

**Artworks Sharing Platform**

**Software Requirement Specification**

Project Code: FU-SPR2024-NJS1709-SR302

Document Code: FU-SPR2024-NJS1709-SR302-SRS–v1.0

**Ho Chi Minh, 28/01/2024**

**Record of change**

\*A - Added M - Modified D - Deleted

| **Effective Date** | **Changed Items** | **A, M, D** | **Change Description** | **New Version** |
| --- | --- | --- | --- | --- |
|  | Use case: Sign up | Add | Add “Sign up” use case detail | 1.0 |
|  | Use case: Login | Add | Add “Register account” use case detail | 1.0 |
|  | Use case: View profile | Add | Add “View profile” use case detail | 1.0 |
|  | Use case: Edit profile | Add | Add “Edit profile” use case detail | 1.0 |
|  | Use case: Reset password | Add | Add “Reset password” use case detail | 1.0 |
|  | Use case: Remove account | Add | Add “Remove account” use case detail | 1.0 |
|  | Use case: Edit information of artworks | Add | Add “Edit information of artworks” use case detail | 1.0 |
|  | Use case: View profile | Add | Add “View profile” use case detail | 1.0 |
|  | Use case: Edit profile | Add | Add “Edit profile” use case detail | 1.0 |
|  | Use case: View artworks | Add | Add “View artworks” use case detail | 1.0 |
|  | Use case: View artwork detail | Add | Add “View artwork detail” use case detail | 1.0 |
|  | Use case: Remove artwork | Add | Add “Remove artwork” use case detail | 1.0 |
|  | Use case: Upload artwork | Add | Add “ Upload artwork” use case detail | 1.0 |
|  | Use case: View list of user accounts | Add | Add “View list of user accounts  ” use case detail | 1.0 |
|  | Use case: Remove account | Add | Add “Register account” use case detail | 1.0 |
|  | Use case: View artistic service | Add | Add “View artistic service” use case detail | 1.0 |
|  | Use case: Reset password | Add | Add “Reset password” use case detail | 1.0 |
|  | Use case: Report creator | Add | Add “Report creator” use case detail | 1.0 |
|  | Use case: View premium package | Add | Add “View premium package” use case detail | 1.0 |
|  | Use case: View reported artwork | Add | Add “View reported artwork” use case detail | 1.0 |
|  | Use case: Add to cart | Add | Add “Add to cart” use case detail | 1.0 |
|  | Use case: Update package | Add | Add “Update package” use case detail | 1.0 |
|  | Use case: Report artwork | Add | Add “Report artwork” use case detail | 1.0 |
|  | Use case: Update package | Add | Add “Update package” use case detail | 1.0 |
|  | Use case: Upgrade to creator | Add | Add “Upgrade to creator” use case detail | 1.0 |
|  | Use case: View cart | Add | Add “View cart” use case detail | 1.0 |
|  | Use case: View artist | Add | Add “View artist” use case detail | 1.0 |
|  | Use case: Search artwork | Add | Add “Search artwork” use case detail | 1.0 |

**SIGNATURE PAGE**

**ORIGINATOR:** <Name> <Date>

<Position>

**REVIEWERS:** <Name> <Date>

<Position>

<Name, if it’s needed> <Date>

<Position>

**APPROVAL:** <Name> <Date>

<Position>

**TABLE OF CONTENTS**

[**1. Introduction 6**](#_1fob9te)

[1.1. Purpose 6](#_rp197lq7j5e3)

[1.2. Scope 6](#_2et92p0)

[1.3. Definitions, Acronyms, and Abbreviations 6](#_tyjcwt)

[1.4. References 6](#_3dy6vkm)

[1.5. Overview 7](#_1t3h5sf)

[**2. Overall Description 8**](#_4d34og8)

[**3. FUNCTIONAL REQUIREMENTS 10**](#_2b5kx9wa1rxx)

[3.1. Use Cases Diagram 10](#_17dp8vu)

[3.2. Login 11](#_yl84cm7f0cl2)

[3.3. Sign Up 14](#_3ujcc0pdomn)

[3.4. Pay by Stripe 17](#_ti8rlnuhp18o)

[3.5. View profile 19](#_zae60i44pyti)

[3.6. Edit profile 21](#_wx7uk0v19iq9)

[3.7. View artworks 23](#_xhdihl1kw0wn)

[3.8. View artwork detail 25](#_g4mv2ppluxnr)

[3.9. Add to cart 27](#_ajubcslvk71o)

[3.10. View list of artworks 29](#_8vfpv1jgfa84)

[3.11. Remove artwork 31](#_hyclbxhyualq)

[3.12. Upload artwork 33](#_utws0z16nls5)

[3.13. View list of user accounts 35](#_fz6q7k2o8md0)

[3.14. Remove account 37](#_47w2vikldsdu)

[3.15. View artistic service 39](#_gynn84tvvdyr)

[3.16. Reset password 41](#_k4gzmuog7xvs)

[3.17. Report creator 43](#_rgk58w6eqszl)

[3.18. Report artwork 45](#_xjz24pyins1s)

[3.19. View premium package 46](#_yrfreyhqg8gh)

[3.20. View Reported Artwork 48](#_ncbhx1uwcwua)

[3.21. View Reported User 49](#_ub95au4oaxrc)

[3.22. Update package 50](#_bkyf543k9i5e)

[3.23. View artist 52](#_uq55afmt6wde)

[3.24. Edit information of artworks 54](#_17j4cxlyjhi)

[3.25. Upgrade to creator 55](#_26in1rg)

[3.26. View cart 57](#_3h0po4201tzo)

[3.27. Upgrade to creator 59](#_pa26cbo6nhgb)

[3.28. Provide drawing service 60](#_y7xszutfgngm)

[3.29. Search artwork 62](#_8ia1p8ejwssw)

[**4. NON-FUNCTIONAL Requirements 64**](#_lnxbz9)

[4.1. Usability 64](#_35nkun2)

[4.2. Reliability 64](#_1ksv4uv)

[4.3. Performance 64](#_44sinio)

# **Introduction**

Creating a wonderful platform for people who are interested in art, named **Artwork Sharing Platform**. Our program is designed to upload, interact, sell and purchase artwork created by artists and provide artistic services for users using the platform.

## ***Purpose***

This document represents the Software Requirements Specification (SRS) of the **Artwork Sharing Platform.** Its purpose is to describe the scope, both the functional and non-functional software requirements, as well as the design constraints of the whole logic sub-component. Furthermore, this document shows how the system's interfaces are designed in detail.

## ***Scope***

Create a web-based platform where artists can share and sell their artwork. Creators pay a fee to register, upload their art, and manage profiles. Buyers can browse and purchase art directly from the platform. The system includes features like user authentication, artwork management, a marketplace, sales tracking, and an admin panel. Monetization options include creator fees, transaction fees, and premium features. Legal considerations cover terms of use, privacy policies, and copyright guidelines.

## ***Definitions, Acronyms, and Abbreviations***

Stripe : Stripe is a platform that enables online and in-person payment processing and provides financial solutions for businesses of all sizes. It offers a suite of APIs and no-code tools to accept payments, send payouts, and automate financial processes.

API : An Application Programming Interface (API) serves as a bridge that allows two or more computer programs or components to communicate with each other.

Premium package: is a service that allows customers to pay a fee to become a creator.

## ***References***

*1.4.1. Website:*

* https://www.ibm.com/docs/en/rational-soft-arch/9.6.1?topic=diagrams-use-case
* www.wikipedia.org

*1.4.2 Books:*

* Software Engineering Seventh Edition Ian Sommerville.
* Software Engineering Ed.2 by Jalota & Pankaj.

## ***Overview***

The Software Requirements Specification (SRS) serves as a guide for software development, outlining requirements and objectives. It covers the overall working and value of the SRS, including development costs and time estimates. The document summarizes the art-sharing platform’s purpose, highlights its main objectives, and emphasizes user objectives and characteristics. Functional requirements specify system behavior, input-output relationships, and valid data ranges, while interface requirements describe communication with other components and users.

# **Overall Description**

**2.1 Guest Requirements**

* Guests, individuals who have not yet authenticated into the platform, have limited access to certain functions:
* Sign Up: Guests can register for an account on the platform.
* Log In: Guests can log in to their existing accounts.
* View Artwork: Guests can browse and explore artwork created by Creators.
* View Artwork Detail: Customers can view an artwork in detail.
* Search: Guests can search for artwork based on genre or keywords.

**2.2. Customer Requirements**

* *Customers, users authenticated with the "Customer" role, have access to a variety of functionalities tailored to enhance their experience on the platform:*
* *View Artwork: Customers can browse and explore artwork created by Creators.*
* *Search: Customers can search for artwork based on genre or keywords.*
* *View Artwork Detail: Customers can inspect artwork in detail, exploring its intricacies.*
* *Purchase Premium Artwork: Customers can buy premium artwork available on the platform.*
* *Interact: Customers can engage with artwork by leaving comments and liking them, fostering community interaction.*
* *Propose Artistic Service: Customers can request specific artistic services from Creators, facilitating collaboration.*
* *Upgrade to Creator: Customers have the option to pay for upgrading their roles to Creator, enabling them to upload their artworks and participate more actively in the platform.*
* *Reset Password: Customers can reset their password to regain access to their account in case of forgetting or compromising it.*
* *View / Edit Profiles: Customers can view and edit their profiles, enabling them to manage their personal information and preferences efficiently.*

**2.3 Creator Requirements**

* Creators, users logged in with the "Creator" role, have specialized functionalities aimed at facilitating their role as artists and contributors to the platform:
* Publish Artwork: Creators can upload and publish their artwork for others to view.
* Sell Artwork: Creators can offer their artwork for sale on the platform.
* Offer Artistic Services: Creators can provide artistic services upon request from customers.
* *Reset Password: Creators can reset their password to regain access to their account in case of forgetting or compromising it.*
* *View / Edit Profiles: Creators can view and edit their profiles, enabling them to manage their personal information and preferences efficiently.*
* *Purchase Premium Artwork: Creators can buy premium artwork available on the platform.*

**2.4 Admin Requirements**

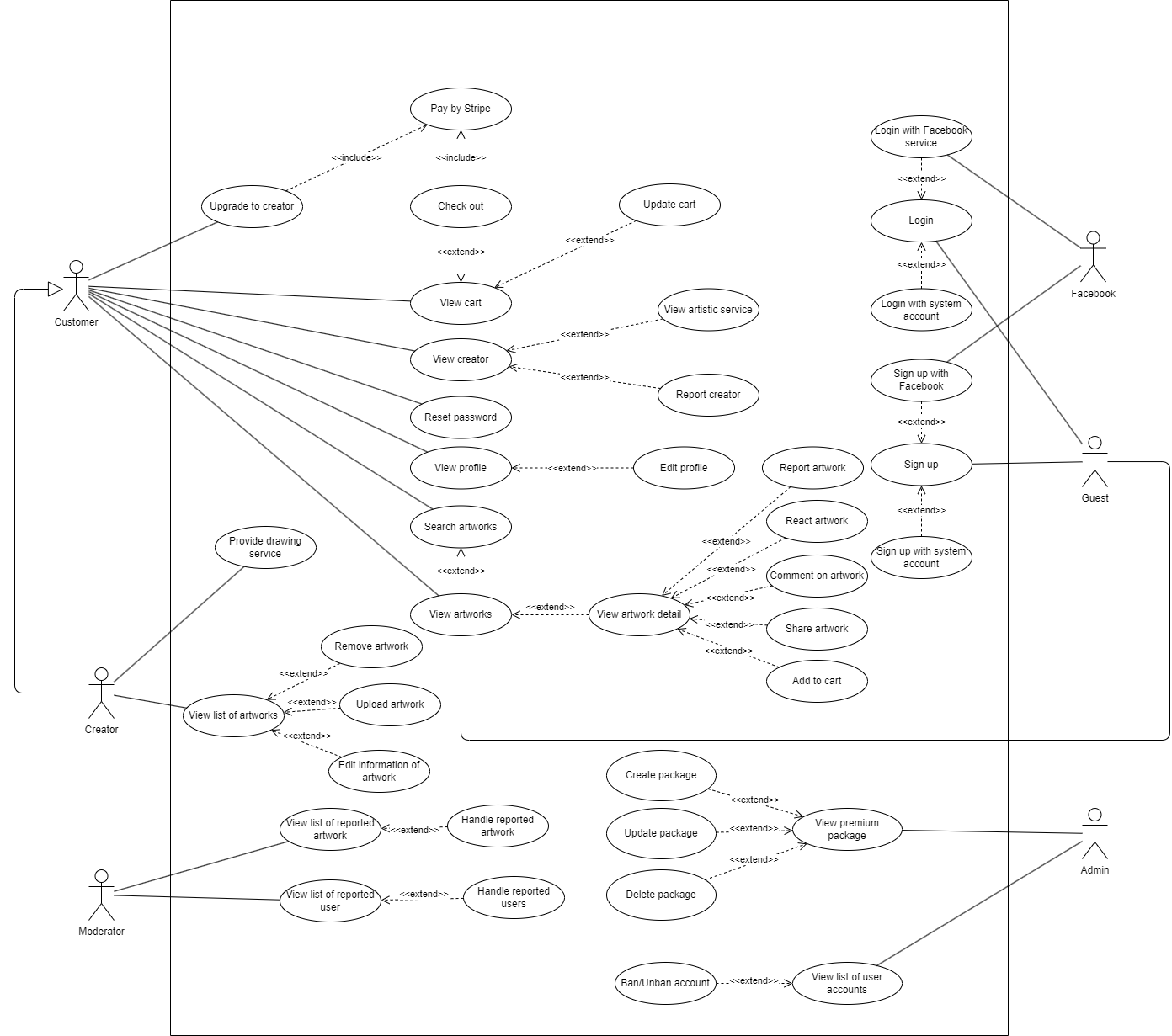
* Administrators, users with administrative privileges, have access to functionalities necessary for managing and overseeing the platform:
* *View Premium Package: Administrators can create, modify, and remove premium packages, adjusting their features as needed.*
* *View List of User Accounts: Administrators can oversee user accounts and efficiently remove them as necessary.*

