**Final Year Project**



SnapArt Pro

**Internal Advisor:**

Dr. Hussam

**External Advisor:**

Sumaib Hamza (CEO, CodeSquare)

**Project Manager:**

Dr. Saad Razzaq (CS & IT Dept, UOS)

**Project Team:**

Muhammad Saad Hussain (BSCS51F20R041)

Moeez Ur Rehman (BSCS51F20R016)

Muhammad Bilal (BSCS51F20R021)

Muhammad Sheraz (BSCS51F20R015)

**Table of Contents**

**Chapter 1: (Proposal)**

[**1. Abstract 3**](#_Toc49658243)

[**2. Background and Justification 3**](#_Toc49658244)

[**3. Project Methodology 3**](#_Toc49658245)

[**4. Project Scope 3**](#_Toc49658246)

[**5. High level Project Plan 3**](#_Toc49658247)

[**6. References 3**](#_Toc49658248)

[**7. Gantt chart 3**](#_Toc49658248)

**Chapter 2: Software Requirements Specification**

**2. Introduction**

**2.1 Purpose of Document**

**2.2 Project Overview**

**2.3 Scope**

**Overall System Description**

**2.1 User Characteristics**

**2.2 Operating Environment**

**2.3 System Constraints**

[4.1 User Class 1 – Image to Sketch Converter](#_Toc17856)

[4.1.1 Functional Requirement 1.1](#_Toc2542)

[4.1.2 Functional Requirement 1.2](#_Toc1392)

[4.1.3 Functional Requirement 1.3](#_Toc19960)

[4.1.4 Functional Requirement 1.4](#_Toc18852)

[4.1.5 Functional Requirement 1.5](#_Toc901)

[4.2 User Class 2 – AI Art Generation](#_Toc2336)

[4.2.1 Functional Requirement 2.1](#_Toc28867)

[4.2.2 Functional Requirement 2.2](#_Toc23024)

[4.2.3 Functional Requirement 2.3](#_Toc25050)

[4.2.4 Functional Requirement 2.4](#_Toc21690)

[4.3 User Class 3 – Camera Overlay](#_Toc7534)

[4.3.1 Functional Requirement 3.1](#_Toc23494)

[4.3.2 Functional Requirement 3.2](#_Toc8726)

[4.3.3 Functional Requirement 3.3](#_Toc11023)

[4.3.4 Functional Requirement 3.4](#_Toc7318)

[4.3.5 Functional Requirement 3.5](#_Toc542)

[4.3.6 Functional Requirement 3.6](#_Toc17630)

[5. Non-functional Requirements](#_Toc32129)

[5.1 Safety](#_Toc35) Requirements

[5.3 Performance](#_Toc22661) Requirements

[5.4 User](#_Toc7527) Documentation

# Chapter 1: Proposal

1. Abstract

The proposed project aims to develop an Android application in Java that serves as a versatile tool for artists, art learners, and enthusiasts. This innovative app will allow users to transform ordinary images into sketches, view the sketches overlaid on their camera feed, support device mirroring for large-scale art projects, and even generate AI-generated artworks. With a target audience comprising artists and individuals interested in honing their artistic skills, this app promises to be a valuable addition to the world of digital art creation and education.

2. Background and Justification

Artistic expression is a timeless human endeavor, and with the advent of technology, it has found new forms and platforms. Digital art creation has gained immense popularity, providing artists with tools to explore their creativity. However, there remains a gap in accessible, user-friendly applications that aid in the transition from traditional art to digital mediums.

The proposed Android application addresses this gap by offering a multifaceted solution for artists and art enthusiasts. It leverages image processing and AI technologies to convert photos into sketches, fostering an environment conducive to learning and artistic development. Furthermore, the inclusion of device mirroring capability facilitates the creation of large-scale art pieces. This application is driven by the need to empower artists, assist art learners in tracking their progress, and provide a creative outlet for all.

3. Project Methodology

The development of this Android app will follow a systematic approach, combining software engineering and creative design principles:

a. Requirements Gathering: We will engage with the target audience, including artists and art learners, to understand their needs and preferences. This will inform the feature set and design of the application.

b. Design and Prototyping: The project team will create wireframes and prototypes to visualize the user interface and functionality. User feedback will be incorporated for refinement.

c. Development: The application will be developed using Java, leveraging Android Studio for the Android platform. Image processing and AI algorithms will be implemented to achieve the desired features.

d. Testing: Rigorous testing will be conducted to ensure the app's functionality, performance, and user experience meet high standards.

e. Deployment: The app will be published on the Google Play Store, ensuring accessibility to a wide user base.

f. User Feedback and Iteration: Continuous feedback collection and updates will be carried out to improve the app based on user experiences and evolving technology.

4. Components

The Android app will consist of the following key components:

a. Image to Sketch Converter: This component will employ advanced image processing techniques to transform user-uploaded photos into artistic sketches.

b. Camera Overlay: The application will overlay the sketch on the camera screen in real-time, allowing users to view their artwork in progress.

c. Device Mirroring: Users will be able to mirror their device screens, enabling the creation of large-scale artworks while tracking progress.

d. AI Art Generation: The app will incorporate AI algorithms to generate unique artworks based on user inputs and preferences.

e. User Profile and Progress Tracking: Users can create profiles to save and track their artistic journey, enabling them to see their progress over time.

5. Project Scope

The scope of this project encompasses the development of an Android application that includes the following features:

- Image to sketch conversion with a variety of artistic filters.

- Real-time camera overlay of sketches on the camera feed.

- Device mirroring for large canvas projects.

- AI-based artwork generation.

