



**SOFE 3650-Fall 2022**

**Software Design and Architectures**

**Project ADD Iteration**

*Date: November 13th, 2022*

Rayan Pasha - 100704599

Andy Lau - 100754380

Mohammed Khan - 100756901

## Use Cases

Use Case	Description
UC1: Course registration	Courses are displayed to the user, and allowed to filter for or search for courses. Upon successful registration, the system database is updated.
UC2: Check grades for registered courses	Students are able to make a request to the system database to check their grades for their current courses.
UC3: Subscribe to exams	Students are able to subscribe to their exams
UC4: Private login	A user logs into the system through a login/password screen. Upon successful login, the user is presented with different options
UC5: Edit Grades	The lecturer adds, removes or edits grades from the system database.
UC6: Schedule an exam	Lecturer adds or edits the date for an exam in the system database.
UC7: Downtime warning	Maintainer inputs a message through the system that is displayed to all users to inform them of the system going down temporarily.
UC8: Add new course(s)	Administration adds, edits or removes a course or courses from the system database.

## Quality Attributes

ID	Quality Attribute	Scenario	Associated Use Case
QA-1	Availability	If the system fails during normal operation then log the fault and resume operations in 1 second.	All
QA-2	Usability	If the user wants to change the system language to Dutch or English during normal operations. They will be able to do so by clicking a button and the system will change language in 3 seconds.	All

QA-3	Security, Performance	If a student logs into the system, it returns a student profile UI with no downtime.	UC4: Private Login
QA-4	Security, Performance	A user performs a change in system data during normal operations. It is possible to know who performed the operation and when it was performed 100% of the time. System data is changed within 24 hours	UC1:Course Registration UC3: Subscribe to Exams UC5: Edit Grades UC6: Schedule an Exam
QA-5	Interoperability, Performance	A student requests data from the system during normal operations. The system sends the data to the student and continues to operate with no downtime.	UC2: Check Grades For Registered Courses
QA-6	Modifiability	It is expected that new students, lecturers, and courses will be added to the system in the future. They should be able to be added to the system with minimal changes.	UC2: Check Grades For Registered Courses UC1:Course Registration

### Constraints

ID	Constraint
Con-1	Students that do not have prerequisite courses should not be able to register for the related course
Con-2	The system must be able to run on different types of browsers and devices
Con-3	Only the administration, system database and, the system maintainer are able to change the data inputs of the course management system
Con-4	Must support at least 100 users using the system at the same time
Con-5	The system must be up and running for the majority of the year. (2 week yearly maintenance )
Con-6	A unique and specific database server must be created and used for the course management

	system and for the course management system alone.
--	--

## Concerns

ID	Concerns
CRN-1	Maintain the privacy and security of the data within the system
CRN-2	A experienced team must be made for the development and maintenance of the course management system
CRN-3	Creating a functional and comprehensive initial structure for the course management system to be developed on

## Step 1: Review Inputs

Category	Details
Design Purpose	This system is a greenfield system in a mature domain, the purpose is to create a course management system for the university
Primary Functional Requirements	UC1: Because it directly supports the core program UC2: Because it directly supports the core program UC3: Because it directly supports the core program UC4: Because it directly supports the core program UC5: Because it directly supports the core program UC6: Because it directly supports the core program
Quality Attributes	QA-1: Availability QA-3: Security, Performance QA-4: Security, Performance QA-5: Interoperability, Performance

Constraints	Con-1 Con-2 Con-3
Concerns	CRN-3

## Iteration 1: Establishing an Overall System Structure

### Step 2: Establish Iteration Goal by Selecting Drivers

This is the first iteration in the design of a greenfield system, so the iteration goal is to achieve the overall architecture of the system keeping in mind:

