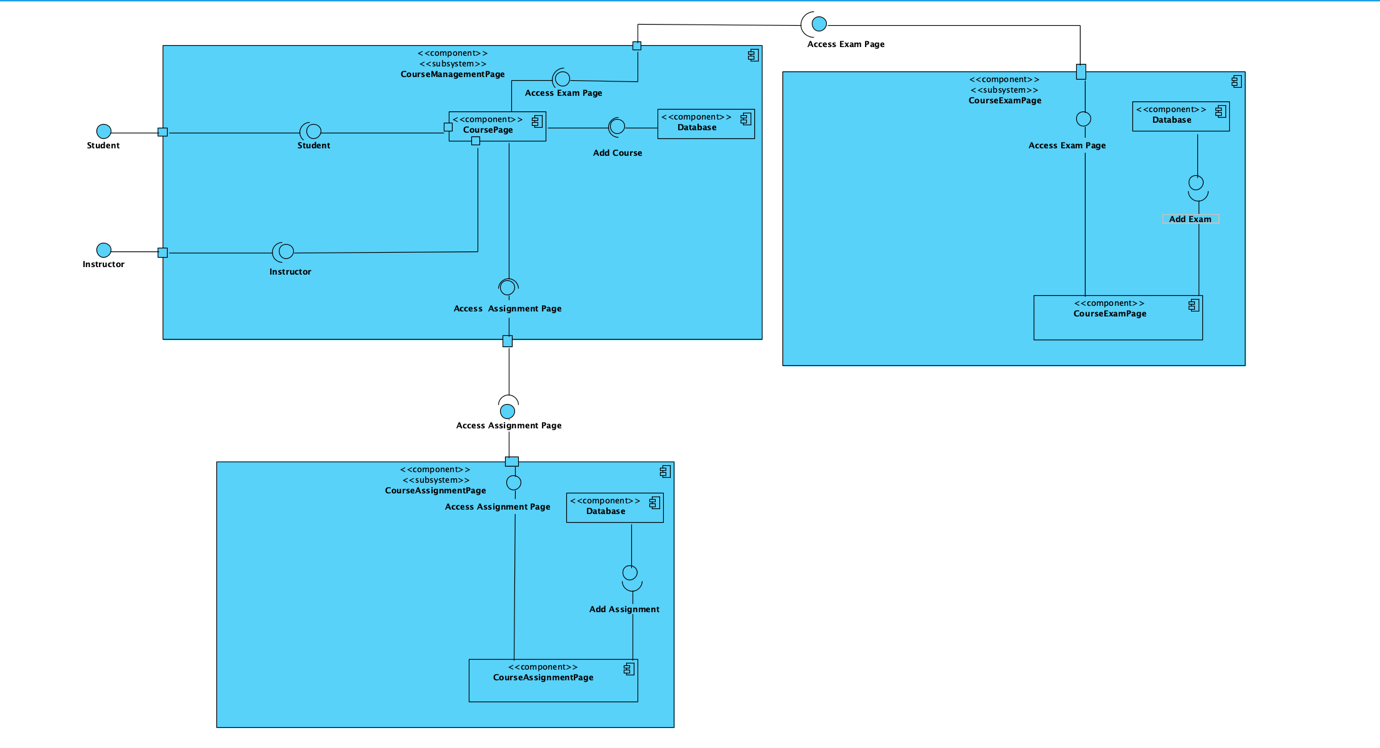
Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, diyagram, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

## Statechart Diagrams diyagram, ekran görüntüsü, tasarım içeren bir resim Açıklama otomatik olarak oluşturuldu

# Subsystem Decomposition



# Hardware / Software mapping

metin, diyagram, çizgi, plan içeren bir resim

Açıklama otomatik olarak oluşturuldu

# Other Design Concerns (use relevant subsections)

## Concurrency

Every transaction other than student course registration can be done simultaneously. If the student's course hours conflict, this should be taken into consideration during course registrations.

## Data Management

metin, ekran görüntüsü, paralel, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

## Global Resource Handling

Discuss authentication/authoritization mechanisms. Provide an “Access Matrix” using f ctors should correspond to actors in uses cases from Analysis Document..

|  |  |  |
| --- | --- | --- |
| **Class/Actor** | Student | Instructor |
| Student |  | createAssignment()  createExam()  setGrade()  grading() |
| Instructor |  |  |
| Course | registerCourse()  attendCourse()  attendExam()  uploadAssignment()  addNote() | startCourse()  endCourse()  createAssignment()  createExam()  createCourse()  setPercentage()  setTitle()  deleteCourse() |
| CourseMaterial | viewMaterial()  attendMaterial() | startMaterial()  materialAlert()  createMaterial() |
| Salary |  | getPayment() |
| Schedule | createSchedule()  editSchedule()  viewSchedule() | createSchedule()  editSchedule()  viewSchedule() |
| Comment | editComment()  deleteComment()  createComment() |  |
| Equipment | checkEquipment()  wearEquipment() | checkEquipment()  wearEquipment() |
| Payment | makePayment() |  |

## Boundary Conditions

Discuss boundary conditions initialization, termination and failure. See below for some questions relevant:

Initialization

* What data need to be accessed at startup time?

At startup, the user's sign-in credentials are accessed.

* What services have to be registered?

Online hosting services and database servers are needed.

* What does the user interface do at start up time?

There is a login page showing.

Termination

* Are single subsystem is allowed to terminate?

If an error occurs in the VR glasses, the related subsystem will automatically terminate.

* Are subsystems notified if a single subsystem terminates?

When a course, exam, virtual trip, or virtual lab is ending, for example, subsystems are notified based on the context.

* How are updates communicated to the database?

The DB Connector is used to update the database when there is a software upgrade. Instructors can also directly update databases.

Failure

* How does the system behave when a node or communication link fails?

When the user disconnects from the system, it alerts them and kicks them out.

* How does the system recover from failure?

The most recent backup's contents are used to restore the system.

# Glossary

SuccesScore: It is the total success percentage affected by the student's course attendance, homework, course participation and exam grades.

Grading: It is the name given to the system's grading of students' exams in courses.

GradingScale: It is the percentage that tells the instructor how much the exams in the course will affect the grade.

exprationDate: Expration date of the card used in the payment system.

Duration: Represents the time frame for a course, with attributes to store and manage start and end dates and total duration in days.

Salary: Manages financial remuneration for instructors, with functions to set and retrieve account numbers and amounts.

# References

<https://www.classvr.com/virtual-reality-in-education/>

<https://en.wikipedia.org/wiki/Artificial_Reality>

<https://medium.com/@seoproctur/a-best-student-management-system-their-features-and-future-scope-proctur-665afde55bd5>

<https://open.lib.umn.edu/humanresourcemanagement/chapter/6-3-types-of-pay-systems/>

# Appendix

1. Examples: Reliability, Modifiability, Maintainability, Understandability, Adaptability, Reusability, Efficiency, Portability, Traceability of Requirements, Fault Tolerance, Backward-Compatibility etc.. Check class slides for examples. [↑](#footnote-ref-1)
2. The non-functional requirement identifiers from Analysis/Software Requirements Document [↑](#footnote-ref-2)