Software Requirements Specification for Hairesthetics: A 3D Hairstyle Simulation iOS App

Team 18
Charlotte Cheng
Marlon Liu
Senni Tan
Qiushi Xu
Hongwei Niu
Bill Song

October 5, 2022

Contents

List of Tables

List of Figures

Revision History

Date	Version	Notes
Oct 1	1.0	Initial Drafts
Oct 4	1.1	Update template based on rubric

1 Project Drivers

1.1 The Purpose of the Project

The purpose for this document is to provide the basic requirements for a software application that helps user simulate hairstyles and hair colors virtually. It should include the essential functional requirements, which specify the behavioral capability of the system, as well as the non-functional requirements in order to create a successful and robust application.

This document should also act like an agreement between the stakeholders and the developers that the design should be based on the critical information provided in this document. The requirements should be the guiding principles in the design phase to guarantee that all the requirements are met.

1.2 Stakeholders

1.2.1 User

People who have the demand to change their hair style and see how other hairstyles and different hair colors match their appearance virtually are welcome to explore in this iOS application.

1.2.2 Client

The clients for this project with be the students from the capstone course. These individuals will review and test the software before being deployed to the general public. In addition, the teach assistants and the supervisor will be actively involved throughout the construction of this project.

1.2.3 Other Stakeholders

Other stakeholder could be the hair dressers and barbers. They can provide this app for their customer to help making a hairstyle decision.

1.2.4 User Characteristics

The identified users of the project can be categorized into 2 distinct groups. These groups are differentiated based on their varying background, experience of the user and their intended use(s) of the system. They are classified as the novice user and administrative user.

It is assumed that all system users have experience operating with iOS applications. More

specifically, all users are assumed to have experience working with iPhone, iOS 14, and above. They have been exposed to the camera app and most common icons by Apple.

- Novice User: The application requires no prior training, and it is meant for the public, so all non-administrative users are novice users. Their intention is to use the application to simulate hairstyles and hair colors virtually. Anyone who does not have optic disabilities and other physical disabilities that make them unable to use the application on a smartphone can use the application.

- Administrative User: Their intention is not to use the application to simulate hairstyles, but to calibrate, configure, and maintain the application to make sure it functions properly and to its maximal capabilities. They wish to fix bugs, maintain database, and implement new features. They have a basic knowledge of computers, data management and app development.

2 Project Constraints

2.1 Mandated Constraints

2.1.1 Solution Constraints

Description: The product shall operate using iOS 14 and above.
Rationale: The users use iOS 14 and do not want to update system.
Fit Criterion: The product shall be approved as iOS 14 compliant by developers.

Description: The product shall use the default camera system on the iphone device to obtain the input image and videos from users.
Rationale: The users will not pay for a new graphic input device.
Fit Criterion: All graphical inputs should come from the default camera with the device.

2.1.2 Implementation Environment of the Current System

- Source code is entirely written in Python and swift.
- Backend framework Flask is used to create user login system.
- MongoDB is used for storing and manipulating the password database.
- iOS AR kit is used for developing AR components.
- OpenCV is mainly used for capturing and analysing user's facial feature

2.1.3 Off-The-Shelf Software

- Facial Contour Detection API from google ML Kit.

2.1.4 Partner or Collaborative Applications

N/A.
All the external libraries / API used are open-sourced and free of charge.

2.1.5 Anticipated Workplace Environment

The workplace could be anywhere with a visible screen and camera. When The workplace is outside, the device must be weather resistant, have displays that are visible in sunlight, and allow for the effect of surrounding environment on output.

2.1.6 Schedule Constraints

Project must be completed before the end of the academic year. This will limit the amount, complexity and quality of the additional features upon initial release date.

2.1.7 Budget Constraints

The cost of the project must not exceed $750. This limitation will constrain the number of requirements that can be included in the project.

2.2 Naming Conventions and Terminology

2.2.1 Abbreviations and Acronyms

1. **UI** - User Interface
2. **GUI** - Graphical User Interface
3. **UI** - A high-level, modern programming language that is used both on the client-side and server-side.
4. **AI** - Artificial Intelligence
5. **API** - Application programming interface
6. **REST** - Representational state transfer
7. **OS** - Operating System
8. **RGB** - Red, Green, Blue

2.3 Relevant Facts and Assumptions

- Users will have access to the internet through WiFi or though their data plan from their mobile phone.
- Users can understand English text.
- Users have healthy vision and have no difficulties in seeing colors, graphs and words on the mobile screen.