**2.5 Moderator Requirements**

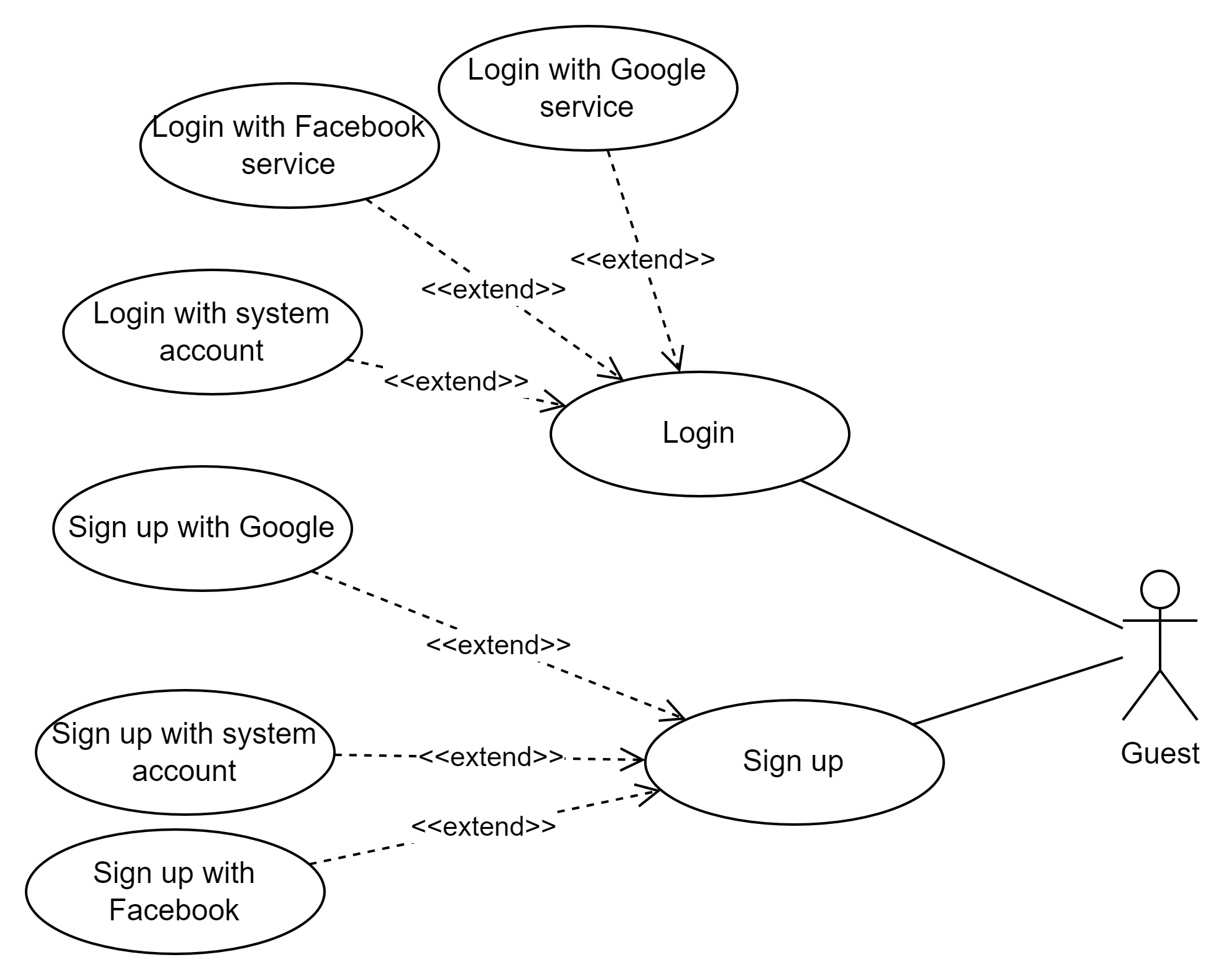
* Moderators, designated users responsible for maintaining a positive and safe environment on the platform, have specific duties and permissions:
* Content Moderation: Moderators oversee content uploaded by Creators and monitor interactions between users.
* Address Complaints: Moderators handle complaints and reports submitted by users regarding content or behavior on the platform.

# **FUNCTIONAL REQUIREMENTS**

## ***Use Cases Diagram***

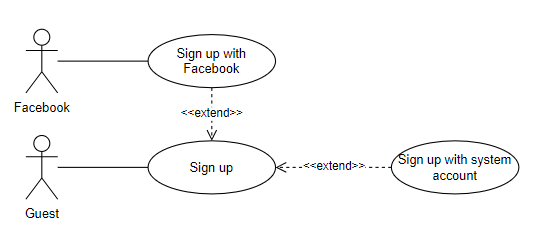
**

## ***Login***



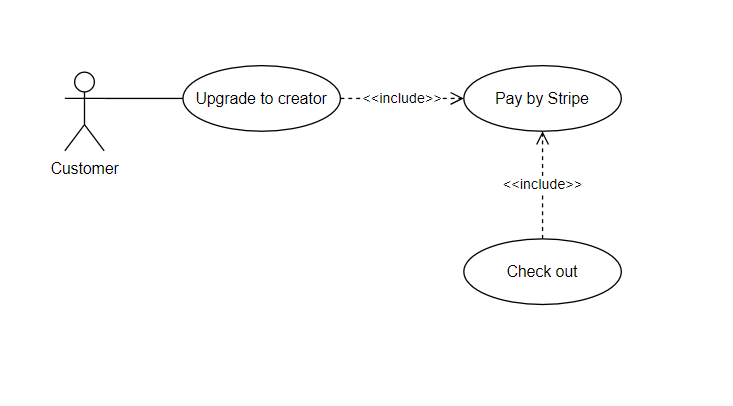
| **USE CASE-1 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC001 | **Use-case Version** | | <1.0> |
| **Use-case Name** | **Login** | | | |
| **Author** | Nguyen Huu Minh Tai | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Guest.*  **Summary:**  *This use case describes the process of guests who want to log in to the Web Application.*  **Goal:**  *This feature allows the actor (Guest) to login to the Web Application.*  **Triggers**  *The actor (Guest) wants to log in to a Web Application.*  **Preconditions:**  *The actor has a Facebook Account, Google Account or an account registered before.*  *The account used to log in must be authorized to access the system. The account must not be banned by admin.*  **Post Conditions:**  ***Success:*** *The system must inform “Login successfully!”, and the actor will be redirected to the Homepage of the application. The system must record user activity in the system log.*  ***Failed:*** *If a user provides incorrect login information, the system must provide an appropriate error message and disallow access to the system. And require the actor to log in again.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *On the home page, the actor clicks the button “Login” on the header or view detailed artwork.* | *The system redirects to the login page and shows a form to log in with these fields:* ***Email****: text input, required, check email format.*  ***Password****: password input, not empty, and at least 8 characters.*  *Below the form is a button used to log in with Facebook.* | | *2* | *In the login page, the actor inputs email and passwords.*  *[Alternative 1]* |  | | *3* | *The actor clicks “Login”.* | *The system redirects to the home page.*  *In addition, the system will display an error if the correct format is not entered or the password is incorrect.*  *[Exception 1]*  *[Exception 2]* |   **Alternative Scenario:**  *1:*   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *In the home page, the actor clicks the button “Login” on the header or view detailed artwork.* | *The system redirects to the login page.* | | *2* | *In the login page, Actor clicks on the Facebook icon.* | *The system redirects to the Facebook interface and shows a confirmation message.* | | *3* | *Actor clicks the “Yes” button.* | *The system redirects to the home page if the account has previously logged in.*  *If you have never logged in before, the system will redirect you to the register page.* |   **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The email is not correct format* | *System informs error message to remind user inputs correct format.* | | *2* | *The email or password is not correct* | *System informs “email or password is incorrect”.* | |  | *Facebook account hasn’t been authenticated.* | *The system redirects to the Sign Up page.* |   **Relationships:**  *Login with Facebook service.*  *Login with a system account.*  **Business Rules:**   * *Require users to provide authentication information such as email and password to access their accounts.* * *Locks the account after five failed login attempts and requires the user to reset their password via email after the account is locked.* | | | | |

## ***Sign Up***



| **USE CASE-2 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC002 | **Use-case Version** | | <1.0> |
| **Use-case Name** | **Sign Up** | | | |
| **Author** | Nguyen Huu Minh Tai | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Guest.*  **Summary:**  *This use case describes the process of guests who want to sign up for an account in a Web Application.*  **Goal:**  *This feature allows the actor (Guest) to create a new account to become a customer or creator in the system.*  **Triggers**  *The actor (Guest) wants to register an account in the web application.*  **Preconditions:**  *The actor has a Facebook Account or an Email Account that has been authorized before.*  *The actor does not have an account.*  **Post Conditions:**  ***Success:*** *The system redirects to the home page. The system must record user activity in the system log.*  ***Failed:*** *The system shows an error message corresponding to the error code.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *On the home page, the actor clicks the button “Sign Up” on the header.* | *The system redirects to the sign-up page and shows a form with these fields:*  *Full Name: text input is not empty.*  *Email: text input is required, check the email format.*  *Female: select 2 options male or female.*  *Select Role: select 2 options customer or creator*  *Password/Confirm Password: two password inputs must be identical, not empty and at least 8 characters.* | | *2* | *Actor inputs all of the fields in the form sign-up.* | *The system will validate input data.* | | *3* | *The actor clicks “Next”.* | *The System sends to that registered email an activation code, then shows a new form with a new field - activation code: A string of numbers that has 6 characters.* | | *4* | *Actor inputs activation.* |  | | *3* | *Actor clicks “Sign Up”.* | *The system redirects to the home page.* |   **Alternative Scenario:**  *1:*   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *In the sign-up page, the actor clicks on the Facebook icon.* | *The system redirects to the Facebook interface and shows a confirmation message.* | | *2* | *The actor clicks the “Yes” button.* | *The system redirects to the home page.* |   **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The email and password are not in the correct format* | *System informs error message to remind the user must input the correct format.* | | *2* | *The email does not exist.* | *The system detects that the registered email does not exist, and then shows a warning error.* | | *3* | *The password and confirm password do not match.* | *The system informs “The password and confirmation password do not match”.* | | *4* | *The email already exists in the system.* | *The system displays a warning.* |   **Relationships:**  *Sign up with Facebook service.*  *Sign up with a system account.*  **Business Rules:**  *- Each account must have a unique email address.*  *- Multiple accounts cannot be registered with the same email address.*  *- Requirements may include minimum length, special characters, uppercase letters, lowercase letters, and numbers.*  *- Validate user input fields such as name, email, and password to ensure they are not empty and adhere to the correct format.* | | | | |

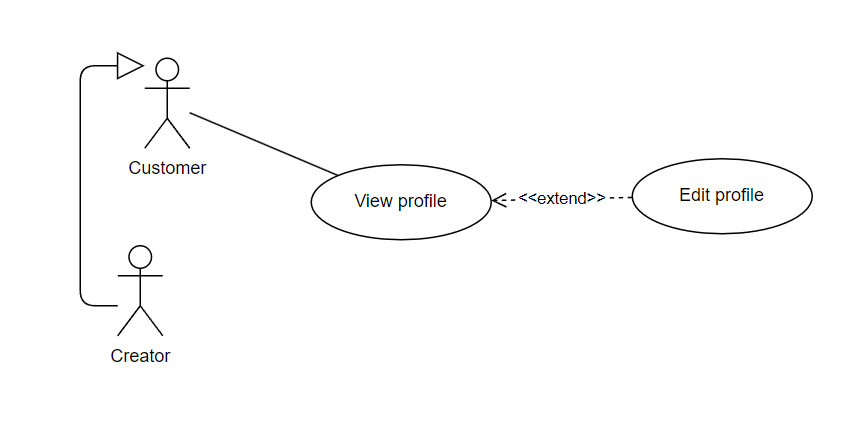
## ***Pay by Stripe***



| **USE CASE-3 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC003 | **Use-case Version** | | <1.0> |
| **Use-case Name** | **Pay by Stripe** | | | |
| **Author** | Nguyen Huu Minh Tai | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator and customer.*  **Summary:**  *This use case describes the process of the actor purchasing artwork on the artwork-sharing platform using the Stripe payment gateway.*  **Goal:**  *This feature allows the actor to easily and securely pay for items on the artwork-sharing platform using the Stripe payment gateway.*  **Triggers**  *The actor wants to pay money for services on the artwork-sharing platform.*  **Preconditions:**  *- The actor must be authenticated and logged into the system as a creator or customer.*  *- The checkout process must be successful.*  **Post Conditions:**  ***Success:*** *The actor receives confirmation of the successful payment and the stripe portal updates the payment status as "successful". Transaction details are recorded in the system of the web application. The checkout status is marked as purchased in the platform's database.*  ***Failed****: The actor receives a payment failure notification and the stripe portal updates the payment status as "failed". Transaction details are recorded in the web application system. The checkout status is returned as unpurchased in the platform's database.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. On the checkout page, the actor clicks the “Pay” button.* | *1. The system redirects to the Stripe Payment Gateway to process the payment and shows a form for payment with these fields: Email, card information, cardholder name, etc.* | | *2* | *2. The actor inputs all of the fields in the form of payment.* | *2. The system will validate input data.* | | *3* | *3. The actor clicks “Pay”.* | *3. The system redirects to the order confirmation page.* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The payment fails due to technical issues.* | *The system prompts the user to retry the payment or try again later.* | | *2* | *The payment fails due to insufficient funds* | *The actor is prompted to use an alternative payment method or add funds to their account* | | *3* | *The payment fails due to invalid card information* | *The actor is prompted to correct the card details and retry the payment.* |   **Relationships:**  *Upgrade to creator.*  *Check out.*  **Business Rules:**  *Users should receive immediate confirmation of successful payment, along with details of the transaction, price, and payment method.* | | | | |