- User profiles and progress tracking.

- Intuitive and user-friendly interface.

- Compatibility with a range of Android devices.

The project will also involve continuous updates and improvements to enhance the user experience and add new features based on user feedback and emerging technologies. The final product will cater to artists and art enthusiasts, offering them a valuable tool for artistic expression and growth.

6. High level Project Plan

The project plan has been illustrated in graphical which called as Gantt chart that involves the duration of tasks that shows what is the start and end time for activities in each stage of SDLC.

# 7. Gantt Chart

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr# | Task | Start Date | End Date | Duration |
| 1 | Creating a plan | 15-aug-23 | 30-aug-23 | 15 days |
| 2 | Documentation | 1-sept-23 | 15-nov-23 | 90 days |
| 3 | Developing a database for Users | 15-nov-23 | 5-dec-23 | 30 days |
| 4 | Creating Application structure | 5-dec-23 | 30-dec-23 | 15 days |
| 5 | Designing | 30-dec-23 | 20-jan-24 | 25days |
| 6 | Functionality | 20-jan-24 | 29-feb-24 | 45 days |
| 7 | Assigning Database | 29-feb-24 | 20-march-24 | 15 days |
| 8 | Testing of Application | 20-march-24 | 30-march-24 | 15 days |
| 9 | Implementation | 30-march-24 | 12-april-24 | 12days |
| 10 | Final Testing | 12-april-24 | 10-may-24 | 23 days |

# Chapter 2: Software Requirements Specification

**Introduction**

**2.1. Purpose of Document**

This document serves as the introduction to the "SnapArt Pro" Final Year Project (FYP) proposal. It outlines the purpose of the project, its scope, and provides a brief project overview.

**2.2 Project Overview**

The "SnapArt Pro" project is aimed at developing an Android application that caters to artists, art learners, and enthusiasts. This versatile application enables users to convert photos into sketches, view these sketches overlaid on their camera feed, support device mirroring for large-scale art projects, and even generate AI-generated artworks. It aims to bridge the gap between traditional and digital art, providing a user-friendly platform for creative expression and learning.

**2.3 Scope**

The scope of this FYP encompasses the development of the "SnapArt Pro" Android application with features such as image to sketch conversion, real-time camera overlay, device mirroring, AI-based artwork generation, user profiles for progress tracking, and a user-friendly interface. The project will also involve continuous updates and improvements based on user feedback and emerging technologies, ultimately catering to artists and art enthusiasts, offering them a valuable tool for artistic expression and growth.

**Overall System Description**

**2.1.1 User Characteristics**

The primary users of the "SnapArt Pro" application will be artists, art learners, and individuals with an interest in artistic expression. These users may vary in their artistic proficiency, ranging from beginners to advanced artists. The application is designed to be user-friendly, making it accessible to a broad audience with varying levels of technical expertise.

**2.2 Operating Environment**

The "SnapArt Pro" Android application will operate within the Android mobile ecosystem. It is compatible with a wide range of Android devices, ensuring accessibility to a diverse user base. Users will install the application from the Google Play Store. The app's functionality will rely on image processing and AI algorithms, and it will require camera access to provide real-time camera overlay features.

**2.3 System Constraints**

While the "SnapArt Pro" application aims for versatility and accessibility, there are certain constraints that the project will need to consider, including.

**Hardware Limitations:**

The app's performance may vary depending on the user's device capabilities, such as processing power and camera quality.

**Data Privacy:**

User data, including photos and preferences, will be stored securely, and privacy concerns must be addressed to protect users' sensitive information.

**Software Compatibility:**

The application's compatibility with different Android versions and devices needs to be ensured to reach a broad user base.

**Continuous Updates:**

As technology and user needs evolve, the project must commit to providing regular updates and improvements to maintain the application's relevance and performance.

This overall system description provides an initial understanding of the project's context, its intended users, and the operating environment while acknowledging the constraints that will shape the development and deployment of the "SnapArt Pro" Android application.

**3. External Interface Requirements**

**3.1 Hardware Interfaces**

The "SnapArt Pro" Android application will require access to the following hardware interfaces:

**Camera:**

The app will interact with the device's camera for capturing images and providing real-time camera overlay.

**Storage:** Access to the device's storage will be necessary to save user preferences, sketches, and AI-generated artworks.

**3.2 Software Interfaces**

The application will need to interact with various software components and services:

**Android OS:**

The app will run on the Android operating system and will need to be compatible with various Android versions.

**Google Play Store:**

The application will be published and distributed through the Google Play Store, requiring adherence to Google's policies and interface standards.

**4. Functional Requirements**

The functional requirements of the "SnapArt Pro" application include, but are not limited to, the following:

**Image to Sketch Converter:**

The application must be able to convert user-uploaded photos into sketches using advanced image processing techniques.

**Camera Overlay:**

The app should overlay sketches on the camera feed in real-time, allowing users to view and interact with their art in progress.

**AI Art Generation:**

The application should incorporate AI algorithms to generate unique artworks based on user inputs and preferences.

**User Profile and Progress Tracking:**

Users should be able to create profiles, save their artistic work, and track their progress over time.

**Intuitive User Interface:**

The app's user interface should be user-friendly and intuitive for users with varying levels of technical expertise.

**Compatibility:**

The application must work seamlessly on a wide range of Android devices and versions.

# Chapter 3: Functional Specification

## **1. Introduction**

## **1.1 Purpose of Document**

The purpose of this document is to give a detailed description of the requirements for the software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface, user characteristics, functional requirement and non-functional requirements for the project.