3 Functional Requirements

3.1 The Scope of the Work

Many people wonder how a different hairstyle would look on them in real life. Most of them rely on their imagination and feelings, and some uses some applications with simple filters. Thus, the existing market lacks mobile applications that accurately simulate their look with a new hairstyle or color.

3.2 The Context of the Work

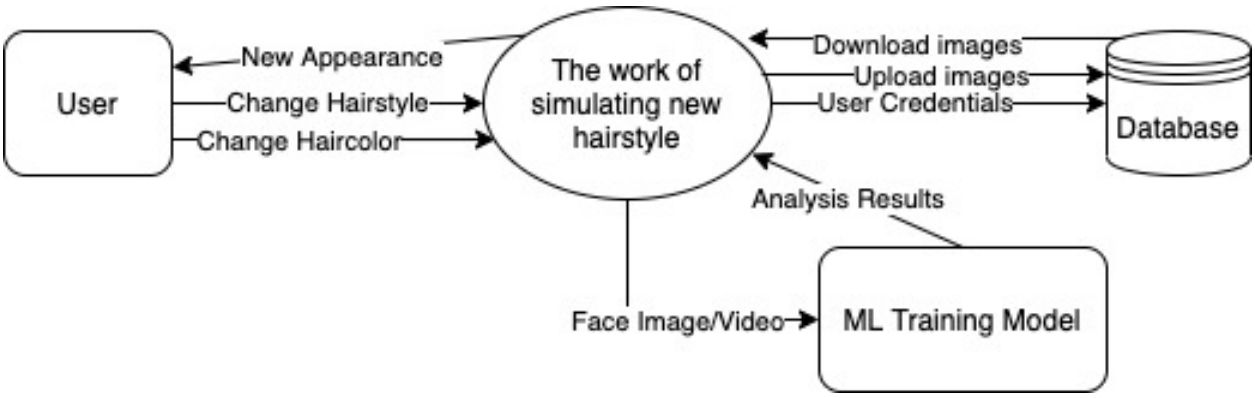

Figure 1: Context of Work Diagram

3.3 The Scope of the Product

3.3.1 In Scope

The goal of our application can be found below:

- Our software should be able to identify the facial features based on the input faces
- Our software should be able to provide the 3D simulation of hairstyles on user's head lively, the virtual hair should be able to move around with the user's head.
- Our software should allow the users to change the hair color of their own hair as well as the virtual hair they are trying on.
- Our software should be able to recommend hairstyles based on the users facial features.

Based on the above goals of our application, the scope of our requirements can be summarized to the following:

- Identify user facial features
- Simulate 3D virtual hairstyles
- Change hair colors
- Make hairstyle suggestions

3.3.2 Out of Scope

The stretch goals are currently out of the scope of our requirements document, which can be summarized to the followings:

- Support virtual modification of the simulating hairstyles: This allows the users to touch and change the simulated hairstyles and the system should modify the virtual hair accordingly.
- Hair style detection: This enables the software to detect the current hairstyle the users have with machine learning algorithms.
- Providing nearby hair salons: This goal suggest nearby hair salons to the users ranked by their ratings and reviews.

3.4 Work Partitioning

Table 1: **Work Partitioning**

Event Name	**Input/Output**	**Summary**
BE1: User pick new hairstyle	image/video(in), transformed image/video(out)	The system will transform desired hairstyle on user's image/video
BE2: User pick new hair color	image/video(in), transformed image/video(out)	The system will transform desired hair color on user's image/video
BE3: User save image	image/video(in)	The system will save the user's image/video
BE4: User download image	image/video(out)	The system will download the user's save image/video

3.5 Individual Product Use Cases

UC1: Change hairstyle: The user wants to change a hairstyle

UC2: Change hair color: The user wants to change a hair color

UC3: Upload images: The user wants to upload images (after transformation) to their account

UC4: Save images: The user wants to save their images (after transformation)