## 

## ***View profile***



| **USE CASE-4 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC004 | **Use-case Version** | | <1.0> |
| **Use-case Name** | View profile | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator, Customer.*  **Summary:**  *This use case describes the process of creators or customers viewing their own profile information including name, phone, email, address, and gender.*  **Goal:**  *Creators or customers are able to view their own profiles.*  **Triggers**  *Creators or customers click on the “Profile” option on the header.*  **Preconditions:**  *The actor must be authenticated and logged into the system as either a customer or a creator.*  **Post Conditions:**   * *Creators or customers successfully view their own profile information.* * *The system successfully displays the actor’s profile information.*   **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1.* | *1. The actor selects the “Profile” option on the header.* | *1. The system redirects to the profile page showing personal information such as name, phone number, email, address, and gender.*  *[Exception 1]* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The system cannot access or find the user's profile information in the database.* | *1. The system displays an error message and instructs the user to try again later.* |   **Relationships:**  *Edit profile*  **Business Rules:**  *<Any concern about the business>* | | | | |
|  | | | | |

## 

## ***Edit profile***

## 

| **USE CASE-5 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC005 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Edit profile | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator, Customer.*  **Summary:**  *This use case describes the process of creators or customers editing their own profile information.*  **Goal:**  *Creators or customers are able to edit their own profiles.*  **Triggers**  *Creators or customers click on the “Edit Profile” option on the profile page.*  **Preconditions:**  *- The user must be authenticated and logged into the system as either a customer or a creator.*  *- The actor must be on the Profile Page.*  **Post Conditions:**  *- Creators or customers successfully edit their own profile information.*  *- Any changes made to the profile are reflected accurately upon return to the profile view.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The actor selects “Edit Profile” option* | *1. The system shows a form containing name, phone number, email, address, and gender.*  *- [Name]: Must not be empty.*  *- [Phone Number]: Must not be empty, start with 0 and have 10 digits.*  *- [Email]: the email format must be valid.*  *- [Address]: Must not be empty.*  *- [Gender]: Must be Male or Female* | | *2* | *2. The actor inputs the form and clicks the “Save” button* | *2. The system validates new information and saves it in the database. The system redirects to the profile page, showing the new information.*  *[Exception 1]*  *[Exception 2]* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The user inputs information in the wrong format.* | *1. The system redirects to the form and shows an appropriate error message.* | | *2* | *2. The actor clicks on the “Save” button with some missing fields.* | *The system displays an error under which the field has not been inputted.* |   **Relationships:**  *View profile*  **Business Rules:**  *- The email format must be valid.*  *- The phone number must start with 0 and have 10 digits. - All inputs are required.* | | | | |

## 

## ***View artworks***

## 

| **USE CASE-6 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC006 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View artworks | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | Normal | |
| **Actor:**  *Creator, Customer, Guest.*  **Summary:**  *This use case describes the process of creators or customers or guests viewing artworks on the homepage.*  **Goal:**  *The actor (Creator or Customer or Guest) is able to view artworks on the homepage.*  **Triggers**  *The actor (Creator or Customer or Guest) wants to view artworks.*  **Preconditions:**  *N/A*  **Post Conditions:**  *The systems successfully display the artworks on the homepage, showing only those artworks that have not been purchased.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. On the homepage, the actor clicks on the “Home” button.*  *[Alternative 1]* | *1. The system shows artworks that have not been purchased.*  *[Exception 1]* |   **Alternative Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The actor clicks on the logo of the website.* | *1. The system shows artworks that have not been purchased.*  *[Exception 1]* |   **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *- The result is empty.* | *The system displays an error message: “‘No data”.* |   **Relationships:**  *View artwork detail.*  **Business Rules:**   * *Each of the artwork must have a record in the storage and still be available.* * *All artwork information must be clear and up-to-date.* * *The artwork that has been bought will be hidden.* | | | | |
|  | | | | |

## 

## ***View artwork detail***

## 

| **USE CASE-7 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC007 | **Use-case Version** | | 1.0 |
| **Use-case Name** | View artwork detail | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | Normal | |
| **Actor:**  *Creator, Customer, Guest.*  **Summary:**  *This feature allows the actor (Creator or Customer or Guest) to view an artwork in detail.*  **Goal:**  *The actor (Creator or Customer or Guest) can view an artwork in detail.*  **Triggers**  *The actor (Creator or Customer or Guest) clicks on the “More” link in the artwork.*  **Preconditions:**  *The actor must be on the homepage or on an artwork detail page.*  **Post Conditions:**  *The system successfully displays the details of an artwork and like, share, and add to cart buttons and a comment field.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. After the actor clicks on the “More” option in the picture of the artwork on the homepage.*  *[Alternative 1]* | *1. The system redirects to the artwork detail page. The page displays the artwork's detailed information such as:*   * *The creator’s name.* * *Price.* * *Picture.* * *Description.* * *Category.*   *and shows like, share, report, view creator, add to cart buttons, and a comment field.*  *[Exception 1]* |   **Alternative Scenario:**  *1*   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. After the actor clicks on the “More” option in the picture of the different artwork on the artwork detail page in the “More Like This” section.* | *1. The system redirects to the artwork detail page. The page displays the artwork's detailed information such as:*   * *The creator’s name.* * *Price.* * *Picture.* * *Description.* * *Category.*   *and shows like, share, report, view creator, add to cart buttons, and a comment field.*  *[Exception 1]* |   **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *- The system cannot find the artwork detail in the database* | *The system displays an error message: “Artwork Detail not found”.* |   **Relationships:**  *- React artwork.*  *- Comment on artwork.*  *- Share artwork.*  *- Add to cart.*  *- Search artworks.*  **Business Rules:**  *<Any concern about the business>* | | | | |
|  | | | | |

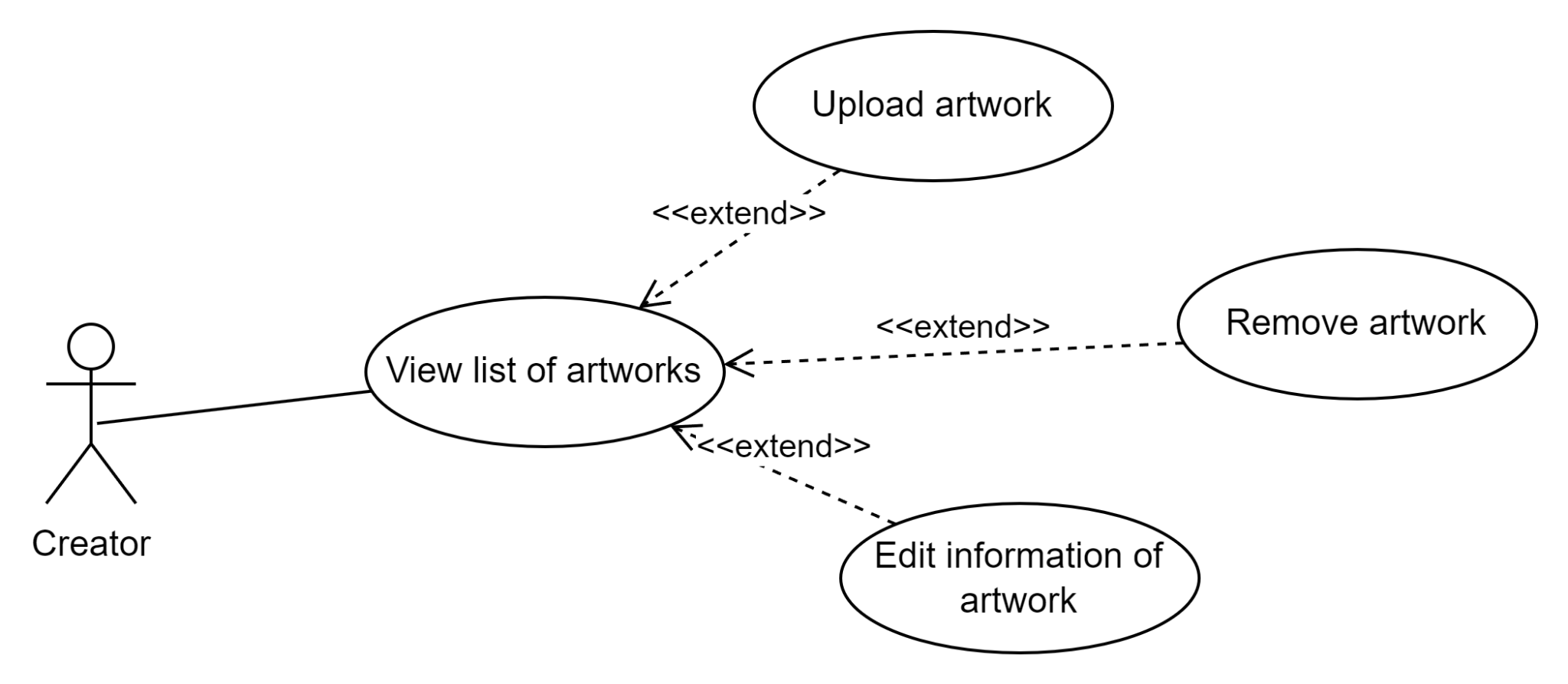
## 

## ***Add to cart***

## 

| **USE CASE-8 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC008 | **Use-case Version** | | <1.0> |
| **Use-case Name** | Add to cart | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator, Customer.*  **Summary:**  *This use case describes the process of creators or customers adding an artwork to the cart.*  **Goal:**  *The actor (Creator or Customer) can add artwork to the cart.*  **Triggers**  *The actor (Creator or Customer) clicks on the “Add to cart” button.*  **Preconditions:**  *- The actor must be authenticated and logged into the system as either a customer or a creator.*  *- The actor should be viewing details of an artwork.*  **Post Conditions:**  *The system saves the artwork to the cart and redirects to the homepage.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. In the artwork detail page, the actor clicks on the “Add to cart” button.* | *1. The system saves the artwork to the cart.* |   **Alternative Scenario:**  *N/A*  **Exceptions:**  *N/A*  **Relationships:**  *View artwork detail*  **Business Rules:**  *The artwork must be available in stock. Creator can add his/her own artwork to his/her cart.* | | | | |
|  | | | | |

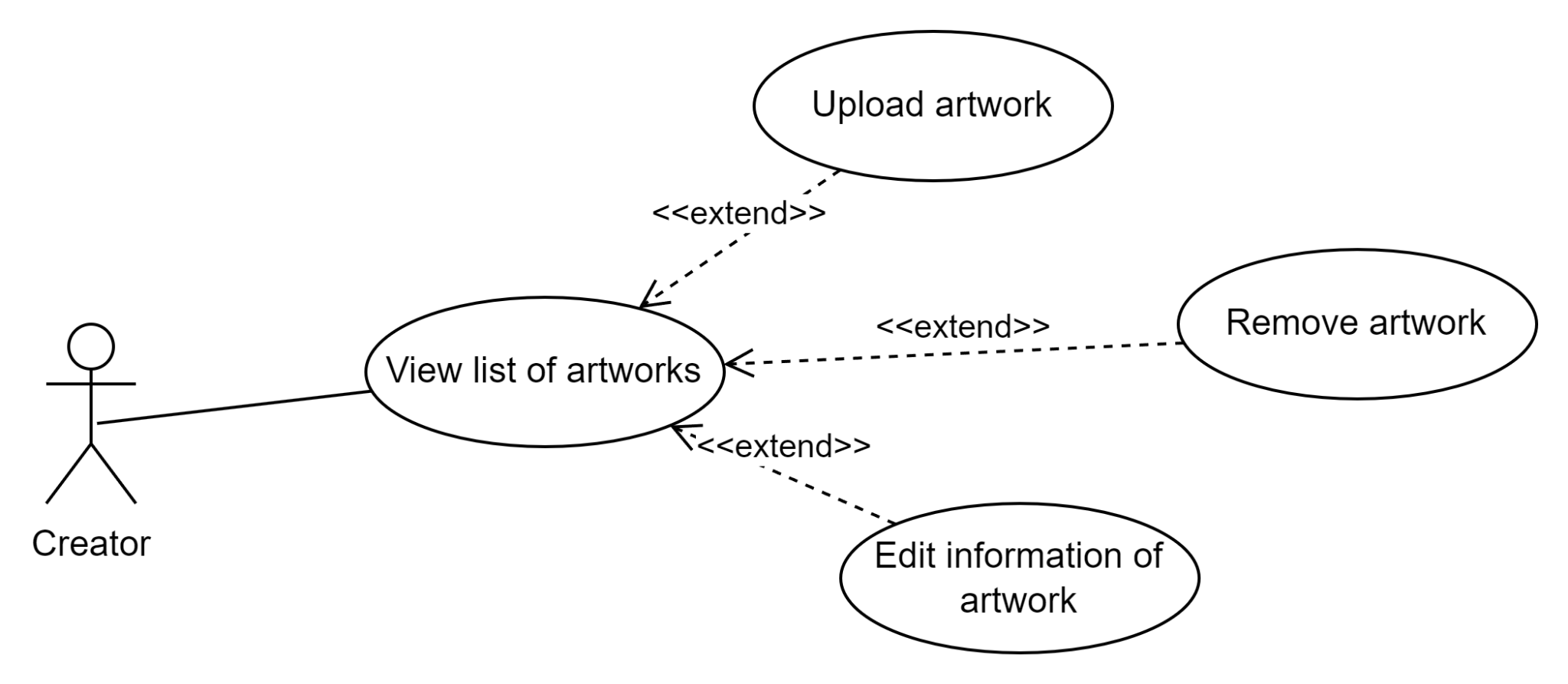
## ***View list of artworks***



| **USE CASE-9 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC009 | **Use-case Version** | | <1.0> |
| **Use-case Name** | View list of artworks | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator.*  **Summary:**  *This use case describes the process of creators viewing the list of artworks on the artwork list page.*  **Goal:**  *The Creator is able to view the list of artworks on the artwork list page.*  **Triggers**  *The Creator clicks on the “Manage Artworks” option on the header.*  **Preconditions:**  *- The user must be authenticated and logged into the system as a creator.*  *- The creator must have the permission to view the list of artworks.*  **Post Conditions:**  *The system successfully displays a list of artworks on the “Artwork List” page, providing options to upload new artwork as well as edit and delete the existing ones, with corresponding buttons for each artwork entry.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The actor clicks on the “Manage Artwork” option on the homepage.* | *1. The system shows the list of artworks including:  - Artwork ID.*  *- Title.*  *- Description.*  *- Image.*  *- Creator Name.  Additionally, there are buttons for editing, deleting, creating an artwork.* |   **Alternative Scenario:**  *N/A*  **Exceptions:**  *N/A*  **Relationships:**   * *Remove artwork* * *Upload artwork* * *Edit information of artwork.*   **Business Rules:** | | | | |

