Software Architecture Design

Secure Face ID v1.0a

Team : 5verFlow

Date : 8 Jun 2021

[**Introduction**](#_g6hqo9h7mxcw) **4**

[Purpose](#_9dspdbpxzqkz) 4

[Scope](#_kf8oqofpompa) 4

[Audience](#_sjqvg6da3fv) 4

[Conventions](#_1hk26h3omfp) 4

[Abbreviations / Terms](#_eavfbahgszhl) 4

[Related Documents](#_sruzge3my9sr) 4

[Revision History](#_xu4he5in8h5r) 4

[**Overview**](#_yg9wm9u2octy) **5**

[**Architectural Drivers**](#_mpq9k9isuxnp) **5**

[SW Main Features](#_het4cxnqp0nk) 5

[Quality Attributes](#_quf69y2u6hio) 6

[Constraints](#_fizt49bga0s) 6

[Technical Constraints](#_9al85ya3tk01) 6

[**SW Architecture**](#_uii11sbmta2b) **6**

[Static View](#_jd1ldsw2vyn9) 6

[External Interfaces of Server Application](#_uof01nrb17xt) 8

[Image Handler](#_wqaehovbyoc2) 8

[Camera Controller](#_6f15v8feqlpo) 8

[Data Manager](#_e826hnj58udu) 8

[Connection Manager](#_ow52y5nml8kq) 8

[Log Manager](#_qsrw75jjm9r) 8

[External Interfaces of Client Application](#_qfjqi8232psq) 9

[GUI Manager](#_7g2r1b03ylv) 9

[Command Handler](#_tj5jsz7gbpw4) 9

[Error Report](#_n2bqzig1levx) 9

[Communication Manager](#_yqgqvvh19hu8) 9

[Connection Manager](#_yqgqvvh19hu8) 9

[SW Module List](#_q6u7ujofjaqt) 9

[Dynamic View](#_2ytggiimjo4r) 11

[Task Design](#_2ytggiimjo4r) 11

[State Design](#_2ytggiimjo4r) 11

[Communication Design](#_pnv6ctaw8nko) 11

[Protocol Definition](#_mol24ac0iz1) 11

[Message Sequences](#_dish0qytb03y) 12

[Request Connection](#_rxg5qoz858e) 12

[Request Disconnection](#_2jdgqdwitpdd) 13

[GUI Design](#_2ytggiimjo4r) 14

[**Resource Consumption Objectives**](#_l20hy4pa6h3l) **14**

[**Failure Design**](#_l20hy4pa6h3l) **14**

[reconnecting when unexpected disconnection between server and client](#_9nm76jnop00t) 14

[respawn server application when unexpected terminated](#_oj4cv83sd4vl) 14

[**Security Design**](#_l20hy4pa6h3l) **14**

[Input Validation](#_tvdjugkrtwm3) 14

[Encrypt Data and Message](#_tvdjugkrtwm3) 14

[2-factor Authentication](#_tvdjugkrtwm3) 14

[**Appendix**](#_6nmaka5qqbvi) **14**

# 

# Introduction

## Purpose

TBD

## Scope

TBD

## Audience

TBD

## Conventions

TBD

## Abbreviations / Terms

TBD

## Related Documents

1. User Requirement - Dan
2. Project Requirement Specification : Secure Face ID v1.0a
3. Software Requirement Specification : Secure Face ID v1.0a

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **History** | **Date** | **Author/**  **Reviewer** |
| v0.1a | Initial draft |  | 5verFlow |
|  |  |  |  |
|  |  |  |  |

# 

# Overview

The Secure Face ID consists of Jetson Nano server and Laptop client and they are connected over the internet via TCP/UDP.



Figure. Context Overview

The camera captures video conference attendees and the server analyzes it using MTCNN and FaceNet. The CUDA GPU is utilized during image processing. The image and data are transferred to the client and the user sees the image and data at the client laptop.