## **1.2 Project Overview**

Our App aims to provide comprehensive image processing capabilities, including converting images to sketches and advanced photo editing features. Leveraging AI, we also enable the generation of unique imagery. Additionally, the project includes a robust system for saving and managing artwork in local storage, ensuring users can easily access and organize their creations. This combination of tools enhances creative workflows and supports artistic expression.

## **1.3 System Architecture**

 **User Interface**: A frontend for seamless interaction and user-friendly design.

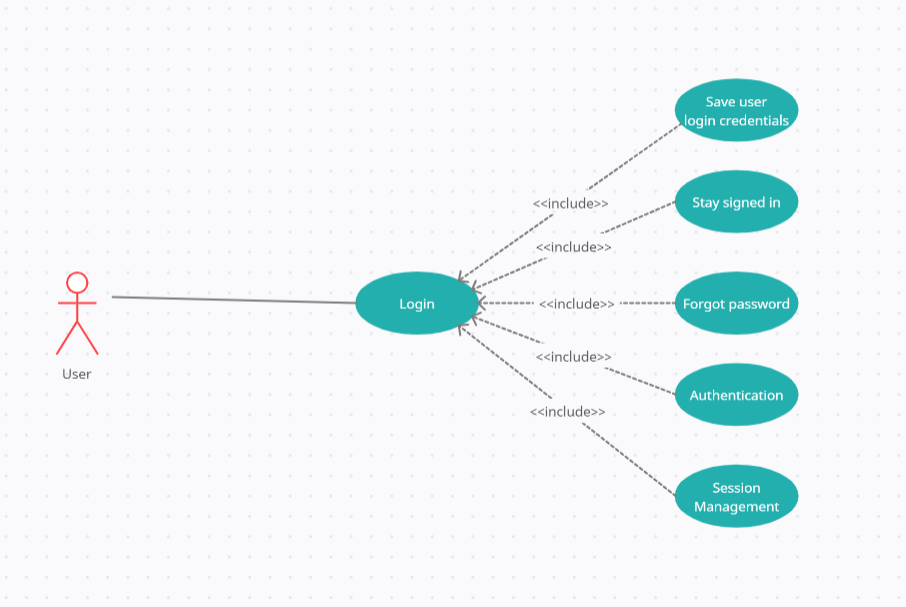
 **Backend Server**: Handles image processing, photo editing tasks, and AI model integration.

 **AI Modules**: Specialized components for image-to-sketch conversion and AI image generation.

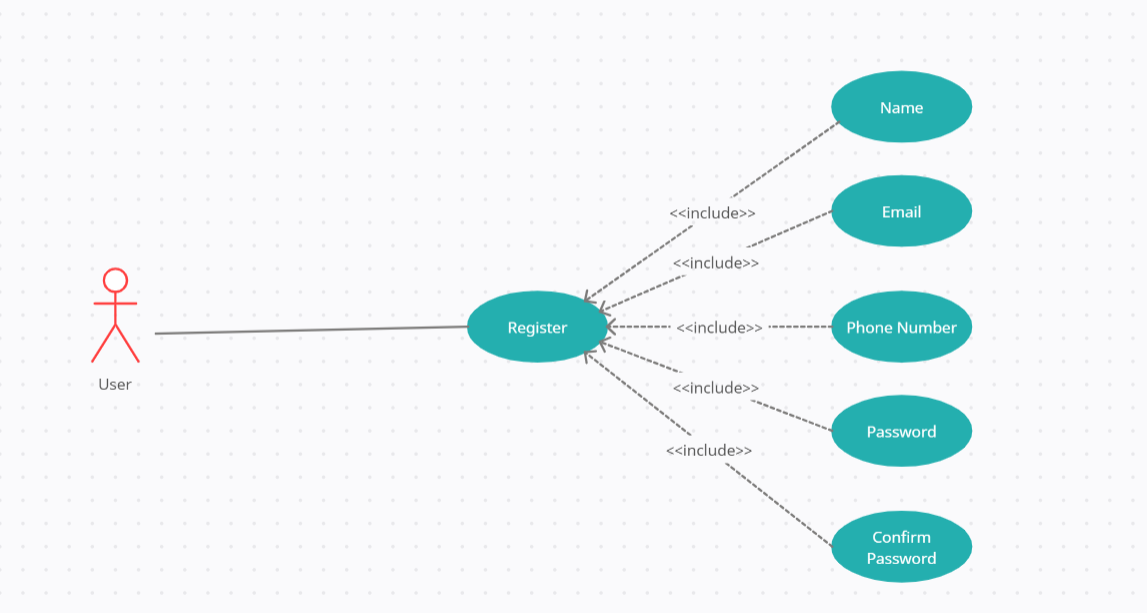
 **Local Storage Integration**: Manages the saving and retrieval of artworks on the user's device.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case 1: Account>** | | | | |
| **Actors:**  User. | | | | |
| **Feature:** Make Profile for Authentication. | | | | |
| **Use case Id:** | | 1. | | |
| **Pre-condition:** | | Before engaging with the SnapArtPro Application we insure a secure user authentication system is there secure login and authentication mechanism to prevent unauthorized access. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1** | User will signup to create Account . | | | An informational page will be required to move forward. |
| **2** | User will make Account. | | | Success Account will be validated. |
| **3** | User will go to Login page. | | | Informational page gets information and authenticate from database. |
| **4** | User click Login Button. | | | There will be Account Login if information is authenticated. |
| **Alternate Scenarios:***.* | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Success Account will be validated. | | | |
| **2** | There will be Account Login if information is authenticated. | | | |
| **3** | User will be able to use to Application if Account persist. | | | |
| **Use Case Cross referenced** | | | *<Related use cases, which use or are used by this use case>* | |
| **User Interface reference** | | | *List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.* | |
| **Concurrency and Response** *Give an estimate of the following*   * *Number of concurrent users* * *Expected response time of the use case* | | | | |