UC5: Log into account: The user wants to log into their account

3.6 The Context of the System

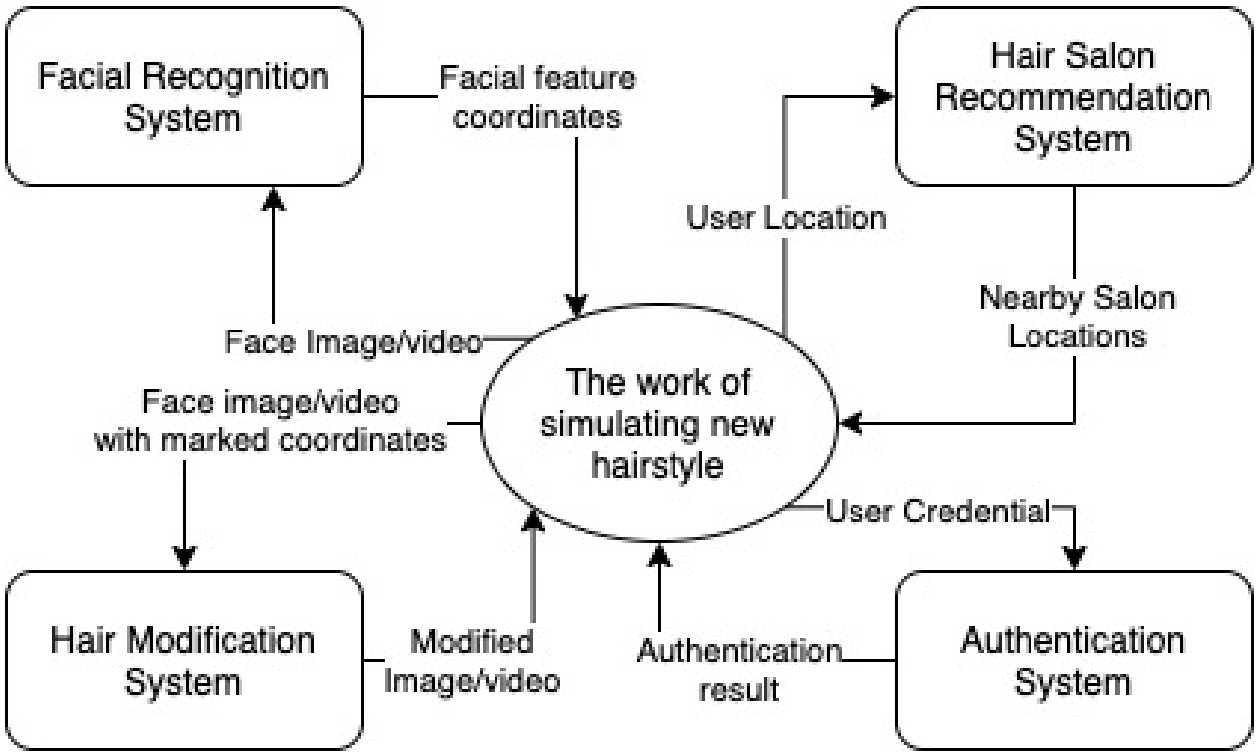

Figure 2: Context of System Diagram

3.7 Functional Requirements

3.7.1 Facial Recognition System

FR1: The user must input their face in image or video format

FR2: The system must pre-process the image by transforming the input facial image to grayscale

Rationale: Increase the contrastness of the image to better detect the edges

Math Specification: newImage := process($image$) $\Rightarrow$ ($\forall p$: Pixels in image | p[Red]*0.21 + p[$Green$]*0.72 + p[$Blue$]*0.07)

FR3: The system must load the grayscale facial image into a pre-trained facial detection model for face shape detection

Rationale: The pre-trained facial detection model needs a grayscale image as an input to provide a more accurate result

FR4: The system must load the pre-processed image into a pre-trained hair detection model for hair extraction

Rationale: The pre-trained hair detection model needs a grayscale image as an input to provide a more accurate result

FR5: The system compute the hair edge coordinates

Rationale: the pre-trained models will compute a series of important coordinates related to the hair. These coordinates will be used for various transformation.

FR6: The system must mark the corners and shape with facial landmark coordinates.

Rationale: Marking the boundary is important, because the transformations will only apply to the hair not the face.

FR7: The system must store the coordinates from the hair and facial features.