## 

## ***Remove artwork***



| **USE CASE-10 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC0010 | **Use-case Version** | | <1.0> |
| **Use-case Name** | Remove artwork | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator.*  **Summary:**  *This use case describes the process of the creator removing an artwork from the artwork list.*  **Goal:**  *The creator can remove an artwork.*  **Triggers**  *The creator clicks on the “Delete” button in a corresponding artwork.*  **Preconditions:**  *- The actor must be authenticated and logged into the system as a creator.*  *- The actor must be on the Manage Artworks page.*  **Post Conditions:**   * *The system removes the artwork from the database and redirects to the Manage Artworks page.* * *Any changes made to the list of artworks are reflected accurately upon return to the Manage Artworks page.*   **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. On the Manage Artwork page, the actor clicks the Delete button corresponding to the artwork.* | *1. The systems will show a popup box with the message: “Do you want to delete this artwork ?” and two buttons: “Yes / No”.* | | *2* | *2. The actor clicks on the “Yes” button.* | *2. The system removes the artwork from the database and redirects to the Manage Artwork page with the changes applied.*  *[Exception 1]* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The artwork has already been bought.* | *The system displays an error message: “Artwork has been bought, cannot delete!”, then redirecting to the Manage Artwork.* |   **Relationships:**  *View the list of artworks.*  **Business Rules:**  *The creator cannot remove the artwork that has been bought.* | | | | |

## 

## 

## ***Upload artwork***

## 

| **USE CASE-11 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC0011 | **Use-case Version** | | <1.0> |
| **Use-case Name** | Upload artwork | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Creator.*  **Summary:**  *This use case describes the process of the creator uploading an artwork to display on the homepage.*  **Goal:**  *The creator can upload an artwork.*  **Triggers**  *The creator clicks the “Upload an artwork” button.*  **Preconditions:**  *- The actor must be authenticated and logged into the system as a creator.*  *- The actor must be on the Manage Artworks page.*  **Post Conditions:**  *The system adds the artwork to the database and displays it on the homepage.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The actor clicks on the “Upload an artwork” button on the Manage Artworks page.* | *1. The system displays a form for adding information about the artwork, including title, description, picture, price, creator's name, a select box for "isPremium," and buttons for "Create" and "Back to List."*  *- [Title] : Must not be empty.*  *- [Description] : Must be more than 50 letters.*  *- [Picture] : Must be a JPG file.*  *- [Price]: Must be a number.*  *- [Creator’s name]: Must not be empty.* | | *2* | *2. The actor inputs the artwork including title, description, price, …* | *2. The system validates the information.*  *[Exception 1]* | | *3* | *3. The actor clicks on the “Image File” to choose the picture on the personal computer.* | *3. The system validates the image file.*  *[Exception 2]* | | *4* | *4. The actor clicks on the “Create” button* | *4. The system saves the information into the database and returns to the Manage Artworks Page.*  *[Exception 3] [Exception 4]* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The actor inputs the information in the wrong format.* | *The system displays an error message under the incorrectly formatted input.* | | *2* | *The actor chooses a file that is not an image.* | *The system displays an error message under the “Image File” button.* | | *3* | *The actor attempts to click "Create" with input that is incorrectly formatted, not an image, or both.* | *The system displays an error message under the incorrectly formatted input or the “Image File” button or both, prompting a redirection back to the form.* | | *4* | *The actor clicks on the “Create” button with some missing fields.* | *The system displays an error under which the field has not been inputted.* |   **Relationships:**  *N/A*  **Business Rules:**   * *The picture-choosing field must be the button to choose from a file or a personal computer.* * *All inputs are required.* * *The information format must be in valid:* * *Price must be a number.* * *Description must be more than 50 letters.* | | | | |

## 

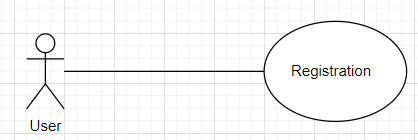
## ***View list of user accounts***

## 

| **USE CASE-13 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC012 | **Use-case Version** | | <1.0> |
| **Use-case Name** | View list of user accounts | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Admin.*  **Summary:**  *This use case describes the process of the admin viewing a list of user accounts in the system.*  **Goal:**  *The admin can view a list of user accounts.*  **Triggers**  *The admin clicks the “Manage Users” button.*  **Preconditions:**  *The actor must be authenticated and logged into the system as an admin.*  **Post Conditions:**  *The system shows a list of user accounts with name, phone number, email, and disabled icon.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. In the homepage, the actor clicks on the “Manage Users” option.* | *1. System redirects to the Manage Users page and shows a list of user accounts with name, phone number, email, and disabled icon.*  *[Exception 1]* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The system cannot find the user account in the database* | *The system displays an error: “User account not found”.* |   **Relationships:**  *Remove account*  **Business Rules:**  *<Any concern about the business>* | | | | |

## 

## ***Registration account***



## 

| **USE CASE-14 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC013 | **Use-case Version** | |  |
| **Use-case Name** | Registration account | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Normal User.*  **Summary:**  *This use case describes the process of a normal user registering for a new account.*  **Goal:**  *The user can register for a new account.*  **Triggers**  *The user clicks the “Register” button.*  **Preconditions:**  *The actor must access the registration page.*  **Post Conditions:**  *- The system creates a new user account with the provided information.*  *- The newly registered user can log in to the system.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The user navigates to the registration page. 1. System displays the registration form.* | *1.The user fills out the required information and submits the registration form. 2. System validates the information and creates a new user account. Inform: “Registration successful”; [Exception 1]* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The user tries to register an account with incomplete or invalid information.* | *The system displays an error message indicating the required fields or validation errors.* |   **Relationships: N/A**  *View list of user accounts.*  **Business Rules:**   * *The user must provide valid information for the new user account to be registered successfully.* | | | | |

## 

## ***Login account***

## 

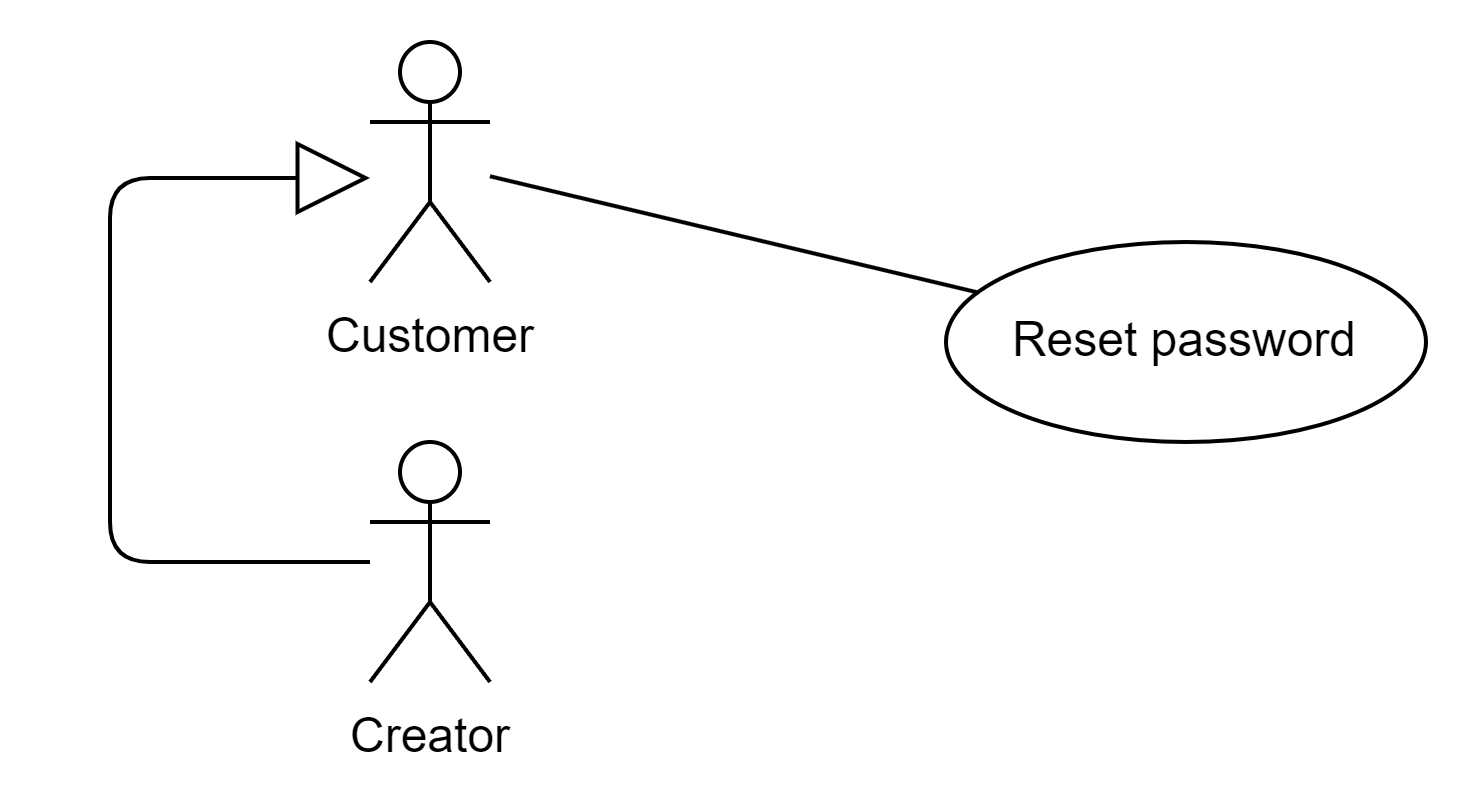
| **USE CASE-15 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC014 | **Use-case Version** | |  |
| **Use-case Name** | Login account | | | |
| **Author** | Dang Cong Hung | | | |
| **Date** | 25/02/2024 | **Priority** | High | |
| **Actor:**  *Normal User.*  **Summary:**  *This use case describes the process of a normal user logging into their account.*  **Goal:**  *The user can log in to their account.*  **Triggers**  *The user clicks the “Login” button.*  **Preconditions:**  *- - The actor must have the account page.*  **Post Conditions:**  *- The system grants access to the user account if credentials are valid.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *1. The user navigates to the login page.*  *2. The user enters their username and password and submits the form.* | *1. System displays the login form.*  *2. System validates the credentials. If valid, the user is logged in. If invalid, the system displays an error message.* |   **Alternative Scenario:**  *N/A*  **Exceptions:**     | ***No*** | ***Cause*** | ***System Response*** | | --- | --- | --- | | *1* | *The user login an account.* | *The system displays an error message: ”Name or password is wrong”.* |   **Relationships:**  *View user accounts.*  **Business Rules:**   * *The user must provide valid credentials to successfully log in to their account.* | | | | |

## ***View artistic service***

## 

| **USE CASE-14 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC014 | | | | **Use-case Version** | | | | 1.0 | | | |
| **Use-case Name** | View artistic service | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 05/02/2024 | | | | **Priority** | | | | High | | | |
| **Actor:**  *Customer, creator.*  **Summary:**  *This use case describes that the creator can view and create a new service, which the creator can provide for the customer.*  **Goal:**  *As a customer, I want to view the service so that I can request from the artist.*  **Triggers**  *The customer clicks on the “View service” button in the artist profile in the web application.*  **Preconditions:**  *The user must log in with the Customer role.*  **Post Conditions:**   * *The user waits for the response from the artist.* * *The user can make an order after having an approved response from the artist.*   **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user clicks the “View service” button in the artist’s profile* | *The system displays the list with the name of the service and selection box* | | *2* | *The user selects the service and submit.* | *The system sends the request to the artist.* |   **Alternative Scenario:**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user submits without choosing the service.* | *The system displays the error to remind the user to choose before submitting.* |   **Relationships:**  View Artist  **Business Rules:**  *N/A* | | | | | | | | | | | | |

## ***Reset password***



| **USE CASE-15 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC015 | | | | **Use-case Version** | | | | 1.0 | | | |
| **Use-case Name** | Reset password | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 05/02/2024 | | | | **Priority** | | | | High | | | |
| **Actor:**  *Customer, creator.*  **Summary:**  *To allow users to securely reset their forgotten passwords and regain access to their accounts.*  **Goal:**  *As a customer or creator, I want to reset my password to regain access to the web application when I forget my password.*  **Triggers**  *The user clicks "Forgot Password" on the login page.*  **Preconditions:**  *The user must have previously registered an account on the system.*  **Post Conditions:**  *The user successfully resets their password and gains access to their account.*  **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user clicks the "Forgot Password" option on the login page.* | *The system displays the "Forgot Password" form. The fields must not be empty.*   * *[Email]: Should follow a valid email format* * *[New password]: contain an uppercase character, lowercase character, a digit, and a non-alphanumeric character. Passwords must be at least six characters long.* * *[Confirm new password]: Must match the new password field.* | | *2* | *The user enters and submits their email address* | *The system validates the email address. The system generates a unique reset link and sends it to the user's email.* | | *3* | *The User receives the reset email and clicks on the reset link.* | *The system directs the user to the password reset page.* | | *4* | *The user enters a new password and confirms it.* | *The system validates and updates the password for the user's account and allows the customer to log in with the new password.* |   **Alternative Scenario:**  *N/A*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user enters and submits the wrong email address* | *The system displays the error and demands the user input again.* | | *2* | *The new passwords are not available.* | *The system shows the message to alert the user.* | | *3* | *The confirm password in the “Reset password” page is not the same as the password.* | *The system shows the message to alert the user.* |   **Relationships:**  *N/A*  **Business Rules:**   * *Password reset links should expire after a certain time (e.g., 24 hours) to ensure security.* * *Users may be limited to a certain number of password reset attempts within a specific timeframe to prevent abuse.* | | | | | | | | | | | | |