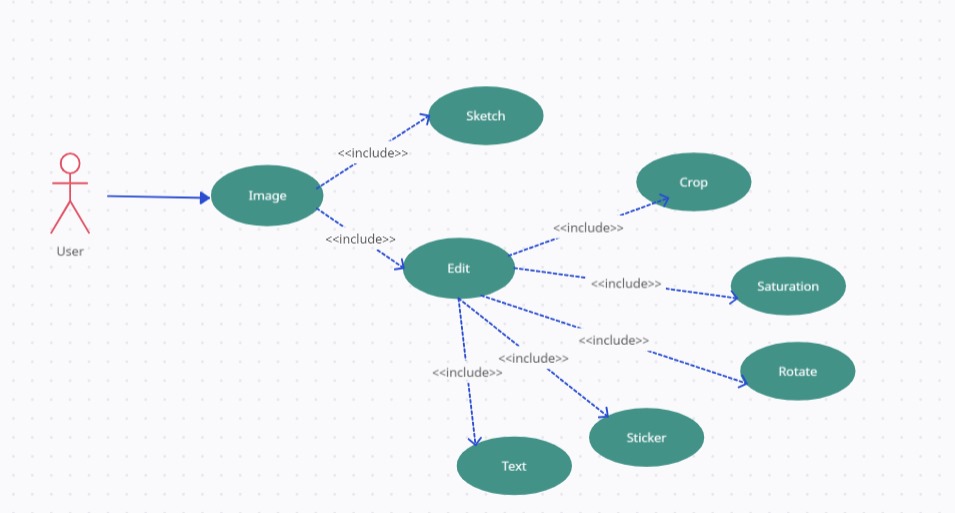
|  |  |  |  |
| --- | --- | --- | --- |
| **<Use case 1.1: User Login>** | | | |
| **Actors:** User | | | |
| **Use case Id:** | | 1 | |
| **Pre-condition:** | | User must be connected with the internet and user registration is required for login | |
| **Scenarios** | | | |
| **Step#** | **Action** | | **Software Reaction** |
| **1.** | User will enter his Email | | The System will check Email in the system |
| **2.** | User will enter his password | | The system will check user password that was stored in the authentication system |
| **3.** | User will click on login button | | The system will redirect the user to the home page of the website |
| **1a:** If user will enter wrong username or password the system will give the error and will ask to user re-enter his email or password.  **2a:** If the user will give wrong password 10 times the user account will be blocked. | | | |
| **Post Conditions** | | | |
| **Step** | **Description** | | |
| **1** | If the user will give right password and username then he will redirect to the home page of the website and perform any functionality. | | |



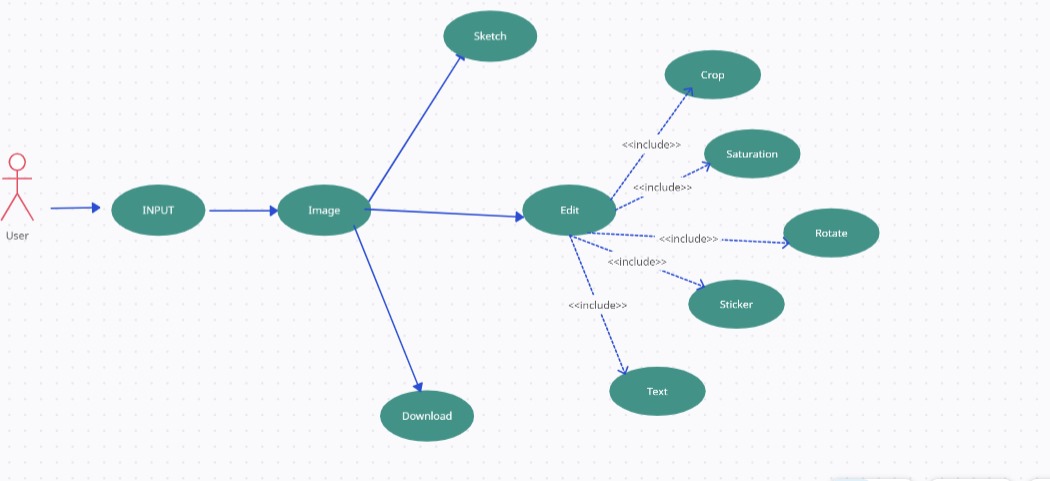
|  |  |  |  |
| --- | --- | --- | --- |
| **<Use case 1.2: User Registration >** | | | |
| **Actors:** User | | | |
| **Use case Id:** | | 2 | |
| **Pre-condition:** | | User must be connected with the internet and will give the website URL on any web browser | |
| **Scenarios** | | | |
| **Step#** | **Action** | | **Software Reaction** |
| **1.** | Normal user will enter his complete detail that will be required for registration. | | The System will store all the information in the Authentication system |
| **2.** | The user will enter his email. | | The system will verify his Email and sends the opt for verification |
| **3.** | The super user will be created using command line system command will be used for super user. | | The system will check the command and allow the user to become a super user for the Application. |
| **4.** | The super user will verify by email | | The system sends the Email to Super user and verifies them |
| **Alternate Scenarios:** *Write additional, optional, branching or iterative steps. Refer to specific action number to ensure understandability.* | | | |
| **1a:** If the user password and re-password didn’t match, the system gives the error and asked the user to re-enter your password.  **2a**. If user will give wrong Email id the system gives the error for correct Email.  **3a.** If email-id is not verified by the user, the registration system will not create a user account. | | | |
| **Post Conditions** | | | |
| **Step** | **Description** | | |
| **1** | After the successful registration the user will login in system as authorized user.  The system will store all the information about the user in the authentication system for future processes. | | |