3.7.2 Hair Modification System

HM1: The user must choose a color from the provided options if hair color transformation is selected

HM2: The system must load the hair feature data for hair color transition

HM3: The system should map the hair coordinates on the image and apply the new rgb values to the region

Rationale: The hair coordinates are retrieved from facial recognition system, so new rgb colors can be applied to those regions

HM4: The system must load the facial coordinates for hair style transformation

HM5: The user must choose a hair style from the provided options if hair style transformation is selected

HM6: The system should load the selected hair style dataset from the database

HM7: the system should adjust the coordinate to be proportional with the image scale

HM8: The system should add the new hair style coordinates above the facial coordinates.

Rationale: depends on the hair style, some can be placed on top of the existing hair style, or re-construct

HM9: The system should output the transformed face in image or video.

3.7.3 Hair Salon Recommendation System

HR1: The user must enter a address or turn on current location to use the recommendation system

HR2: The system must retrieve 5 nearby hair salon locations based on the input address

HR3: The system should label the retrieved locations on a map

Rationale: marking the locations on the map allows the user to see the distance to their current location visually

HR4: The system should rank the locations based on user review

Rationale: the user reviews is a good indicator for deciding which one to go to

3.7.4 Authentication System

AR1: The user must enter valid username and password to access their profile in the system

AR2: The system must allow the user to upload their face image before and after hair transformation

Rationale: users might want to save the images for future references

AR3: The system shall upload images after successful log in

AR4: The system must allow the user to log out

3.8 Finite State Machine

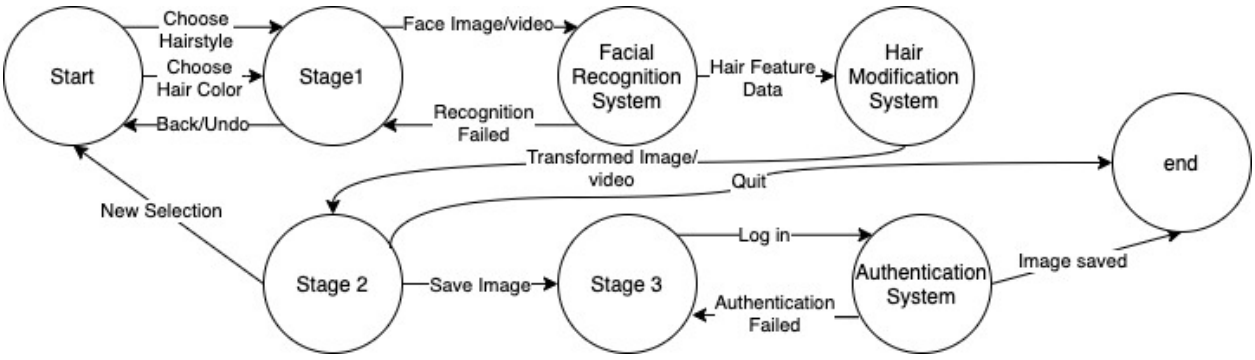

Figure 3: Finite State Machine

4 Nonfunctional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

APR1: The fonts must be outstanding and easily recognizable through the interface.

Fit Criteria: Volunteers from various age ranges will be invited to verify that they can see the text with the application font size.

APR2: The user interface should be well-organized while simple to use.

Fit Criteria: Volunteers from different age group will be invited to use the application and see if they can learn how to use by themselves.

4.1.2 Style Requirements

IAR1: The style of the interface must has a modern aesthetic.

Fit Criteria: The interface will be compared with the popular modern interface style found online. Professional UI Designers will be invited to take a look at the application interface to ensure the interface has a mordern aesthetic.

IAR2: The style of the interface must be simple and clear.

Fit Criteria: Volunteers will be invited to use the application, and there should be no user reflects that the interface is complicated, unclear, and hard to use.

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

EUR1: Interactable components on the interface should be emphasized to signify the user that the components can be interacted with.

Fit Criteria: Volunteers will be invited to use the application and see if they can intuitively find out all the interactable components.

EUR2: Interactable components on the interface should be big enough to prevent the user from misclicking.

Fit Criteria: Volunteers will be invited to use the application and see if they will misclick throughout using the application.

4.2.2 Personalization and Internationalization Requirements

N/A

4.2.3 Learning Requirements

LR1: The installation process must be completed within 3 steps.

LR2: The instructions shall be concise and easy to follow.

Fit Criteria: Volunteers will be invited to use the application and see if they can follow the instructions without problems.