## 

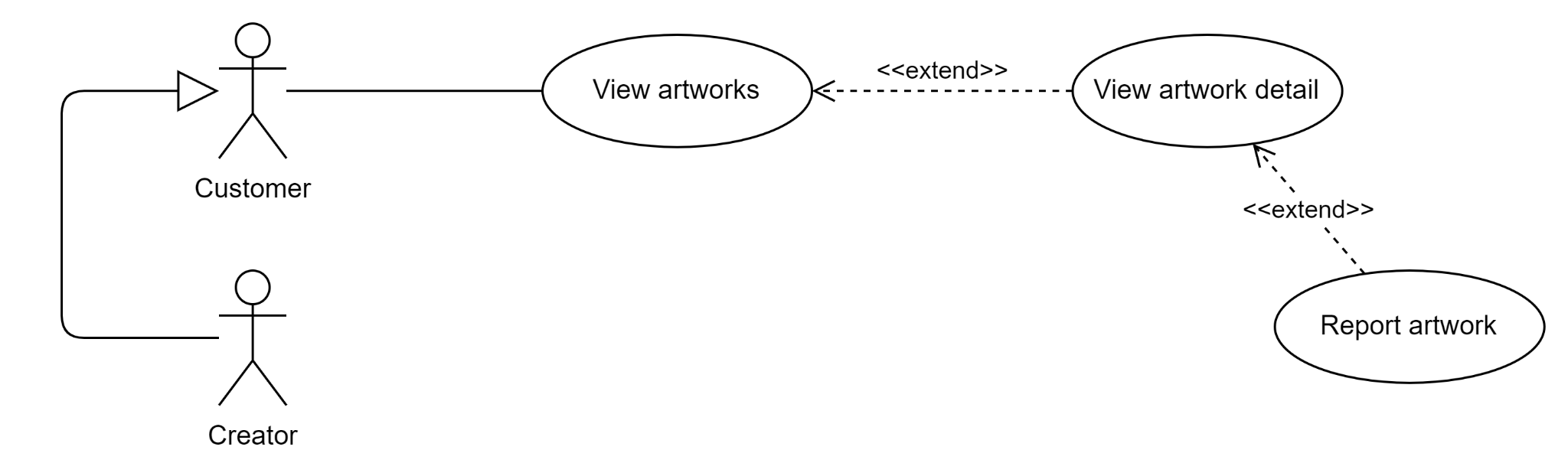
## ***Report creator***

## 

| **USE CASE-16 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC016 | | | | **Use-case Version** | | | | 1.0 | | | |
| **Use-case Name** | Report creator | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 05/02/2024 | | | | **Priority** | | | | High | | | |
| **Actor:**  *Customer, creator.*  **Summary:**  *This use case describes when a customer or creator reports the creator.*  **Goal:**  *As a customer or creator, I want to report the creator.*  **Triggers**  *The customer or creator clicks on the "Reports" button in the creator page.*  **Preconditions:**  *The user must be on the creator page.*  **Post Conditions:**  *The user successfully resets their password and gains access to their account.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Customer or Creator clicks on the report creator button.* | *The system saves the information of reported artwork in the database and notifies the successful report.* |   **Alternative Scenario:**  *N/A*  **Exceptions**  *N/A.*  **Relationships:**  *N/A*  **Business Rules:**   * *The function must be carried out by the creator or customer.* * *The information on the reported creator must be saved in the database.* | | | | | | | | | | | | |

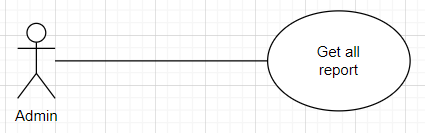
## 

## ***Report artwork***



| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC017 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Report artwork | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Customer, Creator.*  **Summary:**  *This use case describes when a customer or creator reports the artwork.*  **Goal:**  *As a customer or creator, I want to report the artwork in the detail page of it*  **Triggers**  *The customer or creator clicks on the "Reports" button in the artwork detail page.*  **Preconditions**  *The user must be on the detail page of the artwork.*  **Post Conditions:**  *The information of the reported artwork must be shown on the “View Reported Artwork” page of the moderator role.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Customer or Creator clicks on the report artwork button.* | *The system saves the information of reported artwork in the database and notifies the successful report.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *N/A.*  **Relationships**  *View artwork detail*  **Business Rules:**   * *The function must be carried out by the creator or customer.* * *The information on the reported artwork must be saved in the database.* | | | | | | | | | | | | |

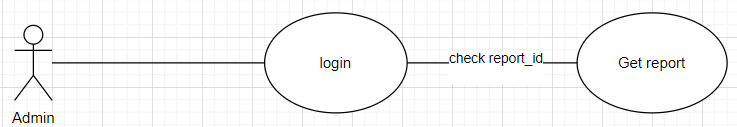
## ***Get all report***



| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC017 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Get all reports | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin, Creator.*  **Summary:**  *This use case describes the process of retrieving all reports made by customers or creators.*  **Goal:**  *As a moderator, I want to access all reports of artwork submitted by customers or creators.*  **Triggers**  *The moderator navigates to the "View Reported Artwork" page.*  **Preconditions**  *The moderator must be logged into the system.*  **Post Conditions:**  *The moderator can view all reported artwork and their details.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Moderator navigates to the "View Reported Artwork" page.* | *The system retrieves and displays all reported artwork along with their details.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *N/A.*  **Relationships**  *View reported detail*  **Business Rules:**   * *Only moderators can access the "View Reported Artwork" page.* * *All reported artwork and their details must be retrievable from the database.* | | | | | | | | | | | | |

## 

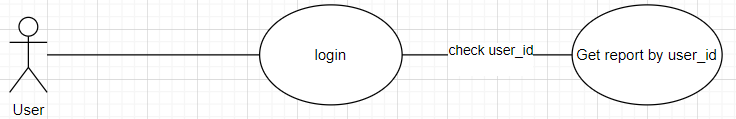
* 1. ***Get report by id***

******

| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC018 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Get report by ID | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin, Creator.*  **Summary:**  *This use case describes the process of retrieving a specific report of artwork by its unique identifier.*  **Goal:**  *As a moderator, I want to access a report of artwork by its ID.*  **Triggers**  *The moderator initiates a search for a specific report by its ID.*  **Preconditions**  *The moderator must be logged into the system.*  **Post Conditions:**  *The moderator can view the details of the specific reported artwork identified by its ID.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Moderator enters the ID of the report into the system.* | *The system retrieves and displays the details of the reported artwork corresponding to the entered ID.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *If the entered ID does not match any existing report, the system displays an error message indicating the report was not found.*  **Relationships**  *View reported detail by report\_id*  **Business Rules:**   * *Only moderators can access the functionality to retrieve a report by its ID.* * *The system must accurately retrieve the details of the reported artwork corresponding to the entered ID from the database.* | | | | | | | | | | | | |

## 

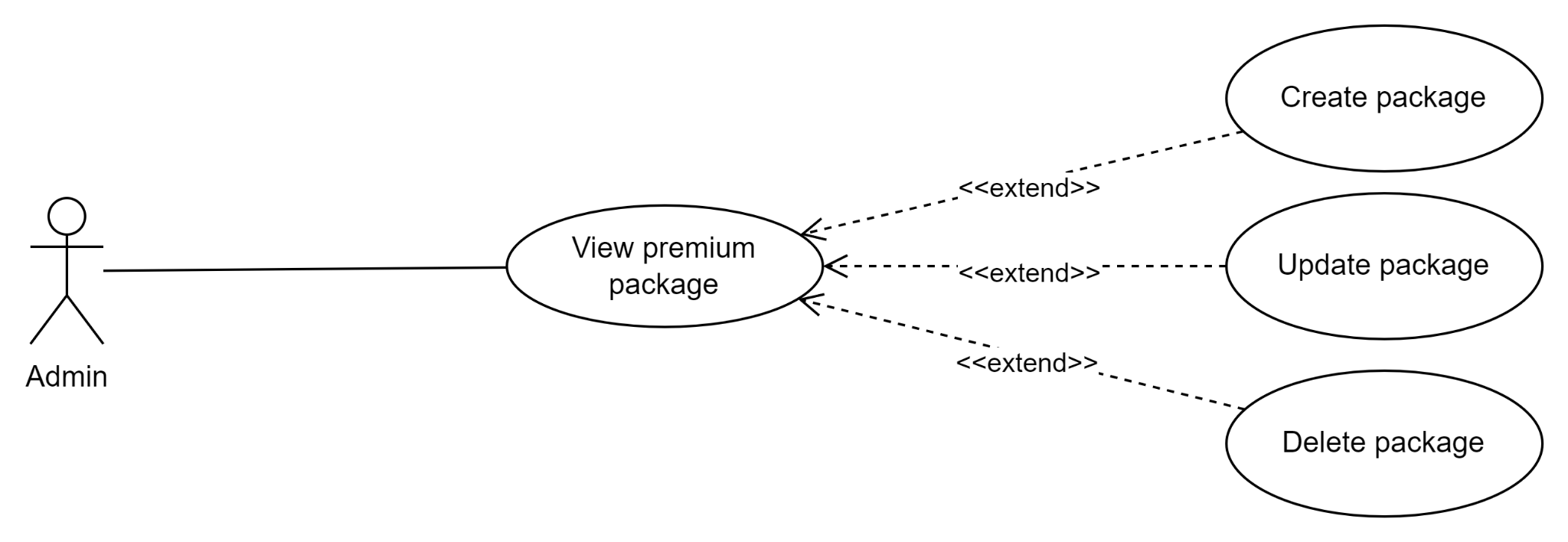
* 1. ***Get report by user id***

******

| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC019 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Get report by User ID | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin, Creator.*  **Summary:**  *This use case describes the process of retrieving all reports of artwork submitted by a specific user identified by their User ID.*  **Goal:**  *As a moderator, I want to access all reports of artwork submitted by a specific user identified by their User ID.*  **Triggers**  *The moderator initiates a search for reports submitted by a specific user by entering their User ID.*  **Preconditions**  *The moderator must be logged into the system.*  **Post Conditions:**  *The moderator can view all reports of artwork submitted by the specified user.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Moderator enters the User ID of the user whose reports they want to retrieve.* | *The system retrieves and displays all reports of artwork submitted by the specified user.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *If there are no reports submitted by the specified user, the system displays a message indicating no reports found for that user.*  **Relationships**  *View reported detail by id of user*  **Business Rules:**   * *Only moderators can access the functionality to retrieve reports by User ID.* * *The system must accurately retrieve all reports of artwork submitted by the specified user from the database.* | | | | | | | | | | | | |

## 

## ***View premium package***



| **USE CASE-18 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC020 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | View premium package | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin*  **Summary:**  *This method describes how the admin can view and modify the package.*  **Goal:**  *As an admin, I want to view and modify the premium package for the user in the web application*  **Triggers**  *The user activates the method possessed by selecting the “View package” option on the navigation.*  **Preconditions**  *The user must have logged in with an admin role account on the system previously*  **Post Conditions:**   * *The admin can view the overall package on the site.* * *The admin can choose the package to edit its information.*   **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The admin clicks “View package”.* | *The system displays the package list and its information on the site.*  *[Package name], [Price], [Amount artwork], [Description], [Edit button]* |   **Alternative Scenario**  *N/A.*  **Exceptions**  *N/A.*  **Relationships**  *Update package.*  **Business Rules:**  *Only the admin can view the list of the package.* | | | | | | | | | | | | |

## 

## ***View Reported Artwork***



| **USE CASE-19 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC021 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | View Reported Artwork | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Moderator*  **Summary:**  *This use case describes the process of a moderator viewing reports about artwork by a user*  **Goal:**  *As a moderator, I want to see the list of reported artwork on the application.*  **Triggers**  *The user activates the handle reported artwork possessed by clicking the “Reported Artwork” link.*  **Preconditions**  *The user must have previously logged in with a moderator account on the system.*  **Post Conditions:**  *The artwork deleted by the moderator cannot appear in the application.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The Moderator clicks the “Report Artwork” button.* | *The system displays a list of reported artworks in the app on the website.*  *[Artwork Id], [Artwork title], [Artist Name], [Hide artwork button]* |   **Alternative Scenario**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The Moderator removes the post/user that another moderator is deleting.* | *The system displays an error message.* |   **Relationships**  *View report.*  **Business Rules:**   * *Only the moderator can handle the reported post.* * *The removed artwork cannot appear in the application.* | | | | | | | | | | | | |