Figure. System Overview

# Architectural Drivers

## SW Main Features

|  |  |  |
| --- | --- | --- |
| **ID** | **Level-1** | **Level-2** |
| MF-1 | Application Login |  |
| MF-2 | Live Mode (Run Mode) |  |
| MF-3 | Playback Mode (Test Mode) |  |
| MF-4 | Register Mode (Learning Mode) |  |
| MF-5 | Reinitialize DB |  |
| MF-6 | Error Report |  |

## Quality Attributes

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Quality Attribute** | **Scenario** | **Priority** |
| QA-01 | Performance | The system must show video as close to real time as possible. | 3 |
| QA-02 | Security | The system must be secured from any threats as specified. | 1 |
| QA-03 | Reliability | The system should provide tolerance of network failure thus it could transfer images continuously. | 2 |

## Constraints

### Technical Constraints

|  |  |
| --- | --- |
| **ID** | **Description** |
| TCN-01 | The server allows only one client connection. |

# SW Architecture

## Static View

The Secure Face ID uses many open sources to handle images and connection. only the application is made by us.



Figure. Layered Software Architecture



Figure. 2-Factor Authentication and Data Encryption



Figure. Server Software Block Diagram

## External Interfaces of Server Application

### Image Handler

* TensorRT :
* CUDA Toolkit :
* OpenCV :

### Camera Controller

* Camera Device :
* GStreamer :
* OpenCV :

### Data Manager

* TensorRT :
* Cryptmount:

### Connection Manager

* OpenSSL :

### Log Manager

* Cryptmount:



Figure. Client Software Block Diagram

## External Interfaces of Client Application

### GUI Manager

* QT
* OpenCV

### Command Handler

### Error Report

* QT

### Communication Manager

### Connection Manager

* OpenSSL

## SW Module List

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Level-1** | **Level-2** | **Level-3** |
|  | Server Application | Image Handler (차승욱, SW Cha) | OpenCV |
|  | CUDA Toolkit |
|  | Camera Controller (차승욱, SW Cha) | GStreamer |
|  | Face Analyzer (한동혁, DH Han) | MTCNN Modeler |
|  | FaceNet Modeler |
|  | Data Manager (최유경, YK Choi) | Video File Storage |
|  | Image File Storage |
|  | Data File Storage |
|  | Certificate File Storage |
|  | Secure Storage Manager |
|  | Server Communication Manager (이영진, YJ Lee) | PDU Handler |
|  | Message Handler |
|  | Transfer manager |
|  | Server Connection Manager (한동혁, DH Han, Session Control은 집단지성) | Session Manager |
|  | TLS Manager |
|  | Log Manager (이영진, YJ Lee) | Log Messages |
|  | Client Application | Client Communication Manager (이성준 SJ Lee, 강우람 WL Kang) | PDU Handler |
|  | Message Handler |
|  | Transfer Manager |
|  | Client Connection Manager (한동혁, DH Han) | Session Manager |
|  | TLS Manager |
|  | Certificate File Manager |
|  | GUI Manager (강우람, WL Kang) | Control Window |
|  | Video Window (OpenCV) |
|  | Register Dialogue Window |
|  | Error Report |
|  | Command Handler (강우람, WL Kang) |  |