4.2.4 Understandability and Politeness Requirements

UPR1: The icons should be taken from common usage icons when appropriate.

Fit Criteria: Volunteers will be invited to use the application and see if all the icons make sense where they were used

4.2.5 Accessibility Requirements

ASR1: Fonts shall be in appropriate color and size

Rationale: The content can be easily read by the user.

Fit Criteria: Volunteers will be invited to use the application and rate the readability of the content from 1-10, below 7 is a fail.

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

SLR1: The response time of hairstyle suggestion must be within 1 second.

SLR2: The response time of users changing hairstyles must be within 1 second.

SLR3: The response time of users clicking any other buttons must be within 0.5 seconds.

4.3.2 Safety-Critical Requirements

N/A

4.3.3 Precision and Accuracy Requirements

PAR1: The position of the AR hair must fit the users' head position on the screen within 1 millimeter accuracy.

PAR2: When users move their heads, the AR hair must move along the users' heads within 2 millimeters accuracy.

4.3.4 Reliability and Availability Requirements

RAR1: For all users, the application will be available to use without crashes over 99 percents of the time. (less than or equal to 1 crash per 100 access)

RAR2: The application must allow the users to use continuously until the hardware device is broken or out of battery.

Fit Criteria: There should be no errors during unit tests, integration test and user tests for the application.

4.3.5 Robustness Requirements

RBR1: The application must have exception handling methods to provide useful feedback if improper information is input.

Fit Criteria: There should be a pressure test on the application and developers will try to inject irrational inputs to break the application, but no errors or exceptions should occur.

RBR2: The application must have error case handling methods so that it can recover from simple errors such as stuck or no response.

Fit Criteria: There should be a pressure test on the application and developers will try to inject irrational inputs to break the application, but the application should not crash.

4.3.6 Capacity Requirements

CCR1: The application must have a database to store every user's account information.

Fit Criteria: Database developer will do tests on the database such that the count of users' account information in the database equals the actual registered number of user accounts.

CCR2: The product must have a database to store at least 10 different hairstyles.

4.3.7 Scalability and Extensibility Requirements

SER1: The application must allow new features to be added in the future.

Fit Criteria: Any possible future developing features will be kept in the section of Watch Room in this document.

4.3.8 Longevity Requirements

LGR1: The application must be supported while the libraries and frameworks being used are supported and all developers constantly maintaining.

Fit Criteria: Developers will regularly run tests to ensure the application is functioning.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

N/A

4.4.2 Requirements for Interfacing with Adjacent System

N/A

4.4.3 Productization Requirements

PDR1: The product must be available to download via Apple App Store.

Fit Criteria: Volunteers and developers will search and download the application via Apple App Store and all people should download the application successfully.

4.4.4 Release Requirements

RER1: Data from previous builds of the product should be compatible with new builds of the product.

Fit Criteria: Developers will regularly run tests to ensure that data can be used between different builds.

RER2: Version updates and descriptions of changes shall be included in change logs available to the user.

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirements

MSR1: The product will be maintained by the developers.

Fit Criteria: Developers will regularly run tests and fix bugs if any occur.

MSR2: The product shall undergo revision to check for any potential bugs.

Fit Criteria: Developers will regularly run tests to check for potential bugs.

MSR3: The mean time to restore the App (MTTRS) following a system failure must not be greater than 60 minutes. MTTRS includes all corrective maintenance time and delay time

4.5.2 Supportability Requirements

SPR1: The product shall be available on iPhone with iOS system 14 and after.

4.5.3 Adaptablity Requirements

ADR1: The product shall be compatible with iOS system 14 and after.

4.6 Security Requirements

4.6.1 Access Requirements

ACR1: Users will be able to access images they previously stored.

Fit Criteria: Developers will run tests as user role to ensure that stored data is accessible.

ACR2: Users will be able to access site data for job sites they are authorized for.

Fit Criteria: Developers will run tests in user role to ensure that data is accessible with permission.

ACR3: Admins and supervisors will be able to unlock locked resources.

Fit Criteria: Developers will run tests in admin role to ensure that locked resources are accessible to admins and supervisors.

ACR4: Only admins will be able to modify application information.

Fit Criteria: Developers will run tests in admin role to ensure that only admins can modify application information.

4.6.2 Integrity Requirements

IR1: The product will not modify data unnecessary.