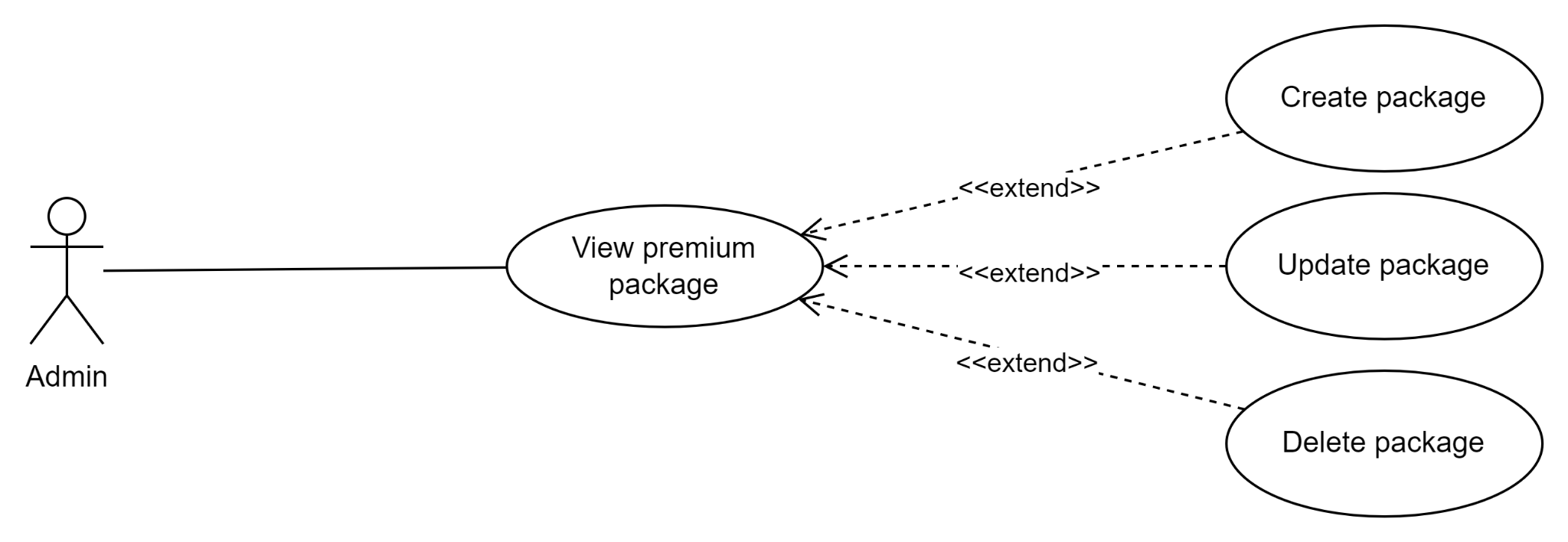
## ***View Reported User***



| **USE CASE-20 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC022 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | View Reported User | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Moderator*  **Summary:**  *This use case describes the process of a moderator viewing reports about the reported user by a user*  **Goal:**  *As a moderator, I want to see the list of reported users on the application.*  **Triggers**  *The moderator clicks the “Reported User” to activate this function.*  **Preconditions**  *The user must have previously logged in with a moderator account on the system.*  **Post Conditions**  *Displays a list of reported users. The administrator can click a button to handle the reported user*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The Moderator clicks the “Reported User” button.* | *The system displays a list of reported users in the app on the website.*  *[Artist Id], [Artist Name], [Ban artist button]* |   **Alternative Scenario**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The Moderator removes the post/user that another moderator is deleting.* | *The system displays an error message.* |   **Relationships**  *View report.*  **Business Rules:**  *Only the moderator can handle the reported post.* | | | | | | | | | | | | |

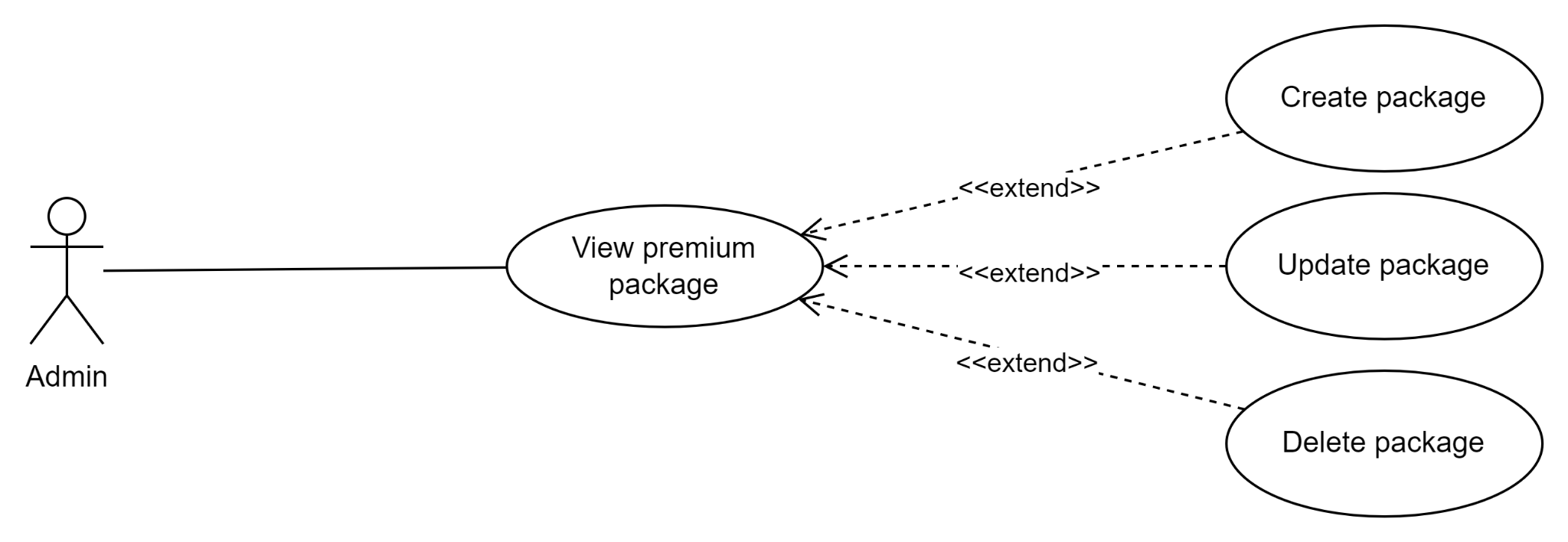
## 

## ***Update package***



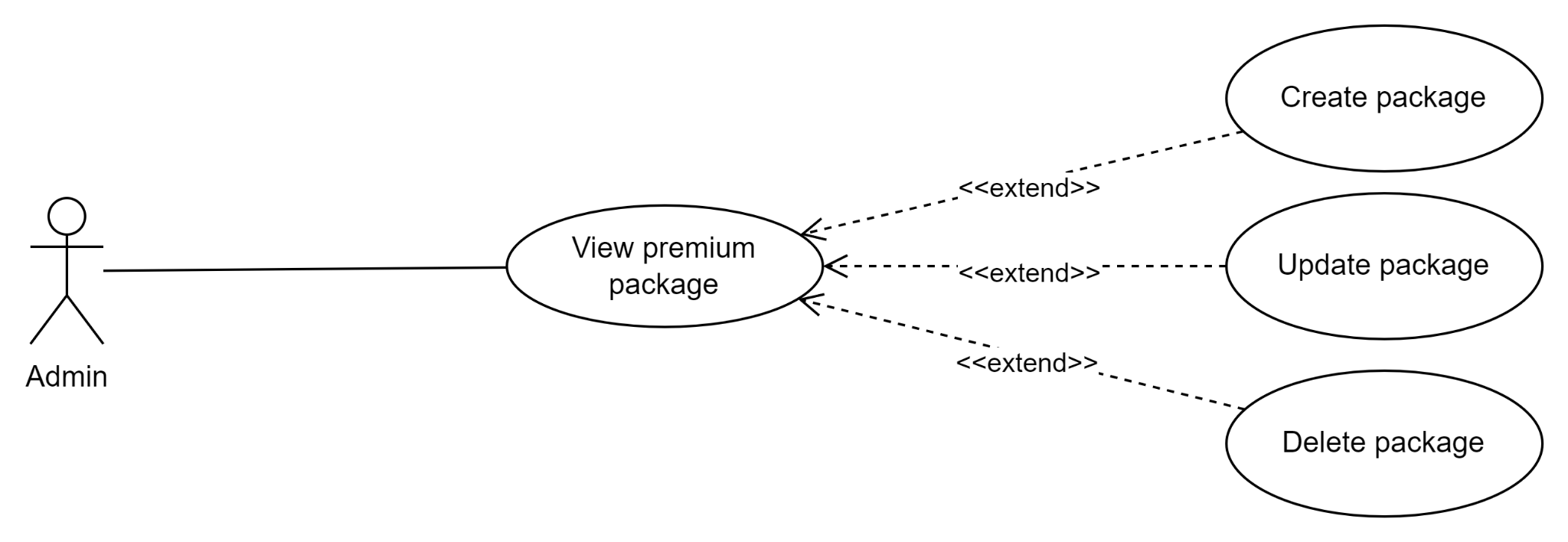
| **USE CASE-21 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC023 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Update package | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin*  **Summary:**  *This use case describes how the admin can update the package in the application.*  **Goal:**  *As an admin, I want to update the premium package in the web application*  **Triggers**  *The admin clicks the “Update package” to activate this function.*  **Preconditions**  *The user must have previously logged in with an admin account on the system.*  **Post Conditions**  *Packages updated by the administrator will display and apply the new information.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The admin clicks “Update package”* | *The system displays the current information of the package being clicked by the admin on the website.*   * *[Package Name]: have 200 characters for max length.* * *[Description]: no validation* * *[Price]: should be a positive numeric value.* * *[Amount artwork]: should be a positive integer.* | | *2* | *The admin changes the information of the package.* | *The system verifies the new information of the package being entered by the admin.* | | *3* | *The admin clicks submit the new information.* | *The system updates the new information to the database.* |   **Alternative Scenario**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user inputs invalid information in the form.* | *The system displays the error and pretends the user saved the new information.* |   **Relationships**  *View premium package.*  **Business Rules:**   * *The new information in the package must be valid.* * *Only the admin can update the package.* | | | | | | | | | | | | |

## ***Create package***



| **USE CASE-22 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC024 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Create package | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin*  **Summary:**  *This use case describes how the admin can create the package in the application.*  **Goal:**  *As an admin, I want to create a premium package in the web application.*  **Triggers**  *The admin clicks the “Create package” to activate this function.*  **Preconditions**  *The user must have previously logged in with an admin account on the system.*  **Post Conditions**  *Packages updated by the administrator will display and apply the new information.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The admin clicks “Create package”* | *The system displays the form to fill in the ìnormation of the new package on the website.*   * *[Package Name]: have 200 characters for max length.* * *[Description]: no validation* * *[Price]: should be a positive numeric value.* * *[Amount artwork]: should be a positive integer.* | | *2* | *The admin changes the information of the package.* | *The system verifies the new information of the package being entered by the admin.* | | *3* | *The admin clicks submit the new information.* | *The system updates the new information to the database.* |   **Alternative Scenario**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user inputs invalid information in the form.* | *The system displays the error and pretends the user saved the new information.* |   **Relationships**  *View premium package.*  **Business Rules:**   * *The new information in the package must be valid.* * *Only the admin can update the package.* | | | | | | | | | | | | |

## ***Delete Package***



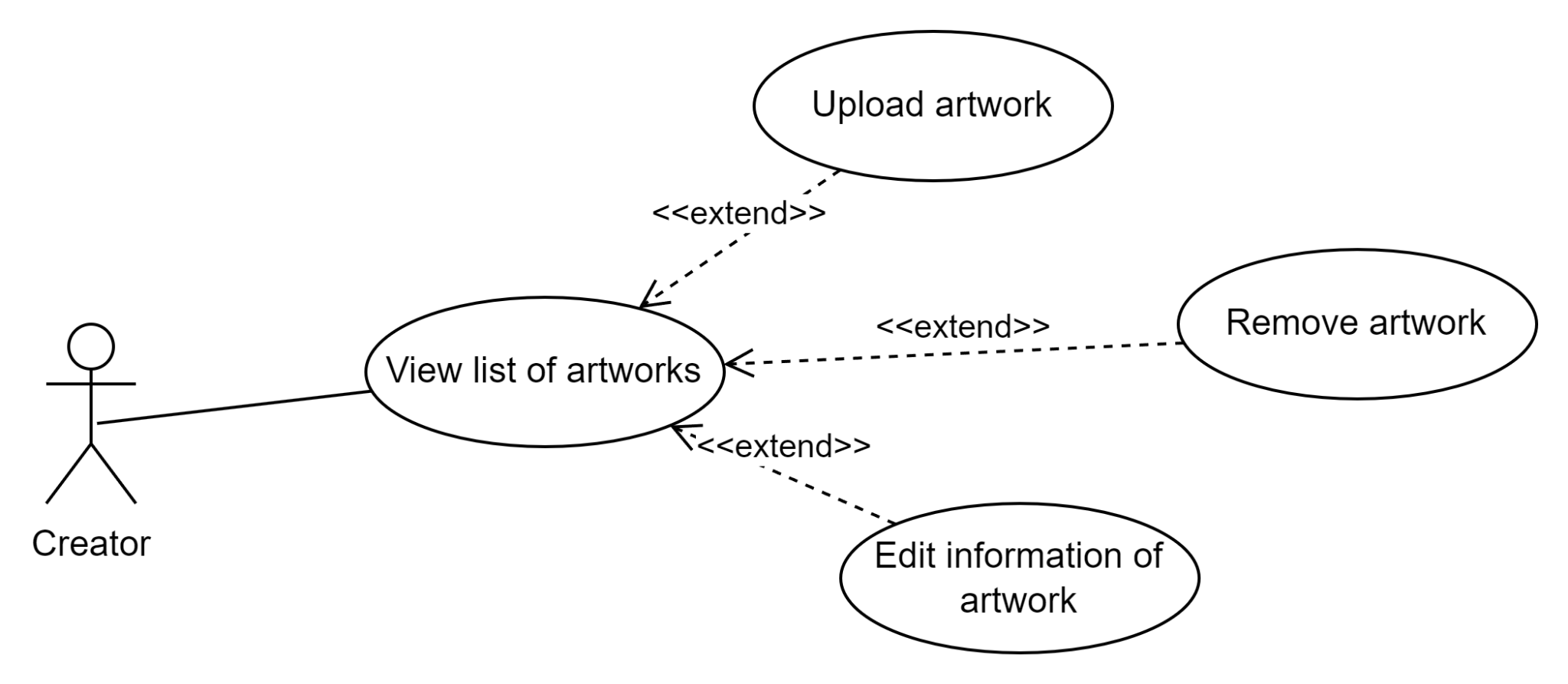
| **USE CASE-23 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC023 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Delete package | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin*  **Summary:**  *This use case describes how the admin can delete the package in the application.*  **Goal:**  *As an admin, I want to delete the premium package in the web application*  **Triggers**  *The admin clicks the “Delete package” to activate this function.*  **Preconditions**   * *The user must have previously logged in with an admin account on the system.* * *The user must be on the View package page.*   **Post Conditions**  *Packages deleted by the administrator will not display on the application.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The admin clicks “Update package”* | *The system asks the user to confirm the process to delete the package from the database.* | | *2* | *The admin clicks “Yes” to confirm.* | *The system deletes the package from the database and shows the successful notification.* |   **Alternative Scenario**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user selects “No” when the system confirms.* | *The system still keeps the database of the package.* |   **Relationships**  *View premium package.*  **Business Rules:**  *Only the admin can update the package.* | | | | | | | | | | | | |