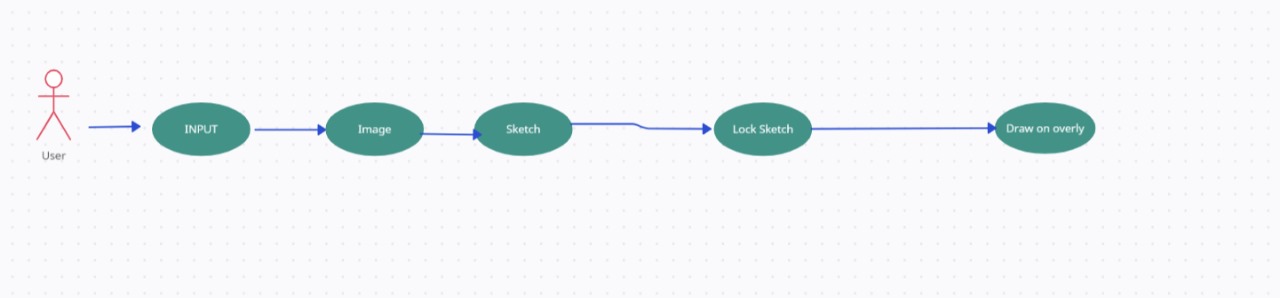
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case 2: Sketch>** | | | | |
| **Actors:**  Artist. | | | | |
| **Feature:** Take image and convert it to a sketch. | | | | |
| **Use case Id:** | | 2. | | |
| **Pre-condition:** | | Before engaging with the SnapArtPro Application we insure a secure user authentication system to prevent unauthorized access and stored image to be converted. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1** | Artist will choose a photo . | | | Photo will get options to be edited or sketched. |
| **2** | Artist will choose Action. | | | Photo will get behavior to be edited or sketched. |
| **3** | Artist can make sketch of photo. | | | Photo will be converted into sketch. |
| **4** | Artist can edit photo. | | | There will be a series of options to change behavior and display of image. |
| **Alternate Scenarios:***.* | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Photo will be converted into sketch. | | | |
| **2** | Visual appeal of photo changes and converted to different graphics formats. | | | |
| **3** | A real change will be in photo will be adopted either in sketch or edited format. | | | |
| **Use Case Cross referenced** | | | *<Related use cases, which use or are used by this use case>* | |
| **User Interface reference** | | | *List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.* | |
| **Concurrency and Response** *Give an estimate of the following*   * *Number of concurrent users* * *Expected response time of the use case* | | | | |



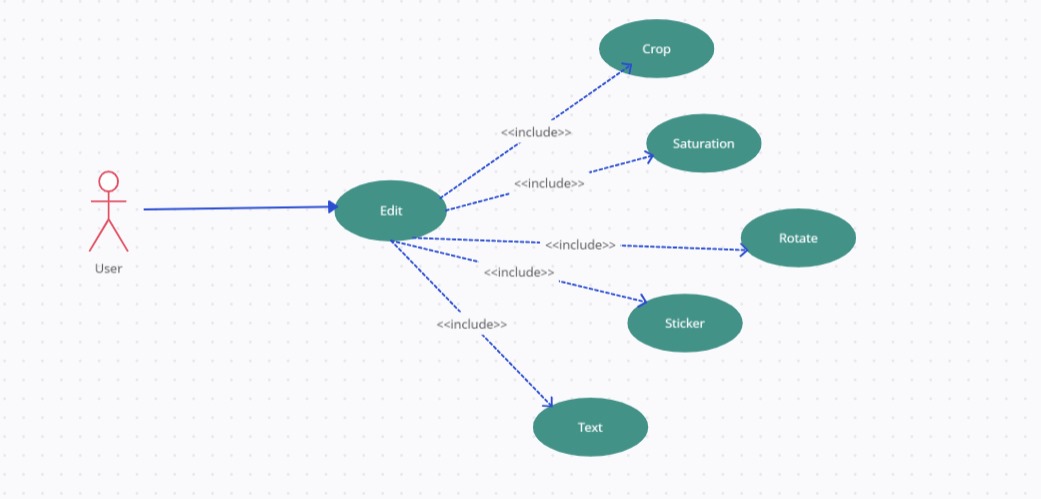
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case 3: AI ArtGeneration >** | | | | |
| **Actors:**  User. | | | | |
| **Feature:** Random idea to create image of input. | | | | |
| **Use case Id:** | | 3. | | |
| **Pre-condition:** | | Before engaging with the SnapArtPro Application we insure a secure user authentication system to prevent unauthorized access and Knowledge to give input and idea to create scenario for image. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1** | User will give an input. | | | Photo will be generated from input through Artificial intelligence. |
| **2** | User will choose Action. | | | Photo will get behavior to be edited or sketched. |
| **3** | User can make sketch of photo. | | | Photo will be converted into sketch. |
| **4** | User can edit photo. | | | There will be a series of options to change behavior and visual appearance of image. |
| **Alternate Scenarios.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Photo will be generated from input and converted into sketch. | | | |
| **2** | Visual appeal of photo changes and converted to different graphics formats. | | | |
| **3** | A real change will be in photo will be adopted either in sketch or edited format. | | | |
| **Use Case Cross referenced** | | | *<Related use cases, which use or are used by this use case>* | |
| **User Interface reference** | | | *List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.* | |
| **Concurrency and Response** *Give an estimate of the following*   * *Number of concurrent users* * *Expected response time of the use case* | | | | |