Fit Criteria: Developers will run tests in a sandbox to verify that data is only modified when necessary.

IR2: The product will not modify any data unrelated to its execution.

Fit Criteria: Developers will run tests in a sandbox to verify that data is only modified when it is related to its execution.

IR3: Data will be automatically backed up daily.

Fit Criteria: Developers will verify that the cloud database is backed up daily.

IR4: Unsaved data will be stored locally on the user's device if it cannot be uploaded to the remote database and is not explicitly discarded.

Fit Criteria: Developers will run tests in user role to ensure that unsaved data is stored locally on the user's device if it cannot be uploaded to the remote database and is not explicitly discarded.

4.6.3 Privacy Requirements

PRR1: Users will not be able to access data generated by other users.

Fit Criteria: Developers will run tests in user role to ensure that users cannot access data generated by other users.

PRR2: Users will be required to register and login to the application with their emails.

Fit Criteria: Developers will run tests in user role to ensure that users cannot access data generated by other users.

4.6.4 Audit Requirements

AUR1: Requirements should be easy to read and verify against the system facilitate regular inspections.

Fit Criteria: SRS requirements will be verified by SE4G06A teaching assistants.

4.6.5 Immunity Requirements

N/A

4.7 Cultural and Political Requirements

4.7.1 Cultural Requirements

CR1: The product will not use any terms that are inappropriate or offensive towards any cultures.

Fit Criteria: Volunteers will be invited to use the application and see if they find any inappropriate terms.

4.7.2 Political Requirements

N/A

4.8 Legal Requirements

4.8.1 Compliance Requirements

CPR1: The product must state that the application does not take responsibility for any accidents resulting from its use.

Fit Criteria: Developers will do user tests on application; when the application starts, application will pop up a window to ask the user to agree on the statements, and in the statements it must state that the application does not take responsibility for any accidents resulting from its use.

4.8.2 Standards Requirements

SDR1: The product must follow the fail-safe-defaults design principle in that it will always error into a safe state.

Fit Criteria: There should be error handling in the application to catch any possible errors. Every time before a developer pushes the changes to GitHub, the developer must test the application to ensure the application is in error-safe state.

SDR2: The product must follow modular design principles to allow for easy maintenance.

Fit Criteria: The project will be split into different modules in later capstone documents, and every developer should follow the modular design in the document to implement the application.

SDR3: The application must be developed using iOS application style guides.

Every time there is a pull request on GitHub, the team should run the application in XCode to see if it violates the iOS application style guides.

4.8.3 Health and Safety Requirements

N/A

5 Likely Changes

LC1: This project will only result in the development of a mobile iOS app. However, an android application may also be developed to work with, or in place of, the iOS application in the future.

LC2: The application may be extended to offer compatibility as a web application.

LC3: The application will not initially pursue functionality for users to "cut" their 3D simulated hair.

LC4: The application may require more data for the facial recognition feature.

LC5: The load capacity of the system is initially set low for the first edition of this application. As demand for the application increases, the load capacity must also increase to support a larger user base.

LC6: The application may be required to include new features as discovered through use of the system in real-world scenarios.

LC7: The format of the input and output data of facial recognition, hair modification, hair salon recommendation system.

LC8: The interface style requirements might change since no metric has been provided for this by stakeholders.

LC9: The speed and latency requirements might change since no metric has been provided for this by stakeholders.

LC10: The precision and accuracy requirements might change since no metric has been provided for this by stakeholders.

6 Unlikely Changes

ULC1: Since several of the features might require exposing API endpoints, an offline-only version of the application is unlikely to be necessary.

ULC2: As the main demographic of this application is less computer literate, the ease of use of this application must remain a high priority consideration throughout the design of this system.

ULC3: The application is designed to ensure that data will only be modified when necessary.

ULC4: The application is designed to ensure that only admins and supervisors will be able to unlock locked resources.

ULC5: The application is designed to ensure that no inappropriate or offensive language towards any culture will be used.

ULC6: The application is designed to ensure compliance with the Data Privacy Act of Canada. All user personal information will not be stored or used without consent.

ULC7: The application is designed to ensure that users are only able to see data that they have proper authorization for.

ULC8: The application is designed to help users simulate hairstyles and hair colors virtually.

ULC9: The primary user of this project is unlikely going to change since the project purpose is determined already, only certain types of users will be interested in this kind of application.