## ***View creator***

## 

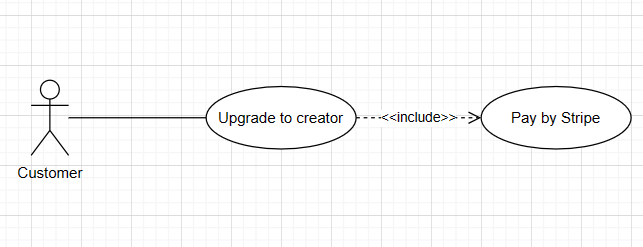
| **USE CASE-24 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC024 | **Use-case Version** | | <1.0> |
| **Use-case Name** | View Creator | | | |
| **Author** | Phạm Lê Quý Anh | | | |
| **Date** | 01/02/2024 | **Priority** | Normal | |
| **Actor:**  *Customer*  **Summary:**  *This use case describes the process of a customer viewing the artists’ profile of a particular artwork on this platform.*  **Triggers**  *The customer wants to see the artist’s information.*  **Preconditions:**  *The customer must be logged into their account (appropriate authorization)*  **Post Conditions:**  *The artist’s profile is displayed successfully to the user.*  **Main Success Scenario:**   | **Step** | **Actor action** | **System response** | | --- | --- | --- | | *1* | *User logs into the system and directs to the homepage.* | *The system displays the homepage including artworks, cart, search, home, add to cart* | | *2* | *The customer accesses to the artwork detail page* | *The system displays the artwork’s details including its title, creator, description* | | *3* | *The customer click to the artist’s name on artwork detail page* | *The system displays the artist’s profile including their own information and artworks.* |   **Alternative Scenario:**  *N/A*  **Exceptions:**  *The platform cannot connect with the server/ core system (due to network malfunction/ technical issues), the system displays an error message.*  **Relationships:**  *The “View creator” extends to “View artistic service”, “Report creator” and “Follow/ Unfollow creator”*  **Business Rules:**  *N/A* | | | | |

## ***Edit information of artworks***



| **USE CASE-25 SPECIFICATION** | | | | |
| --- | --- | --- | --- | --- |
| **Use-case No.** | UC025 | **Use-case Version** | | 1.0 |
| **Use-case Name** | Edit information of artworks | | | |
| **Author** | Phạm Lê Quý Anh | | | |
| **Date** | 01/02/2024 | **Priority** | Normal | |
| **Actor:**  *Creator*  **Summary:**  *This use case describes the process of a creator editing the detailed information of their particular artwork on this platform.*  **Triggers**  *The information of the artwork changes, and the creators want to edit it*  **Preconditions:**   1. *The creator logs into the system with the role “Creator” (appropriate authorization)* 2. *The creator has already uploaded artwork.*   **Post Conditions:**  *The artwork’s information is edited*  **Main Success Scenario:**   | **Step** | **Actor action** | **System response** | | --- | --- | --- | | *1* | *The creator accesses his detail page* | *System display creator’s detail page* | | *2* | *The creator clicks to the artwork they want to change detail.* | *System displays the artwork’s edit profile form* | | *3* | *The creator edits artwork’s information* | *System changes the information of artwork* | | *4* | *The creators save new information of the artwork* | *System save the new information* |   **Alternative Scenario:**  *N/A*  **Exceptions:**  *The platform cannot connect with the server/ core system (due to network malfunction/ technical issues), the system displays an error message.*  **Relationships:**  *The “View list of artworks” extends to the “Edit information of artworks*  **Business Rules:**  *N/A* | | | | |

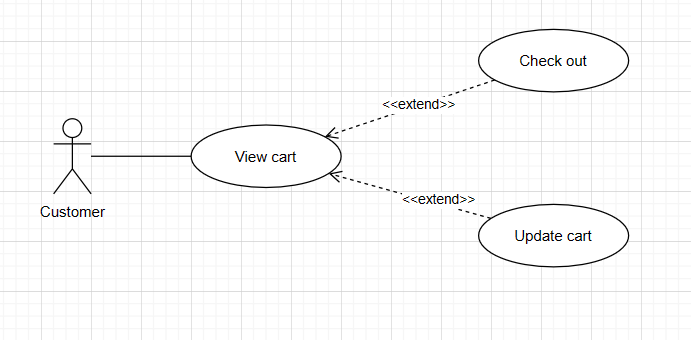
## ***Upgrade to creator***



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 26 | | |
| **Use Case Name:** | Upgrade to creator | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | High |
| **Actors:** | Customer | | |
| **Summary:** | The Customer wants to become creator in order to sell, and upload their artworks | | |
| **Trigger:** | The user clicks on an upgrade premium button. | | |
| **Preconditions:** | The user is logged into their account on the app as a customer. | | |
| **Post–conditions:** | After clicking on the upgrade button, user can fill in upgrading form, and redirect to the payment page(Stripe) | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users click on upgrade premium button | The system show the the price, what things creator can do, the button to redirect to payment page and cancel button(back to homepage) | | 2 | Users clicks on the payment button | The system redirects to payment pages.. Change the upgrading status to “Paying” | | 3 | Users fill in payment form in payment pages(Stripe) | The system will receive “success payment” status from Stripe pages. Change the role of customer to creator then update the status of payment to “Paid” and show the notification(“You are now a Creator !!!”) to customers. | | | | |
| **Alternative Flows:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 2.1 | User clicks on cancel button | The system redirects to homepage | | 3.1 | User do not complete or cancel the payment | The system receives “unsuccess payment” status from Stripe pages. Change the status of payment to “Fail”. Redirect users to the homepage. | | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on upgrade premium button | The system displays an error message and notifies “Please try again!!!” | | 2 | User receive an error when trying to click on pay button | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**  *- Pay by Stripe* | | | |
| **Business Rules:** | 1. The website must store the paying history of the customers. 2. Users must select valid payment methods provided by Stripe. 3. Users must receive an email after paying successfully. | | |

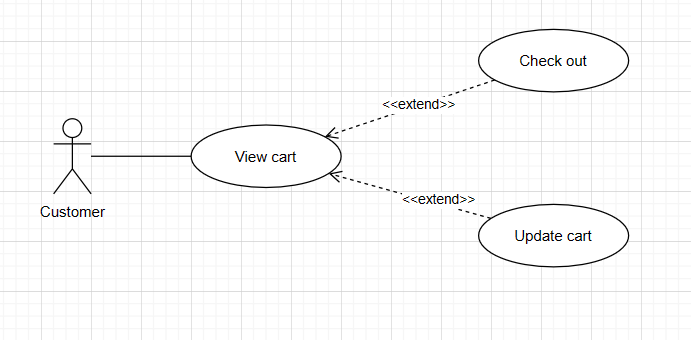
## 

## ***View cart***



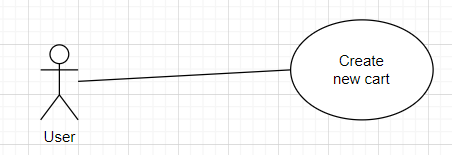
| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 27 | | |
| **Use Case Name:** | View cart | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | High |
| **Actors:** | Customer | | |
| **Summary:** | The Customer wants to see/update all the artworks they have added to the shopping cart. | | |
| **Trigger:** | The user clicks on a cart button. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | Show all the available artworks in the shopping cart and allow users to update. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users click on the cart button. | The system redirects to view cart pages.  This page must show the basic information such as:   * name of artwork, price, numbers of artworks, total money; * the check out/update button;   Allow users to remove artworks from the cart. | | 2 | Users click on the check out button.[Exception2] | The system redirects and sends all artworks in cart to checkout pages. | | 3 | Users click on the update button.[Exception1] | The viewcart page must show the latest update. | | | | |
| **Alternative Flows:** | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on the update button. | The system displays an error message and notifies “Please try again!!!” | | 2 | Users receive an error when trying to click on the checkout button. | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**   * *The base use case “View cart” has extension use cases : “Check out” and “Update cart”.* | | | |
| **Business Rules:** | 1. If the total money is less than 1, do not show total money. 2. The money is shown only 2 number after decimal point and have the dollar sign($) at the end(ex: 22.12$) 3. Check out buttons only available if there is at least 1 artwork in the cart. | | |

## ***3.30. View cart by id***



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 27 | | |
| **Use Case Name:** | View cart | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | High |
| **Actors:** | Customer | | |
| **Summary:** | The Customer wants to view and manage the artworks they have added to the shopping cart by specifying the cart ID. | | |
| **Trigger:** | The user clicks on a cart button. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | Show all the available artworks in the shopping cart and allow users to update. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users click on the cart button. | The system redirects to view cart pages.  This page must show the basic information such as:   * name of artwork, price, numbers of artworks, total money; * the check out/update button;   Allow users to remove artworks from the cart. | | 2 | Users click on the check out button.[Exception2] | The system redirects and sends all artworks in cart to checkout pages. | | 3 | Users click on the update button.[Exception1] | The viewcart page must show the latest update. | | | | |
| **Alternative Flows:** | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on the update button. | The system displays an error message and notifies “Please try again!!!” | | 2 | Users receive an error when trying to click on the checkout button. | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**   * *The base use case “View cart by ID” has extension use cases: “Check out” and “Update cart”.* | | | |
| **Business Rules:** | 1. If the total money is less than 1, do not show total money. 2. The money is shown only 2 number after decimal point and have the dollar sign($) at the end(ex: 22.12$) 3. Check out buttons only available if there is at least 1 artwork in the cart. | | |

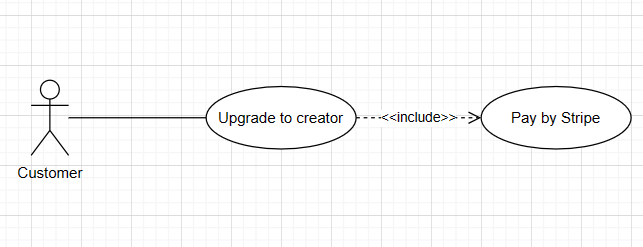
## 3.31. Create new cart



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 27 | | |
| **Use Case Name:** | Create new cart | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | High |
| **Actors:** | Customer | | |
| **Summary:** | The Customer wants to create a new shopping cart to add artworks to. | | |
| **Trigger:** | The user wants to start a new shopping session. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | A new shopping cart is created and ready for the user to add artworks. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | User wants to start a new shopping session. | The system creates a new empty cart for the user. The system notifies "New cart created successfully." | |  |  |  | |  |  |  | | | | |
| **Alternative Flows: N/A** | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | The system encounters an error while trying to create a new cart. | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**  *This use case is associated with the "Add to cart" use case, where users can add artworks to their shopping cart.*  *- The "View cart" use case allows users to view the contents of their shopping cart.*  *- The "Remove from cart" use case enables users to remove items from their shopping cart.* | | | |
| **Business Rules:** | 1. If the total money is less than 1, do not show total money. 2. The money is shown only 2 number after decimal point and have the dollar sign($) at the end(ex: 22.12$) 3. Check out buttons only available if there is at least 1 artwork in the cart. | | |

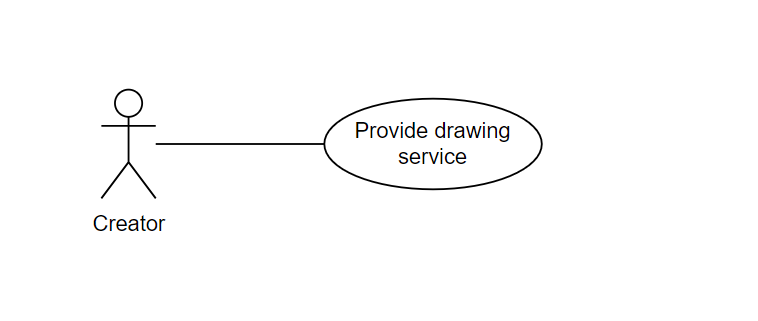
## 

## ***Upgrade to creator***



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 28 | | |
| **Use Case Name:** | Upgrade to creator | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | High |
| **Actors:** | Customer | | |
| **Summary:** | The Customer wants to become creator in order to sell, upload their artworks | | |
| **Trigger:** | The user clicks on an upgrade premium button. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | After clicking on the upgrade button, user can fill in upgrading form, and redirect to the payment page(Stripe) | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users click on upgrade premium button  [Exception1] | The system show:   * the the price * what things creator can do   the button to redirect to payment page and cancel button(back to homepage) | | 2 | Users clicks on the payment button  [Exception2] | The system redirects to payment pages. Change the upgrading status to “Paying” | | 3 | Users fill in payment form in payment pages(Stripe)  [Exception3] | The system will receive “success payment” status from Stripe pages. Change the role of customer to creator then update the status of payment to “Paid” and show the notification(“You are now a Creator !!!”) to customers. | | | | |
| **Alternative Flows:** | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on upgrade premium button | The system displays an error message and notifies “Please try again!!!” | | 2 | User receive an error when trying to click on pay button | The system displays an error message and notifies “Please try again!!!” | | 3 | User do not complete or cancel the payment | The system receives “unsuccess payment” status from Stripe pages. Change the status of payment to “Fail”. Redirect users to homepages. |     **Relationships:**   * A base use case called “Upgrade to creator” that has an inclusion use case called “Pay by Stripe”. | | | |
| **Business Rules:** | 1. The website must store the paying history of the customers. 2. Users must select valid payment methods provided by Stripe. 3. Users must receive an email after paying successfully. | | |

