Iteration 1: Establishing an Overall System Structure

Iteration 1 outlines the initial results of the design process for the ADD Steps .

Step 2: Establish Iteration Goal by Selecting Drivers

The goal of this iteration is to achieve the architectural concern CNR-1 of establishing an overall system structure. Keeping this in mind, there are several drivers that we as the architects must account for:

1. QA-2: The User has many methods of paying for the art they desire. The system should reflect this and offer many payment methods (bank transfer, e-transfer, cash by mail etc…)
2. QA-4: User's will store private financial information on the site to buy and sell art. Therefore, the system should encrypt the private information of its users like login and financial info.
3. QA-6: The system must be able to complete art transactions
4. QA-7: The system must continue to operate and be available all the time. For UC-4 availability must be predefined
5. QA-8: The system must exchange data with external systems such as banking services and online payment providers
6. CON-1: The system must be accessible and run smoothly by all popular web browsers (mozilla, chrome, edge etc…) and all popular operating systems (Windows, mac os x). Additionally, the system must also support mobile devices.

7. CON-2: All user data including financial and transactional records must be stored indefinitely.
8. CON-3: The system must have a mobile-friendly design.
9. CON-4: The system must work properly when viewed with different resolution monitors. The system must work when the view is stretched or shrunk.
10. CON-5: All art uploaded to the system must have the author's permission.
11. CRN-1: Establishing an overall system structure.

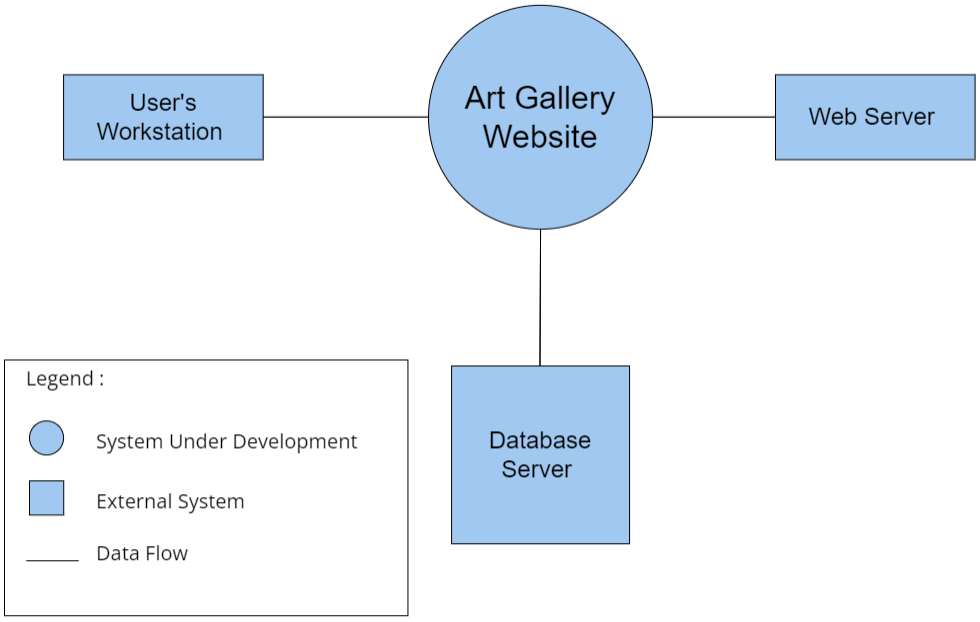

Figure 2: Context Diagram for the Art Gallery Website

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

This is a greenfield project for a mature domain, so the entire art gallery system is to be refined.

All components should be revised so the overall performance and functionality can be improved. refinement is performed through decomposition.

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

The following table summarizes the selection of design decisions:

Design Decisions and Locations	**Rationale**
Logically structure the client and server part of the system using the Web Application reference architecture	The Web Applications reference architecture is selected for its overall usability to construct non-rich web applications. This is because our system does not require a rich user interface, it does not need to install anything on the client machine, and it must be accessible over the internet by web browsers (CON-1). This design decision partially affects all of the other drivers, but does not directly impact any other as substantially as the first concern.
Physically structure the application using the three-tier deployment pattern	This system needs to be accessed through a web browser (CON-1) and there should be an existing database that should be used . Hence the three-tier layer deployment is appropriate .

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

The instantiation design decisions made in this iteration are summarized in the following table:

Design Decisions and Location	**Rationale**
Remove Application Facade for the business layer of the server	The Application facade component is unnecessary to fulfill the requirements of the project overall.
Add Payment System as a external system that communicates with the service agents	The Payment System will be an external system to process payments for the Users. Adding the payment module ensures that many payment methods can be used (UC-3, QA-2), the system can complete art transactions (QA-6), and the system uses an external payment module (QA-8).