ULC10: The client of this project is unlikely going to change because this project is a student capstone project and it has limited marketability.

ULC11: The schedule and budget constraints are not likely going to change, because this project is a student capstone project. It needs to strictly follow SE4G06 course constraints.

7 Project Issues

7.1 Open Issues

One of the issues is that the team does not have sufficient background knowledge in technologies we are going to use: AR, Computer Vision, and iOS development. Developers must learn these technologies before implementation of the application.

7.2 Off-the-Shelf Solutions

Until now, there are a few similar open source projects online for us to go through. Since they are all written in different coding styles and design structures, we will only be gathering the basic concepts from these projects. Besides, to ensure that everyone in our team will be at the same pace and participate in the actual project, everyone in our team will take some time to play around with ARkit and swift before getting our hands on the actual design.

7.3 New Problems

7.3.1 Effects on current environment

1. The application will occupy the camera of the iOS mobile system when the application runs, and other applications in the same system can not use the camera function during this time.

2. The application will occupy some memory in the iOS mobile system for it to run.

7.3.2 Potential User Problem

1. Since pictures of users' faces will be saved in the system database, any potential data breach will exposes user privacy.

2. If the user uses the application under strong sunlight or in a dark environment, the user can potentially damage the eyes.

3. If the user uses the application when crossing traffics, there is a potential of traffic accident.

7.3.3 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

1. The back-end server may not be powerful enough to handle high volume of requests from users.

2. The size of the application may be too large to install in a normal iOS mobile.

7.3.4 Follow-Up Problems

1. Will we create a demand for our product that we are not able to service?

2. Will the new system cause us to run afoul of laws that do not currently apply?

3. Will the existing hardware on users' iOS mobile cope?

7.4 Tasks

Table 2: **Phase In Plan**

Task	**Timeline**
Discuss & Revise requirements	Oct 5th
Submit SRS	Oct 5th
Initial Draft of Hazard Analysis	Oct 12th
Revision of Hazard Analysis	Oct 18th
Submit Hazard Analysis	Oct 19th
Initial Draft of VnV Plan	Oct 24th
Revision of VnV Plan	Nov 1nd
Submit VnV Plan	Nov 2nd

Table 3: **Project Implementation Tasks**

Task Name	**Description**
Implement basic UI	An iOS UI that can access the camera and get camera inputs and pass it to back end
Implement Computer Vision	A component in the back end that receives camera inputs and use computer vision technique to analyze human face and output to the front end
Implement AR functionality in UI	A component in the front end that takes the analyzed output from computer vision in back-end and apply AR technique to add hairs on the user's face on the screen
Implement Database	A database to store user information
Implement Database interaction on UI	Add a component to fulfill read and write access to database

7.5 Migration to the New Product

N/A

7.6 Risks

Some of the testing process will be difficult to automate, since the product will need actual human to judge whether the hairstyle fits smoothly on the 3D model. For this part of the testing, we will draw random samples and ask people around us to perform physical testing.

7.7 Costs

Other than the time it takes to develop the application there might be a cost for using some licensed libraries.

7.8 User Documentation and Training

7.8.1 User Documentation Requirements

1. A readme file will be provided in the documentation for user to read. The file will provide a full guideline of how to install and use our application.
2. A technical specification will be provided for other developers who would like to make further changes to our application.
3. The guideline will also be available on the help page of the running application.

7.8.2 Training Requirements

N/A

7.9 Waiting Room

The application shall allow users to cut and make changes to their hairstyles, in order to enhance user experiences.

7.10 Ideas for Solutions

- Add function in computer vision to capture user hand gestures, with the specific gesture motion on a specific area of hair, the hair beneath the hand will be cut.