## Dynamic View

## Task Design

## State Design

## Communication Design

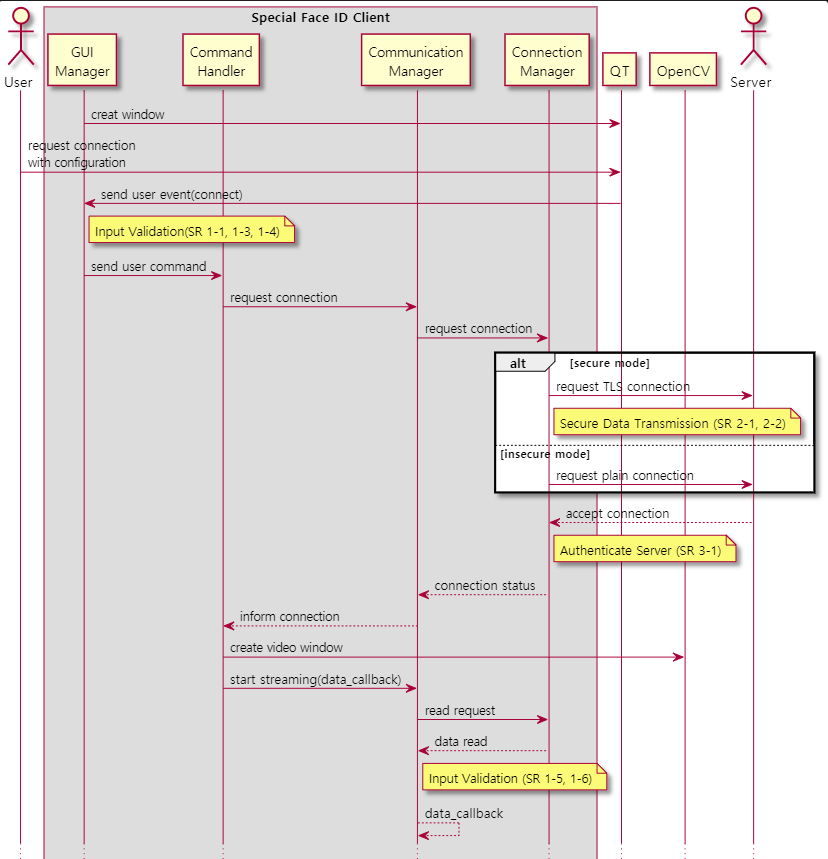
### Protocol Definition

Data from server to client

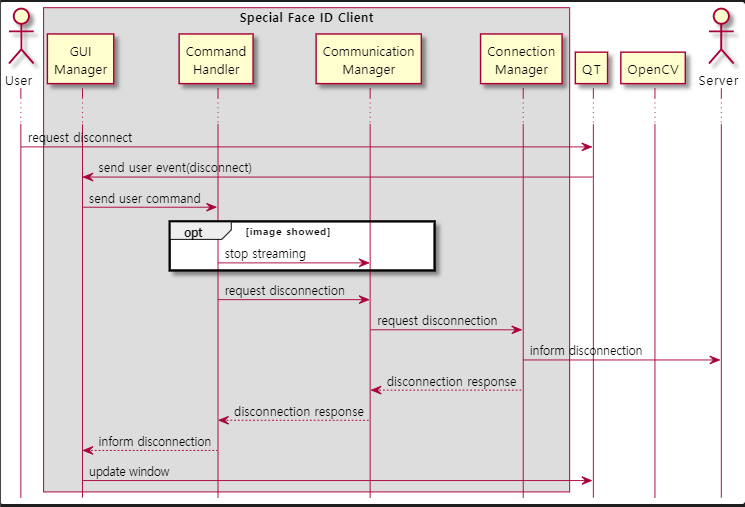
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Total  Length |  | Image Length | Face Data | Image Data |
| Byte Length | 4 | 4 | Total Length - Image Length | Image Length |
| Total Length | | |

### Message Sequences

#### Request Connection



#### Request Disconnection



## GUI Design

# Resource Consumption Objectives

# Failure Design

## reconnecting when unexpected disconnection between server and client

## respawn server application when unexpected terminated

# Security Design

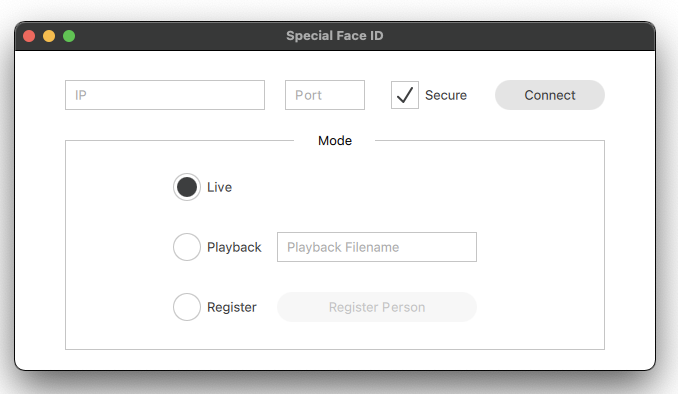
## Input Validation

## Encrypt Data and Message

## 2-factor Authentication

# Appendix

Draft version of GUI



- The images from the server are to be shown on the other window.

- When the “Register Person” button is clicked, a new window pops up and users can write the name of the detected face.