

Iteration 1: Establishing an Overall System Structure

The Design Process:

Iteration 1: Establishing an Overall System Structure

This section presents the results of the activities that are performed in each of the steps of ADD in the first iteration of the design process.

ADD Step 1: Review Inputs

The first step of the ADD method involves reviewing the inputs and identifying which requirements will be considered as drivers.

Category	Details		
Design Purpose	This is a greenfield system from a mature domain. This is a new system based on existing patterns and styles		
Primary Functional Requirements	From the use cases presented earlier, the primary ones were determined to be: UC-1: Because for a Course management system, these are the core components UC-2: Because for a Course management system, these are the core components UC-7: Because for a Course management system, this case collects all the course related contents associated with it		
Quality Attribute Scenarios	ID	Importance to the Customer	Difficulty of implementation according to the architect
	QA-1	High	Low
	QA-2	High	High
	QA-3	High	Low
	QA-4	High	High

	QA-5	Low	Low
	QA-6	High	Medium
	QA-7	High	Low
	QA-8	Medium	Medium
	QA-9	Low	Low
	QA-10	Low	High
	QA-11	Medium	Low
	QA-12	Medium	Low
From this list only QA-1, QA-2, QA-3, QA-4, QA-10 and QA-11 are selected as drivers			
Constraints	All the constraints are discussed earlier		
Concerns	All the concerns are discussed earlier		

ADD Step 2: Establish Iteration Goal by Selecting Drivers

Selected Drivers for QA:

- QA-2: Availability
- QA-4: Usability
- QA-10: Scalability

Selected Drivers for CON:

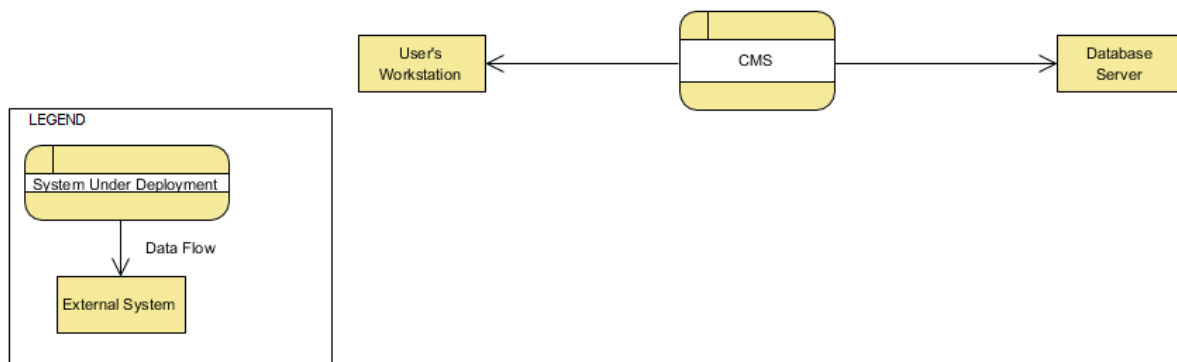
- CON-1: The system must be accessed through a web browser (Chrome, Firefox, IE, etc.) in different platforms: Windows, IOS, Linux, Tablets and phones
- CON-5: A large relational database is needed to maintain the progress and registration requirements of students
- CON-3: The network connection to user workstations must have a high bandwidth

Selected Drivers for CRN:

- CRN-1: Architecture planning of a large relational database with many complexities.

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

This is a Greenfield development effort, so in this case the element to refine is the entire CMS (Course Management System), which is shown in the diagram below.



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

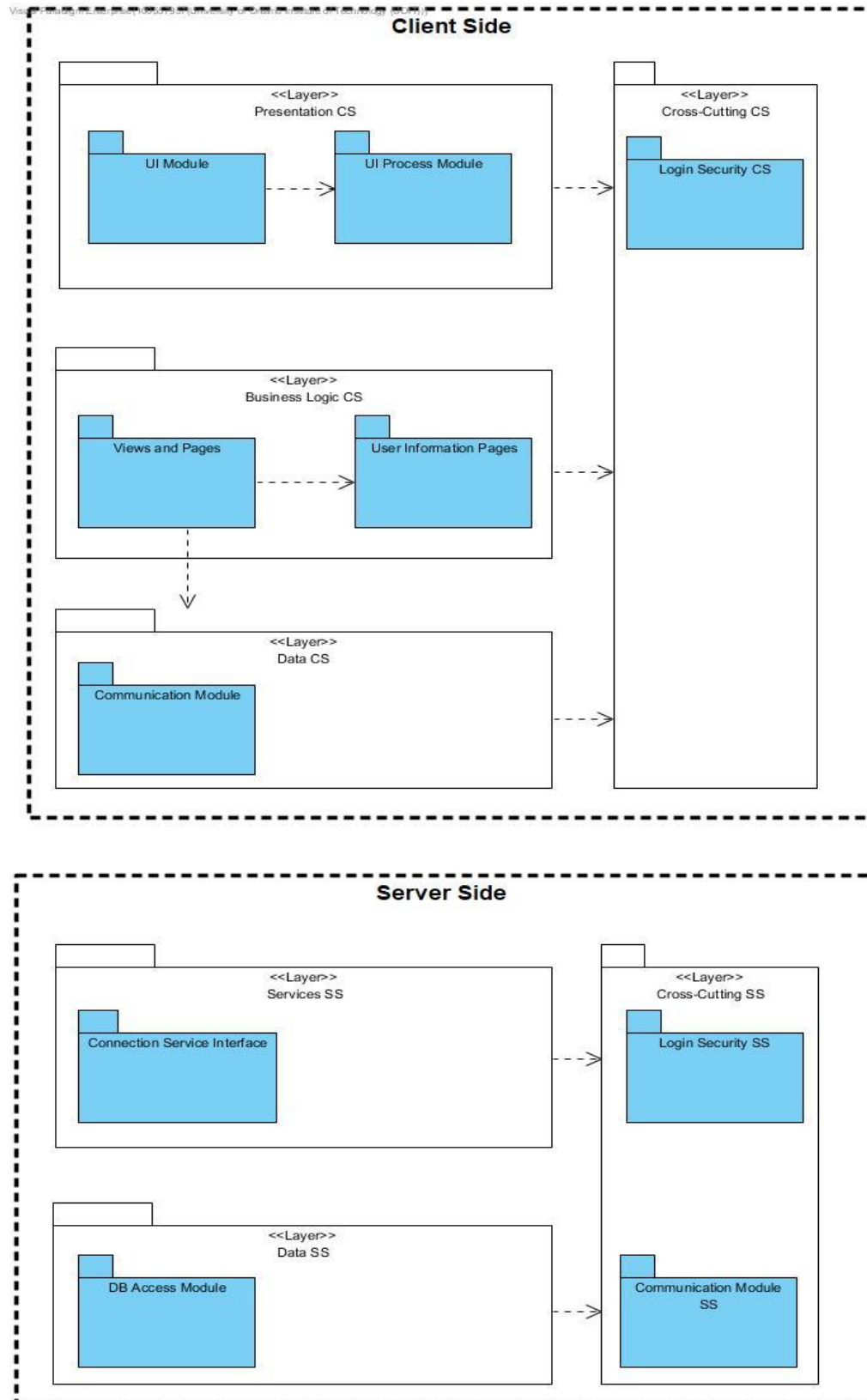
Design Decisions and Location	Rationale
Web Applications reference architecture to structure the client part of the system	The bulk of the application resides on the server, and its architecture is typically composed of three layers: the presentation, business, and data layers. But in this case we have presentation and data layers. The presentation layer contains modules that are responsible for managing user interaction (CON-5). The data layer contains modules that manage data that is stored either locally or remotely (CON-3).
Three-tier deployment pattern	Since the system must be accessed from a web browser (CON-1) and an existing database server must also be