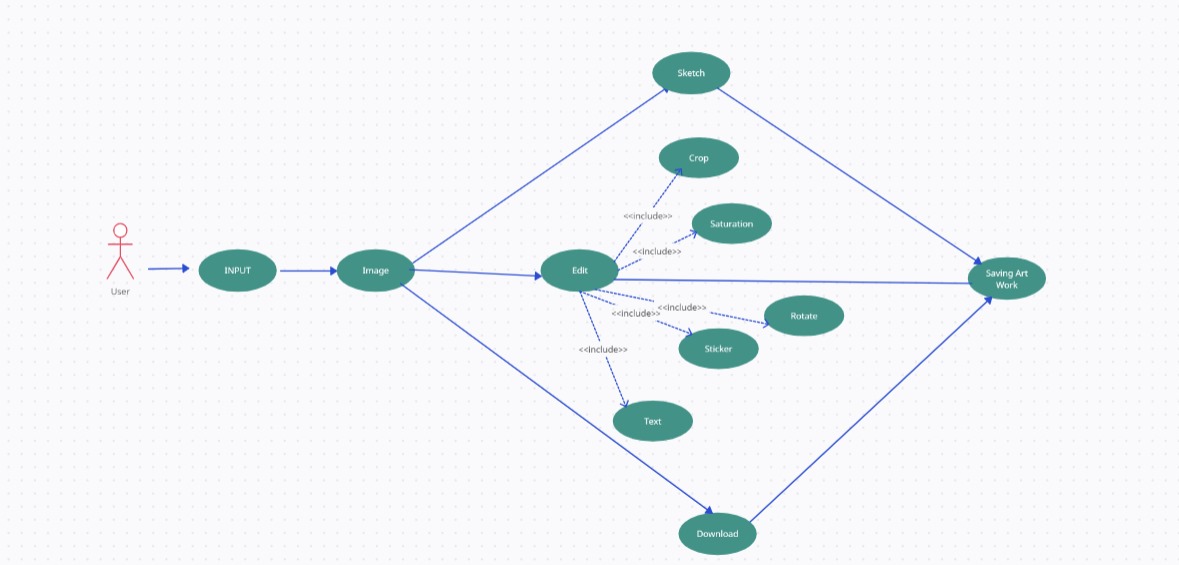
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case 4: Camera Overlay>** | | | | |
| **Actors:**  User. | | | | |
| **Feature:** Camera will make an overly for sketch to be drawn by hands. | | | | |
| **Use case Id:** | | 4. | | |
| **Pre-condition:** | | Before engaging with the SnapArtPro Application we insure a secure user authentication system to prevent unauthorized access and Camera on android device to capture photo. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1** | User will make a sketch on app. | | | Sketch will be made in Application. |
| **2** | User will choose Action. | | | Sketched photo will be locked on camera. |
| **3** | User will try to hold sketch locked. | | | Locked sketch will not move. |
| **4** | User can make sketch on paper or tripod. | | | The sketch on camera will overlay background that is ready for hand made sketch where user will make sketch by hand using camera overlay Functionality. |
| **Alternate Scenarios.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Sketch will be made in Application. | | | |
| **2** | Locked sketch will not move. | | | |
| **3** | The sketch on camera will overlay background that is ready for handmade sketch where user will make sketch by hand using camera overlay Functionality. | | | |
| **Use Case Cross referenced** | | | *<Related use cases, which use or are used by this use case>* | |
| **User Interface reference** | | | *List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.* | |
| **Concurrency and Response** *Give an estimate of the following*   * *Number of concurrent users* * *Expected response time of the use case* | | | | |