- UC8: Add new course(s)
- Con-4
- Con-5

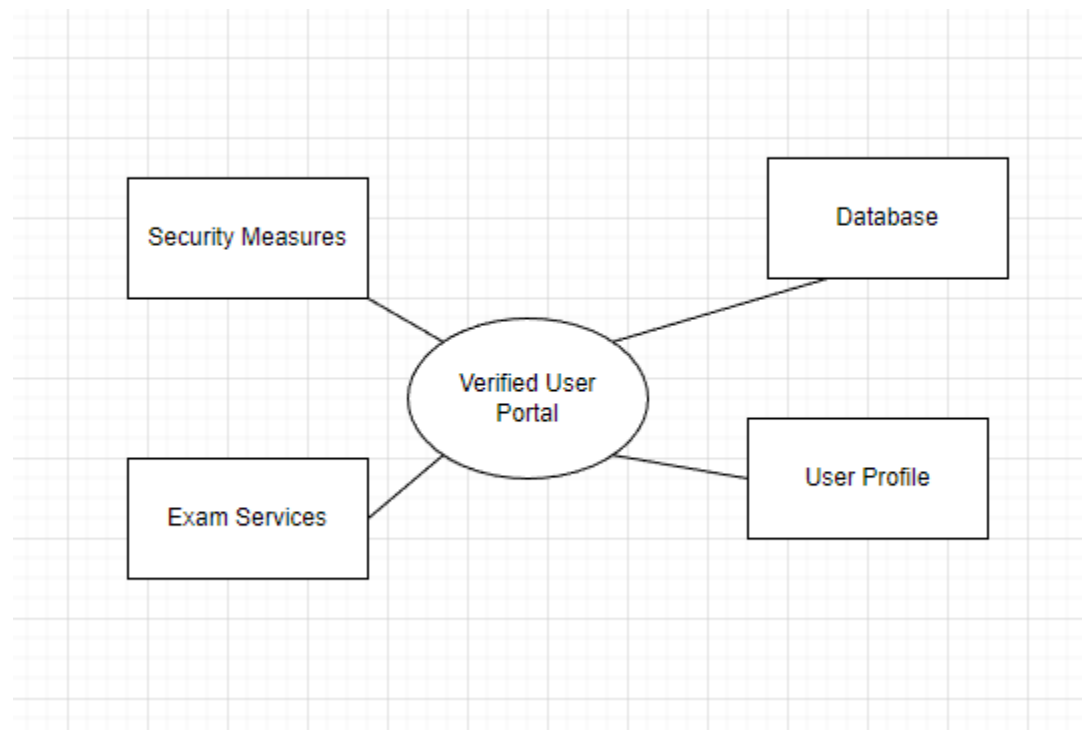


Figure 1. Context diagram of the Verified User Portal

### Step 3: Choose One or More Elements of the System to Refine

We want to refine the whole course management system.

We would like to make the system as accessible to all users as possible.

Understanding the system shouldn't require a learning curve, system should be just as easy to use for new users as to old users

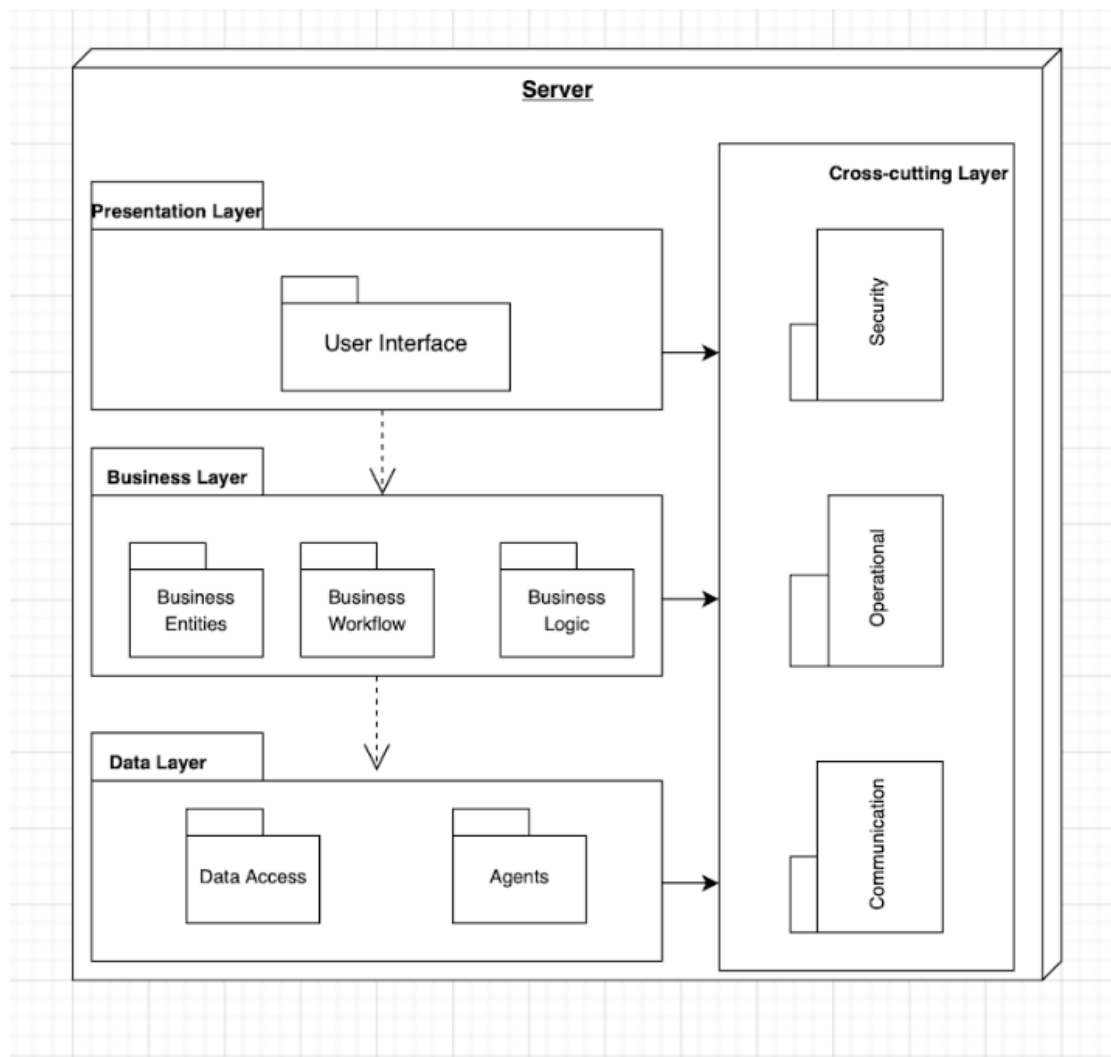
#### Step 4: Choose One or More Design Concepts That Satisfy the Selected Drivers