	<p>used (CON-5), a three-tier deployment is appropriate. The three-tier deployment:</p> <ul style="list-style-type: none"> • Allows flexibility in connection with components in different levels. • Allows scalability as additional levels can be added. • Adds firewall security between layers and security policy to the layers. • The database can be stored separately from the application in a layer.
--	--

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

Design Decision and Location	Rationale
UI	The User Interface will control the flow of the pages and make it clean and easy to understand as possible

ADD Step 6: Sketch Views and Record Design Decisions



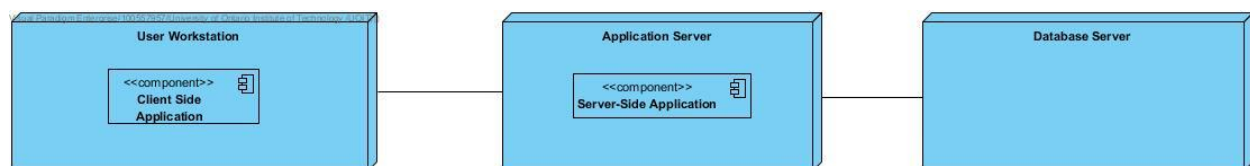
Client Side:

Element	Responsibility
UI Module	These modules render the user interface and receive user inputs
UI Process Module	These modules are responsible for control flow of all the system use cases
Login Security	It handles security and user identification so that the web client can identify privileges.
User Information Page	Personal information of the user
Views and Pages	The user can access multiple views to get the right information
Communication Module	Communication between databases

Server Side:

Element	Responsibility
Connection service interface	Receives the connection from the server side
Permissions	Asks the user for information stored in the database
DB Access Module	This is the location of which the information is stored into the relational database

Deployment diagram of the CMS:



Element	Responsibility
User Workstation	The user's workstation, which hosts the client side logic of the application
Application Server	The server that hosts server side logic of the application and also serves web pages
Database Server	The server that hosts the legacy relational database

Relationship	Description
Between application server and database server	Communication with database can be done using PostgreSQL

ADD Step 7: Perform Analysis of Current Design and Review Iteration

Goal and Achievement of Design Purpose

Not Addressed	Partially Addressed	Completely Addressed	Design Decisions Made During the Iteration
	UC-1		Selected reference architecture establishes the module that will support this functionality.
	UC-2		Selected reference architecture establishes the module that will support this functionality.
	UC-3		The layered pattern and web application

			architecture have modules that will support this functionality
UC-14			No relevant decisions made, as it is necessary to identify the elements that participate in the use case that is associated with the scenario
	QA-2		The web server architecture have modules that will support this functionality
	QA-4		A simple UI architecture will have modules that will support this functionality
	QA-10		The web server architecture have modules that will support this functionality.
CON-1			No relevant decisions made, as it is necessary to identify the elements that participate in the use case that is associated with the scenario

	CON-2		Cloud Architecture Patterns have modules that will support this functionality
CON-3			No relevant decisions made, as it is necessary to identify the elements that participate in the use case that is associated with the scenario
	CON-4		Physically structure the application using the 3 tier deployment pattern and isolate the data base by providing components in the data layer of the application server.
	CON-5		Physically structure the application using the 3 tier deployment pattern and isolate the data base by providing components in the data layer of

			the application server.
	CRN-1		By correctly identifying primary and foreign keys, any conflicts within the tables in the database can be eliminated
		CRN-2	Selection of reference architectures and deployment pattern
	CRN-3		Languages were considered and taken into account in terms of the developer's knowledge
CRN-4			No relevant decisions made