8 Traceability Matrix

	FR1	FR2	FR3	FR4	FR5	FR6	FR7	HM1	HM2	HM3	HM4	HM5	HM6	HM7	HM8	HM9
APR1																
APR2																
IAR1																
IAR2																
EUR1	X			X	X		X									
EUR2		X	X		X		X									
LR1																
LR2																
UPR1																
ASR1																
SLR1	X											X		X		
SLR2												X		X	X	
SLR3										X		X			X	
PAR1											X		X			
PAR2												X		X		
RAR1																
RAR2																
RBR1																
RBR2																
CCR1																
CCR2																
SER1																
LGR1										X			X			
PDR1																
RER1																

Table 4: Traceability Matrix Showing the Connections Between Functional Requirements and Non-functional Requirements

	FR1	FR2	FR3	FR4	FR5	FR6	FR7	HM1	HM2	HM3	HM4	HM5	HM6	HM7	HM8	HM9
RER2																
MSR1																
MSR2																
MSR3																
ADR1																
ACR1																
ACR2																
ACR3																
ACR4																
IR1																
IR2																
IR3																
IR4																
PRR1																
PRR2																
AUR1																
CR1																
CPR1																
SDR1																
SDR2												X				
SDR3																
RER1																

Table 5: Traceability Matrix Showing the Connections Between Functional Requirements and Non-functional Requirements

	HR1	HR2	HR3	HR4	AR1	AR2	AR3	AR4
APR1								
APR2								
IAR1								
IAR2								
EUR1	X	X						
EUR2	X		X	X				
LR1								
LR2								
UPR1								
ASR1								
SLR1								
SLR2								
SLR3								
PAR1								
PAR2								
RAR1								
RAR2								
RBR1								
RBR2								
CCR1								
CCR2								
SER1								
LGR1								
PDR1								
RER1								

Table 6: Traceability Matrix Showing the Connections Between Functional Requirements and Non-functional Requirements - Continue

	HR1	HR2	HR3	HR4	AR1	AR2	AR3	AR4
RER2								
MSR1								
MSR2								
MSR3								
SPR1								
ADR1								
ACR1						X		
ACR2					X	X	X	X
ACR3					X	X		X
ACR4					X	X		
IR1								
IR2								
IR3								
IR4								
PRR1								
PRR2								
AUR1								
CR1								
CPR1								
SDR1								
SDR2								
SDR3								

Table 7: Traceability Matrix Showing the Connections Between Functional Requirements and Non-functional Requirements - Continue

Appendix — References

- templates: https://gitlab.cas.mcmaster.ca/smiths/pub/-/blob/master/VolereTemplate16.doc
- J. Cook, "Converting color to grayscale," John D. Cook — Applied Mathematics Consulting, 29-Apr-2019. [Online]. Available: https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/.
- Inc., A. (n.d.). Designing for IOS. Designing for iOS - Platforms - Human Interface Guidelines - Design - Apple Developer. Retrieved October 5, 2022, from https://developer.apple.com/design/human-interface-guidelines/platforms/designing-for-ios/

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain-specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

   Upon completing your project, all members will learn more about writing software development documentation and prototype presentation. In this project, ARkit is a new tool for everyone. As a result, after completing this project, we all should be familiar with the technical knowledge which can help us to build more AR-related applications.

   Although Charlotte and Senni had some prior experience with swift, this project will be their first time developing a serious iOS application, not only will they have the opportunity to implement application UI/UX, but they will also learn more about swift which will help them gain insights about iOS/mobile application development. Although Jack had some experience with database development, such as using SQL and MATLAB, this is the first time he take the challenge to deal with project-related data. For sure he will also become better at data processing after this project. Last but not least, Marlon, Hongwei and Bill will be the pioneers to adventure machine learning using OpenCV with regard to facial recognition detection. They will learn about many machine learning concepts. During the whole development, all team members will also share their knowledge about the new technology they have learnt to help each other become better engineers, and also the things they already know to help each other

during the development. Besides, all members will practice the design pattern we have learned in previous years and improve our leadership, communication and cooperation skills.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?

   (a) For swift programming learning. The first approach will be gathering information from a book called "Develop In Swift". And applying the knowledge they learned to our project will help them to master Swift. The second approach would be from the other team members who had some knowledge about it. In this way, Senni and Charlotte will get familiar with the basic syntax in a short time.

   (b) For database developing, Jack will be adventuring pandas from python and following the tutorial online to learn the basic functions of it. After that, he will also learn from the actual developing process to put the knowledge into physical development.

   (c) For back end and AR developers, they will first go through a few examples on github to get the basic concept about how to use a machine learning model. And then during the actual project building, they will also learn along the development and gather all the information needed to build a successful AR platform.

   (d) For teamwork skills mentioned above, we will first take turns to be the leader and host the weekly meetings during the development. Besides, we will also follow an agile standard during the development to enhance our teamwork and cooperation skills.