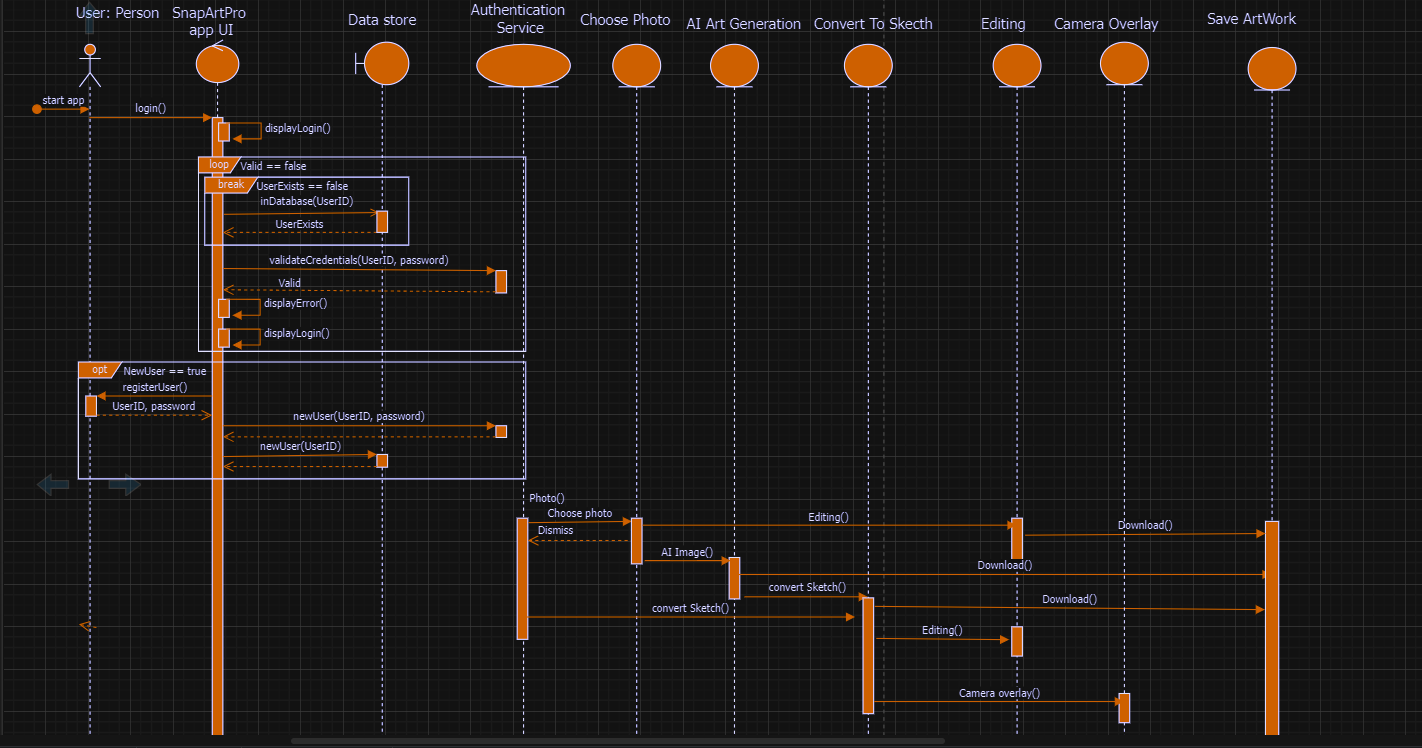
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case 5: Photo Editing>** | | | | |
| **Actors:**  Artist. | | | | |
| **Feature:** Photo editing kits will edit photo. | | | | |
| **Use case Id:** | | 5. | | |
| **Pre-condition:** | | Before engaging with the SnapArtPro Application we insure a secure user authentication system to prevent unauthorized access and sketch on a mirroring device. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1** | Artist will make a sketch on device. | | | Sketch can be edited by the using different values. |
| **2** | Photo will be Edited. | | | Captured photo will be edited by the device. |
| **3** | All fields will give new look to photo. | | | Photo will get new colors and different hue intensities. |
| **4** | Photo add multiple stickers | | | Photo will add multiple stickers on it to change the shape or add in shape. |
| **Alternate Scenarios.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Captured photo will be edited by the device. | | | |
| **2** | Photo will get new colors and different hue intensities. | | | |
| **3** | Photo will add multiple stickers on it to change the shape or add in shape. | | | |
| **Use Case Cross referenced** | | | *<Related use cases, which use or are used by this use case>* | |
| **User Interface reference** | | | *List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.* | |
| **Concurrency and Response** *Give an estimate of the following*   * *Number of concurrent users* * *Expected response time of the use case* | | | | |

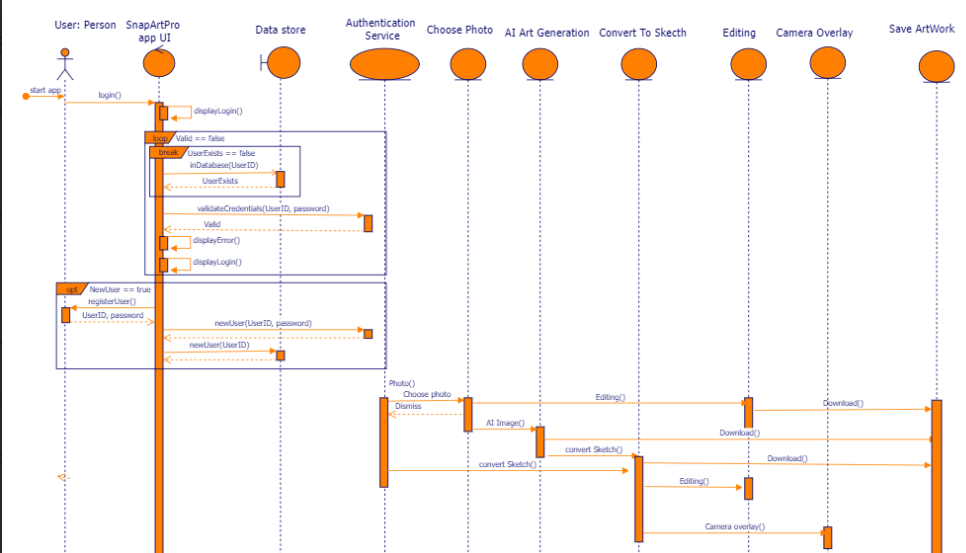


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case 6: ArtWork Storage>** | | | | |
| **Actors:**  User. | | | | |
| **Feature:** capability to Store the sketch on internal storage of Android Device. | | | | |
| 6. | | |
| Before engaging with the SnapArtPro Application we insure a secure user authentication system to prevent unauthorized access and storage capacity in device for saving Artwork. | | |
| **Scenarios** | | | | |
| **Action** | | **Software Reaction** | |
| Artist will make a sketch on device. | | Sketch will use generated on device. | |
| Artist capture image by camera. | | Image will be captured on device. | |
| Artist will generate AI images. | | Images will be generated. | |
| Artist will click save button. | | All Artwork and images will be saved in device storage. | |
| **Alternate Scenarios.** | | | | |
| **Post Conditions** | | | | |
| **Description** | | | |
| Sketch will use generated on device. | | | |
| All Artwork and images will be saved in device storage. | | | |
| This Artwork can be used for future usage or learning process. | | | |
| **Use Case Cross referenced** | *<Related use cases, which use or are used by this use case>* | | | |
| **User Interface reference** | *List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.* | | | |
| **Concurrency and Response** *Give an estimate of the following*   * *Number of concurrent users* * *Expected response time of the use case* | | | | |