Design Decisions and Location	Rationale
Use external database for courses, completed courses	This will allow students to register for courses (UC1) while checking what courses they have completed for courses with prerequisites (Con-1)
Each student should have their own account	Having personalized accounts increases privacy and security for students(QA-3), these accounts can be used to check grades (UC2), to register for courses, etc.
Change the language of the course management system	Required by user (QA-2)

#### Step 5: Instantiate Architectural Elements, Allocate Responsibilities, and Define Interfaces

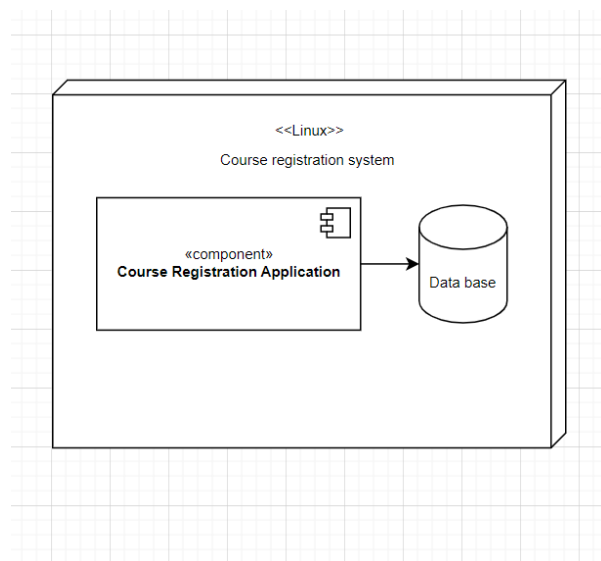
Design Decisions and Location	Rationale
Create a module to dedicated to accessing external database	Facilitates access to the database (UC1) (UC2)
Create a module to dedicated to accessing login	Ensures security and privacy (UC4)
Remove UI Logic module from the Rich Client architecture	The Portal does not require extensive processing on presentation layer so the UI Interface can directly interact with the Business Layer

## Step 6: Sketch Views and Record Design Decisions



Element	Responsibility
Browser	Browser displays UI on the clients machine
UI	Presents UI elements and information to the user
Admin interface	Displays and analyzes user information, stores personal information, courses, grades, etc
Business entities	Responsible for business logic and

	business entities
Business workflow	Coordinates multi step, business procedures
Business logic	Handles information exchange between the database and UI
Agents	Maintain the main and all associated systems
Operational	Makes the system easy to use
Data access	Manages and updates access to be stored or new data,
Communication	Communicates between the different parts and layers of the system
Security	Maintains client server side to insure privacy



Element	Responsibility
Course registration application	Hosts and retains the applications logic in the database

Step 7:



Not Addressed	Partially Addressed	Completely Addressed	Design Decisions Made During the Iteration
		UC1: Course registration	
		UC2: Check grades for registered courses	
	UC3: Subscribe to exams		
	UC4: Private login		
	UC5: Edit Grades		
	UC6: Schedule an exam		
	QA-1: Availability		
	QA-3: Security, Performance		
	QA-4: Security, Performance		
	QA-5: Interoperability, Performance		
con-1			
con-2			