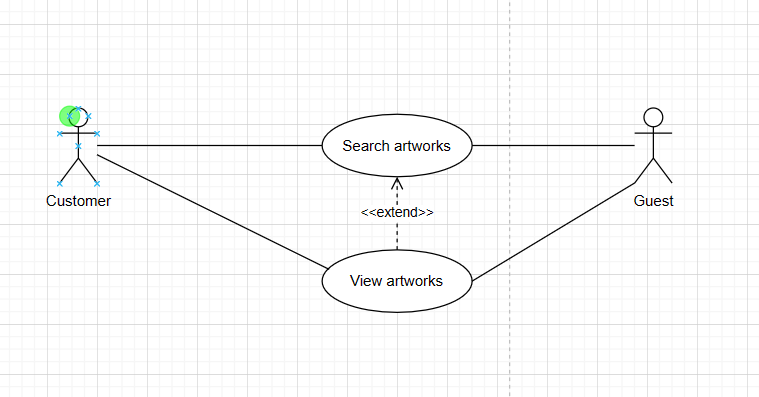
## ***Provide drawing service***



| **USE CASE-29 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC029 | | | | **Use-case Version** | | | | 1.0 | | | |
| **Use-case Name** | Provide drawing service | | | | | | | | | | | |
| **Author** | Nguyen Huu Minh Tai | | | | | | | | | | | |
| **Date** | 05/02/2024 | | | | **Priority** | | | | High | | | |
| **Actor:**  *Creator.*  **Summary:**  *This use case describes that the creator can modify the artwork drawing service listed for customer order service.*  **Goal:**  *As a creator, I want to modify the artwork drawing service listed so that I can modify it from the manager drawing service page.*  **Triggers**  *The creator wants to modify their service on the manager drawing service page.*  **Preconditions:**  *The user must log in with the creator role.*  **Post Conditions:**   * *The user waits for the response from the artist.* * *The user can make an order after having an approved response from the artist.*   **Main Success Scenario:**     | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user clicks the “Manager service” button in the artist’s profile* | *System redirects to the Manage Service page and shows a list of services with name, detail service, price, and a Remove button.*  *Below the table service is the button save.* | | *2* | *The user clicks the “Save” button.* | *The system redirects to the artist page.* |   **Alternative Scenario:**  *N/A.*  **Exceptions:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *The user input is the wrong format.* | *The system displays the error to remind the user.* |   **Relationships:**  *N/A*  **Business Rules:**  *N/A* | | | | | | | | | | | | |

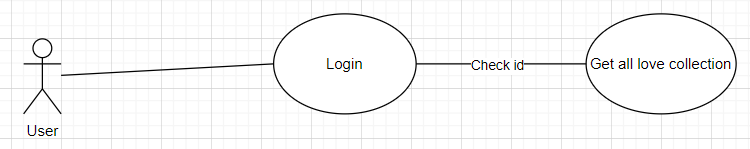
## 

## ***Search artwork***



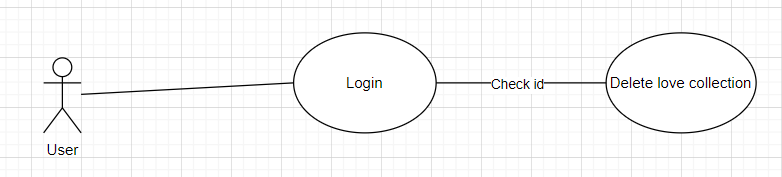
| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 30 | | |
| **Use Case Name:** | Search artworks | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | Normal |
| **Actors:** | Customer, Guest | | |
| **Summary:** | The Customer wants to search for artworks. | | |
| **Trigger:** | The user clicks on a search button. | | |
| **Preconditions:** |  | | |
| **Post–conditions:** | After clicking on the search button, the user can see the short information of the artwork. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users type in the searching box. | The system show:   * Name of related artworks * Creator of related artworks * Price of related artworks.   Allow users to click on artwork. | | 2 | Users click on the artwork. | The system redirects to the detail page of artworks. | | 3 | Users click on the search icon. | The system will show the page with all the artworks related to searching artworks. | | | | |
| **Alternative Flows:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1.1 | User type in invalid name(name of artworks that are not stored in database or name contains special characters) | The system show message “Can not found” | | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on upgrade premium button | The system displays an error message and notifies “Please try again!!!” | | 2 | User receive an error when trying to click on pay button | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**  *- the base use case “Search artworks” has an extension use case “View artworks”.* | | | |
| **Business Rules:** | 1. The search box allows 1-255 characters. 2. The search box does not allow special characters. 3. Only show a maximum 5-6 artworks for recommended results. | | |

# ***3.35.* Get all love collection by user\_id**



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 30 | | |
| **Use Case Name:** | Get love collection by user ID | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | Normal |
| **Actors:** | Customer, Guest | | |
| **Summary:** | The Customer wants to retrieve their favorite love collection by their user ID. | | |
| **Trigger:** | The user wants to view their favorite love collection. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | Display the details of the requested love collection. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | User requests to view their favorite love collection by their user ID. | The system retrieves the details of the user's favorite love collection based on their user ID. The system displays the details of the love collection. | | | | |
| **Alternative Flows:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1.1 | love collection does not exist because the user does not have one yet | The system show message “Can not found” | | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on upgrade premium button | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**  *- users can view their favorite love collections.* | | | |
| **Business Rules:** | 1. Love collections are collections marked as favorites by users. 2. - Users can retrieve details of their favorite love collection by providing their user ID. | | |

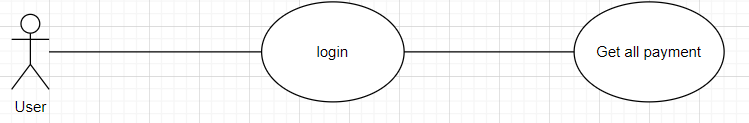
# ***3.36.*** Delete love collection by user\_id



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 30 | | |
| **Use Case Name:** | Delete love collection by user ID | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | Normal |
| **Actors:** | Customer, Guest | | |
| **Summary:** | The Customer wants to delete their favorite love collection by their user ID. | | |
| **Trigger:** | The user wants to remove their favorite love collection. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | The specified love collection is removed from the user's favorites. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | User requests to delete their favorite love collection by their user ID. | The system removes the specified love collection from the user's favorites. | | | | |
| **Alternative Flows:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1.1 | love collection does not exist because the user does not have one yet | The system show message “Can not found” | | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on delete love collection button | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**  *- This use case is associated with the "View love collection" use case, where users can view their favorite love collections.*  *- The "Add to love collection" use case enables users to mark artworks as favorites and add them to their love collection.*  *- The "Update love collection" use case allows users to modify their favorite love collections.* | | | |
| **Business Rules:** | 1. Love collections are collections marked as favorites by users. 2. - Users can retrieve details of their favorite love collection by providing their user ID. | | |

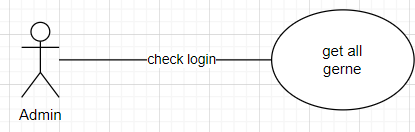
# 

***3.37. Get all payment***



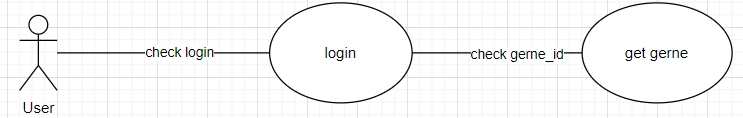
| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 30 | | |
| **Use Case Name:** | Get all payments | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | Normal |
| **Actors:** | Customer, Guest, Admin | | |
| **Summary:** | This use case describes the process of retrieving all payments made within the system. | | |
| **Trigger:** | As an administrator/customer, I want to access a list of all payments made. | | |
| **Preconditions:** | The administrator must be logged into the system. | | |
| **Post–conditions:** | The administrator can view a comprehensive list of all payments made within the system. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Administrator selects the option to view all payments. | The system retrieves and displays a list of all payments made within the system. | | | | |
| **Alternative Flows:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1.1 |  |  | | | | |
| **Exceptions: If there are no payments recorded in the system, the system displays a message indicating no payments found.**  **Relationships:**  *- View payment details* | | | |
| **Business Rules:** | 1. Only administrators can access the functionality to retrieve all payments. 2. The system must accurately retrieve and display all payment records from the database. | | |

# ***3.38. Get all gerne***



| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC017 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Get all gerne | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin, Creator.*  **Summary:**  *This use case describes the process of retrieving all gerne made by customers or creators.*  **Goal:**  *As a moderator, I want to access all gerne submitted by customers or creators.*  **Triggers**  *The moderator navigates to the "View gerne" page.*  **Preconditions**  *The moderator must be logged into the system.*  **Post Conditions:**  *The moderator can view all gerne and their details.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Moderator navigates to the "View Gerne" page.* | *The system retrieves and displays all gerne along with their details.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *N/A.*  **Relationships**  *View gerne detail*  **Business Rules:**   * *Only moderators can access the "View gerne" page.* * *All gerne and their details must be retrievable from the database.* | | | | | | | | | | | | |

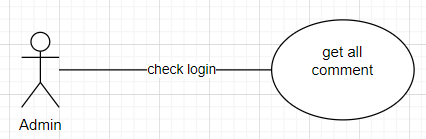
## ***3.39. Get gerne by id***



| **Use Case Specification** | | | |
| --- | --- | --- | --- |
| **Use Case No.:** | 27 | | |
| **Use Case Name:** | View cart | | |
| **Created By:** | Truong Nhat Hoang | | |
| **Date:** | 28/2/24 | **Priority:** | High |
| **Actors:** | Customer | | |
| **Summary:** | The Customer wants to view and manage the gerne by specifying the gerne ID. | | |
| **Trigger:** | The user clicks on a gerne button. | | |
| **Preconditions:** | The user is logged into their account on the app. | | |
| **Post–conditions:** | Show all the available gerne and allow users to update. | | |
| **Main Success Scenario/Main Flow/Normal Flow/Main Path:**   | **Step.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users click on the gerne button. | The system redirects to view gerne pages.  This page must show the basic information such as:   * name of gerne, price, numbers of gernes, total money; * the check out/update button;   Allow users to remove gerne | | | | |
| **Alternative Flows: N/A** | | | |
| **Exceptions:**   | **No.** | **Actor Action** | **System Response** | | --- | --- | --- | | 1 | Users receive an error when trying to click on the update button. | The system displays an error message and notifies “Please try again!!!” | | 2 | Users receive an error when trying to click on the checkout button. | The system displays an error message and notifies “Please try again!!!” |   **Relationships:**   * *The base use case “get gerne by id” has extension use cases: “Check out” and “Update gerne”.* | | | |
| **Business Rules:** | Genre IDs must be unique within the system.  Only authorized users or systems can request genre information by ID.  The system must ensure data integrity and accuracy when retrieving genre details by ID from the database. | | |

## 

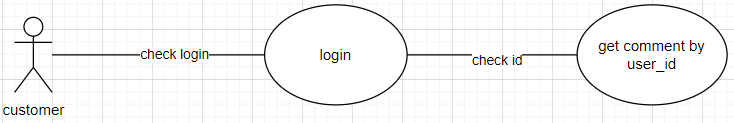
***3.40. Get all comment***

******

| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC017 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Get all comments | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Admin, Creator.*  **Summary:**  *This use case describes the process of retrieving all comments made by users, including customers and creators.*  **Goal:**  *As a moderator, I want to access all comments submitted by users for moderation purposes.*  **Triggers**  *The moderator navigates to the "View Comments" page.*  **Preconditions**  *The moderator must be logged into the system.*  **Post Conditions:**  *The moderator can view all comments and their details.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Moderator navigates to the "View Comments" page.* | *The system retrieves and displays all comments along with their details.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *N/A.*  **Relationships**  *View comment detail*  **Business Rules:**   * *Only moderators can access the "View Comments" page.* * *All comments and their details must be retrievable from the database.* | | | | | | | | | | | | |

## 

***3.41. Get comment by user id***

******

| **USE CASE-17 SPECIFICATION** | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Use-case No.** | UC017 | | | | **Use-case Version** | | | | UC003 | | | |
| **Use-case Name** | Get comments by User ID | | | | | | | | | | | |
| **Author** | Nguyen Tan Hung | | | | | | | | | | | |
| **Date** | 25/02/2024 | | | | **Priority** | | | | UC003 | | | |
| **Actor:**  *Customer, Creator.*  **Summary:**  *This use case describes the process of retrieving all comments made by a specific user identified by their User ID.*  **Goal:**  *As a moderator, I want to access all comments submitted by a specific user identified by their User ID.*  **Triggers**  *The moderator initiates a search for comments submitted by a specific user by entering their User ID.*  **Preconditions**  *The moderator must be logged into the system.*  **Post Conditions:**  *The moderator can view all comments submitted by the specified user.*  **Main Success Scenario:**   | ***Step*** | ***Actor Action*** | ***System Response*** | | --- | --- | --- | | *1* | *Moderator enters the User ID of the user whose comments they want to retrieve.* | *The system retrieves and displays all comments submitted by the specified user.* |   **Alternative Scenario:**  *N/A.*  **Exceptions**  *If there are no comments submitted by the specified user, the system displays a message indicating no comments found for that user.*  **Relationships**  *View comment detail*  **Business Rules:**   * *Only moderators can access the functionality to retrieve comments by User ID.* * *The system must accurately retrieve all comments submitted by the specified user from the database.* | | | | | | | | | | | | |

## 

# **NON-FUNCTIONAL Requirements**

*[This section describes the non-functional requirements of the system. Some examples are listed as below]*

## ***Usability***

### **4.1.1 User Interface Design**

*Direct intuitive interface, easy to use for users with many different roles. Art projects are clearly displayed whether the user is logged in or not*

### **4.1.2 Navigation and Accessibility**

*The platform shall provide easy and intuitive navigation for users to browse, search, and interact with artwork effectively. Navigation menus, search functionalities, and browsing options shall be prominently displayed and easily accessible from any page within the platform.*

### **4.1.3 Consistency and Familiarity**

*The platform shall maintain consistency in layout, terminology, and user interaction patterns to ensure a familiar and predictable user experience.*

## ***Reliability***

### **4.2.2 Navigation and Accessibility**

*The artwork sharing platform shall demonstrate high reliability and stability, with minimal downtime and system failures. It shall be capable of handling concurrent user interactions and large volumes of data without compromising performance.*

### **4.2.2 Data Integrity**

*It shall implement robust data validation and error-checking mechanisms to prevent data corruption or loss.*

## ***Performance***

### **4.3.1 Response Time**

*The platform shall exhibit fast response times for user interactions, with minimal latency in loading pages and displaying content.*