Step 6: Sketch Views and Record Design Decision

The diagram below shows the sketch of the module view of the two reference architectures that were selected for the client and server applications. These have now been adapted according to the design decisions we have made.

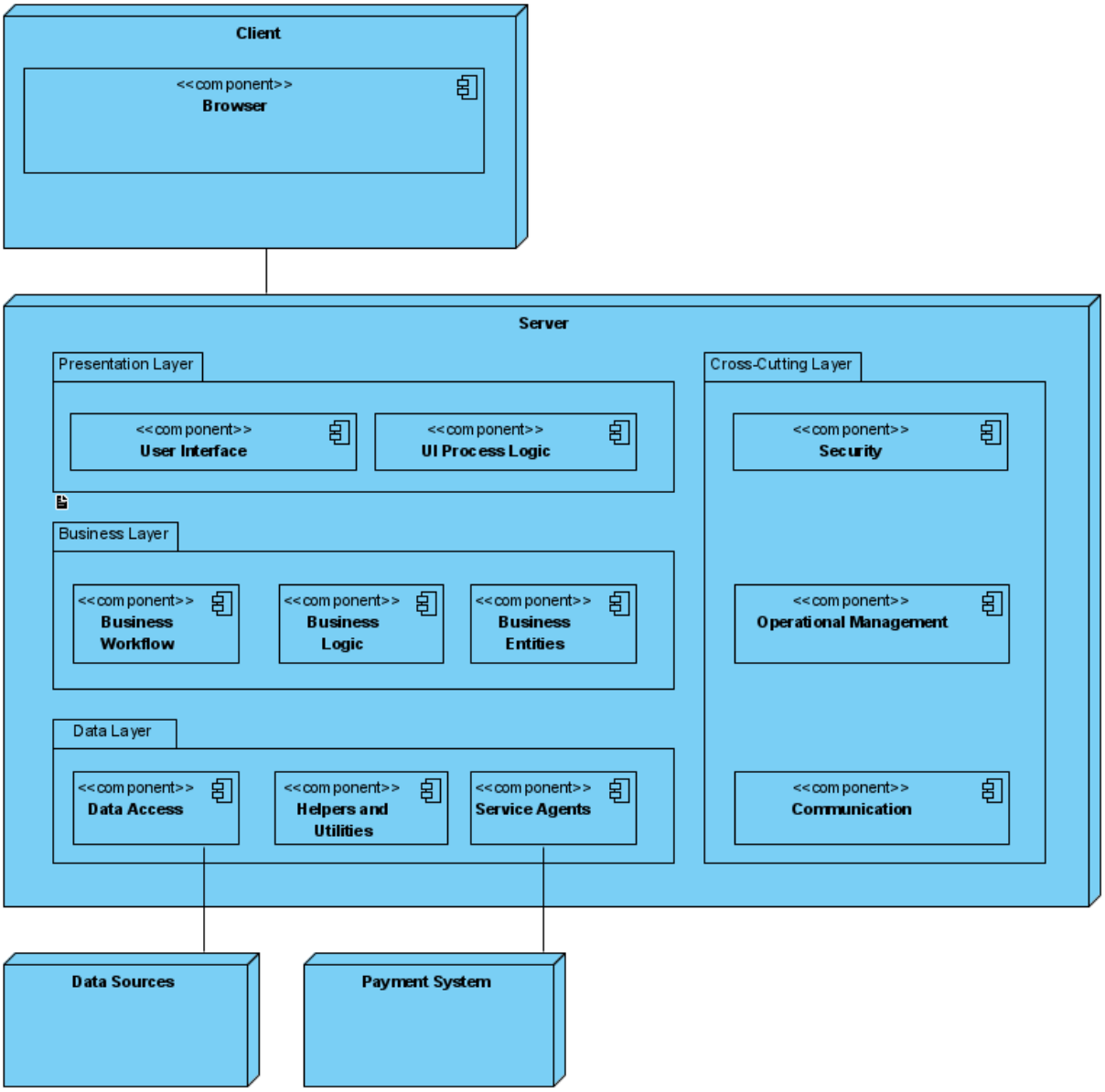

Figure 3: Module View of the website after initial design decisions:

The following table summarize each element of the module view of the diagram alogn with its responsibilities:

Layer	Component	Responsibility
Client		The presentation layer that communicates with the client service modules

	Browser	Application module that is used by the client to interact with the server applications to provide or display information. Runs on the client machine
Server		This layer exposes the modules and components that users will interact with
Presentation		This layer controls user interaction with the server directly, including use case interactions
	User Interface	These components are responsible for receiving/sending information to the users through inputs like buttons, text fields, etc...
	UI process logic	These components are used to direct the flow of the applications use cases. This can include data validation, providing data from business layer to presentation layer, etc...
Business Logic		Contains modules that perform business logic operations on the server side
	Business Workflow	These components are responsible for managing the processes of the business operations, involving the execution of use cases
	Business Logic	These components retrieve and process the data with business rules.
	Business Entities	These components represent the entities of the business domain
Data		This layer contains modules for data persistence and communication with external systems

