**Logo

Description automatically generated**

**SOFE 3650U Software Design and Architecture**

**Iteration 3 of Art Gallery Project**

Matthew Gardiner 100768198

**Project Use Cases :**

Table

Description automatically generated

**Quality Attributes :**

Table

Description automatically generated

**Constraints:**

**Table

Description automatically generated with medium confidence**

**Iteration 3: Further identifying structures to support functionality**

The goal of this iteration is to further implement structural functionality to support the primary use cases.

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

The main driver of this iteration will be UC-2: Buy or Sell Arts. The broad approach used to implement functionality in iteration 2 resulted in overlooking some requirements of UC-2. Specifically, the ability for a user to buy art was implemented, but the ability for a user to sell their own art was neglected.

To fulfill UC-2, CON-5 must also be fulfilled as all art uploaded to the website to make a profit must be verified.

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

The elements that are necessary to refine to implement the functionality are listed below:

|  |  |
| --- | --- |
| **Module/Element** | **Rationale** |
| **Tier: Server** | |
| **Layer: Business** | |
| Home Controller | Functionality must be added to the Home Controller that allows users to post artwork on the website. Additionally, functionality to support new modules must be added |
| **Layer: Presentation** | |
| User Interface | A new form for displaying payment options, total profit, revenue, etc. should be added to display to users who sell art. |
| UI Process Logic | New logic to display forms should be added to support UC-2 functionality |
| **Layer: Data** | |
| Payment System Connector | The payment system connector should be updated to add automatic functionality to pay users selling their art. |

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

|  |  |
| --- | --- |
| **Design Decision** | **Rationale** |
| Update Domain Model | The Domain model should be updated to include new functionality |
| Payment Controller | A payment controller module to handle operations regarding paying users must be implemented (UC-2). |
| Art Verification | An external verification system for new art added by users must be added to ensure that all art on the website is sold legally (CON-5). |

\

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

|  |  |
| --- | --- |
| **Design Decision** | **Rationale** |
| Update Domain Model | The Domain Model must be updated to show the new module. |
| Add Payment Controller to Business Layer | The Payment Controller must be added to the business layer to communicate with other modules. The Payment Controller will communicate with the Payment System Connector and the Home Controller. |
| Add functionality to Home Controller to support Payment Controller and Art Verification Connector | Functionality that supports the Payment Controller must be added to the Home Controller. This functionality includes a function makePayment() that communicates with the Payment Controller for paying users.  Additionally, functionality that supports verifying art via connecting to an external art verification system must be instantiated. |
| Add functionality to Payment System Connector to support Payment Controller | Functionality to support automatic paying of users must be added to the Payment System Connector. |
| Add Art Verification Connector to Data Layer | An Art Verification Connector must be added to communicate with the external Art Verification System. The connector should be in the data layer to communicate with external systems |
| Add external Art Verification System | An external Art Verification System must be added to ensure art sold on the website by its users is sold legally. |

**Step 6: Sketch Views and Record Design Decisions**

Updated Domain Model:

Diagram

Description automatically generated

The updated domain module provides clarity to the previous payment module, specifying its use as payments via customers buying art. On the other hand, User Payment has been added to support the functionality of paying users for selling their art. Additionally, the verifyArt() method was added to the model for providing functionality to the external system.

New Domain Objects:

* Added Payment Controller which communicates with the Payment System Connector for handling payments to users selling art.

Diagram

Description automatically generated

* Added Art Verification Connector to connect to the new external Art Verification System

Diagram

Description automatically generated

Updated Domain Object:

Diagram, schematic

Description automatically generated

* The Payment system connector has now also been updated to include UC-2 as one of its responsibilities.

Updated Module View containing new module Payment Controller:

Diagram

Description automatically generated

|  |  |
| --- | --- |
| **Element** | **Responsibility** |
| Payment Controller | This controller contains business logic pertaining to UC-2. Specifically, the controller will process payments to users that are selling art on the website |
| Art Verification System Connector | This connector is responsible for communication between the Home Controller and the external Art Verification System, in other words, a service agent |
| External Art Verification System | Added to verify art posted on the website by the users. |

**Sequence Diagrams:**

**Updated UC-2: *Buy or Sell Art***

A picture containing chart

Description automatically generated

All method names will be further refined as they are implemented.

This sequence diagram now includes posting art to the system.

|  |  |
| --- | --- |
| **Method** | **Definition** |
| **Element: Browser** | |
| Post Art | The User inputs information regarding their artwork like price, the image file, quantity, etc. |
| **Element: User Interface** | |
| Input Art Information | The User inputs information regarding their artwork like price, the image file, quantity, etc. |
| **Element: Home Controller** | |
| Verify Art | The Home Controller must verify the art provided by the user with a external art verification system |
| Input Art Information | This Home Controller must the new artwork into the database |
| **Element: Art Verification System Connector** | |
| Verify Art | Connect to the external Art Verification System to verify art from Home Controller |
| **Element: Art Verification System** | |
| Verify Art | Self message where the Art Verification System verifies data given |
| **Element: Database Access** | |
| Input Art Information | Access Database to input art (price, quantity etc.) |

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

The design decisions in this iteration provided further understanding of the functionality supported in the system with regards to the primary functional use cases. Modules associated with the functionality of the system were identified and defined.

|  |  |  |  |
| --- | --- | --- | --- |
| Not Addressed | Partially Addressed | Fully Addressed | Rationale |
|  |  | UC-2 | The payment system to pay for art has been implemented and the ability for artists to post their art for sale has also been implemented. |
|  |  | CON-5 | The constraint that all art must be verified to be legally sold has been fully addressed with the implementation of an external art system and its modules/interfaces. |