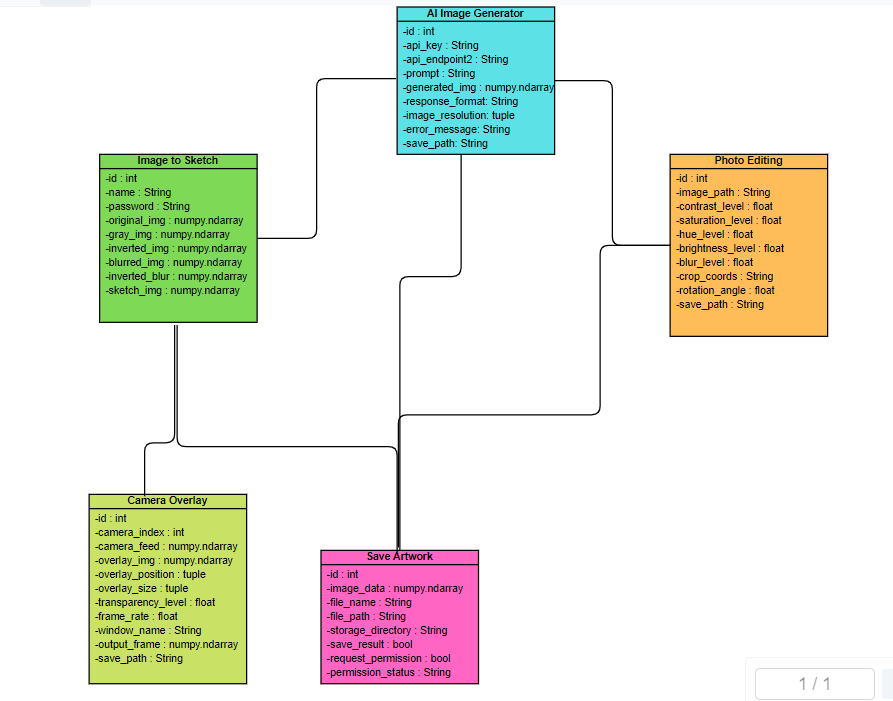


# Sequence Diagram:

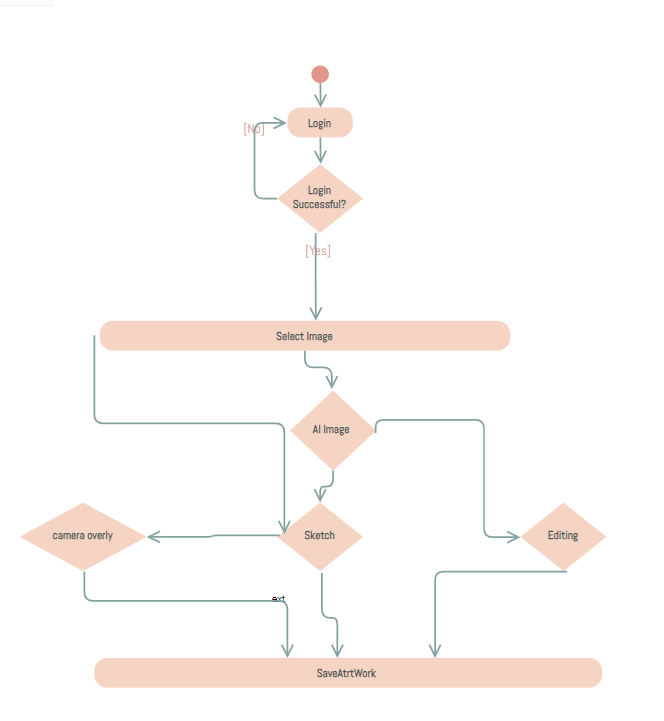




**Class Diagram:**



**Activity Diagram:**



**5. Non-Functional Requirements**

**5.1 Performance Requirements**

The performance requirements for the "SnapArt Pro" application include:

**Responsiveness:**

The application should respond promptly to user inputs and deliver real-time camera overlay without significant delays.

**Scalability:**

It should be able to handle a growing user base and the storage of user-generated content.

**Resource Efficiency:**

The application should utilize device resources efficiently to prevent excessive battery drain or slowdowns.

**5.2 Safety Requirements**

The "SnapArt Pro" application should ensure user safety by:

**Data Security:**

Protecting user data, including images, preferences, and personal information, through robust data encryption and secure storage practices.

**Privacy:**

Adhering to privacy regulations and providing clear and transparent privacy policies to users.

**5.3 Security Requirements**

To maintain the security of the application and user data:

**User Authentication:**

Implementing secure login and authentication mechanisms to prevent unauthorized access.

**Protection from Malware:**

Regular security audits and measures to protect the app from malware and hacking attempts.

**5.4 User Documentation**

The application should provide comprehensive user documentation, including:

**User Guide:**

A detailed guide explaining how to use the application's features.

**Frequently Asked Questions (FAQs):**

Addressing common user queries.

**Contact Support:**

A method for users to seek help or report issues within the app.

These requirements ensure the functionality, performance, safety, and security of the "SnapArt Pro" application while offering users the necessary documentation and support for a seamless experience.

## GitHub link for project repositories

[moeez9593/SnapArt-Pro (github.com)](https://github.com/moeez9593/SnapArt-Pro)