	Data access	These components encapsulate persistence mechanisms to provide basic operations like retrieving and storing data
	Helpers and Utilities	These components contain functionality common to other modules in the data layer
	Service Agents	These components are necessary for communicating and transferring data between external services and the system itself
Cross-Cutting		These modules have functionality that are designed to work across multiple layers
	Security	These components include functionality to handle security aspects such as authorization and authentication
	Operational Management	These components handle cross-cutting concerns such as exception management, logging, and instrumentation and validation
	Communication	These components handle communication across the layers and physical tiers of the system
Payment Server		This layer is an external system that will communicate with the system to provide payment methods to the users on the website (UC-3, QA-2, QA-6, QA-8)

The following deployment diagram sketches an allocated view that illustrates where the components associated with the modules in Figure 3 will be deployed.

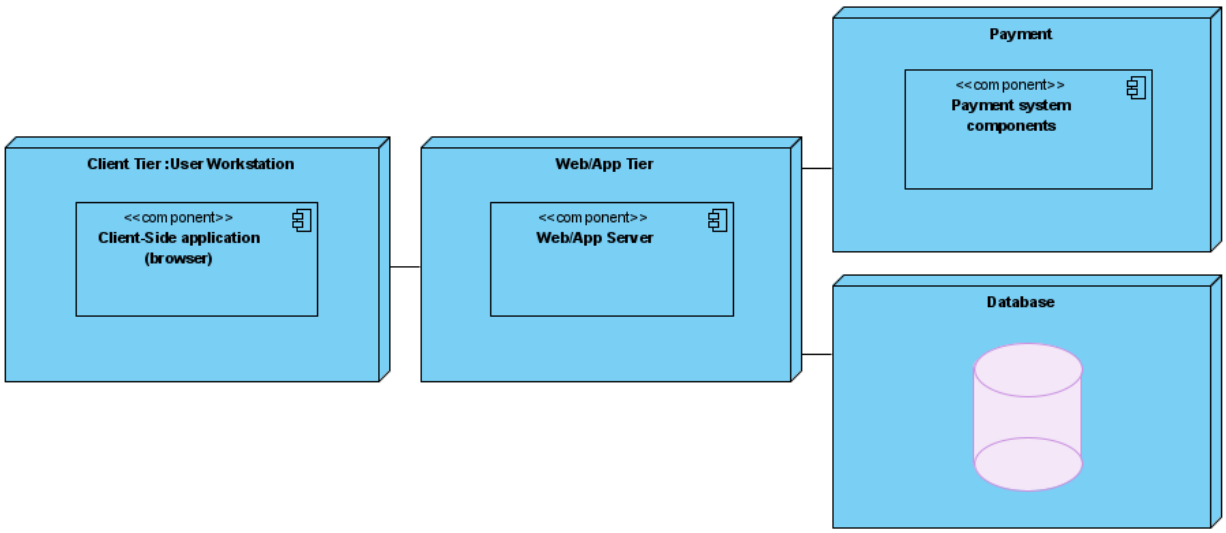

Figure 4:Deployment Diagram of the system

The following table summarizes the responsibilities of the elements :

Element	**Responsibility**
User Workstation	Users PC, hosts client side application, in this case the user would only need a browser to connect to the server
Web/App Server	Hosts the server-side logic and web pages of the application
Database Server	The server that hosts the database
Payment system	The external system used to make payments on the website

Step 7: Perform Analysis of Current Design and Review Iteration Goal and Achievement of Design Purpose

Not Addressed	Partially Addressed	Fully Addressed	Rationale
	UC-2		Introduction of an external payment system supports the functionality of this use-case
	UC-5		Selected reference architecture establishes required modules that will support this functionality
	UC-8		Selected reference architecture establishes required modules that will support this functionality
	QA-2		Introduction of an external payment system supports the functionality of this quality attribute
	QA-4		Selected reference architecture establishes required modules that will support this functionality
	QA-6		Selected reference architecture establishes required modules that will support this functionality
	QA-7		Selected reference architecture establishes required modules that will support this functionality
		QA-8	Introduction of an external payment system

	CON-1		Selected reference architecture establishes required modules that will support this functionality
	CON-2		Selected reference architecture establishes required modules that will support this functionality
CON-3			No relevant decisions were made
	CON-4		Selected reference architecture establishes required modules that will support this functionality
CON-5			No relevant decisions were made
		CRN-1	Selection of reference architecture and deployment pattern
CRN-2			No relevant decisions were made