Automated IC Card Dispenser System for Residential College

By

Zhirong Chen (zhirong4@illinois.edu)

Xiaoyang Chu (xzhu458@illinois.edu)

Zicheng Ma (zma17@illinois.edu)

Dongshen Ye (dye7@illinois.edu)

Sponsored by

Asst. Prof. Meng Zhang

Project #1

Project Proposal for ECE445/ME470, SP2024

Mar 27th, 2024

Contents

1 Introduction

1.1 Problem

According to our investigation through the records of the residential college, there are 287 cases where a resident requests for a temporarily access card to their own dormitory room because they accidentally leave their card in the room during the period of Feb. 25th, 2024, to Mar. 3rd, 2024. The number didn't include the cases after the office hour, when the process is especially inconvenient as one must contact the security personals for help. Due to the cosmopolitan nature of the students on campus, there could be confusion in the communication during the interaction with the security personal. From the perspective of safety, the current procedure of temporary card issuance poses the risk of unauthorized breach as usually the person made the request is not fully identified.

1.2 Solution & Visual Aid

Our solution to the problem is an automated system consisting of a server and several KIOSK terminals.

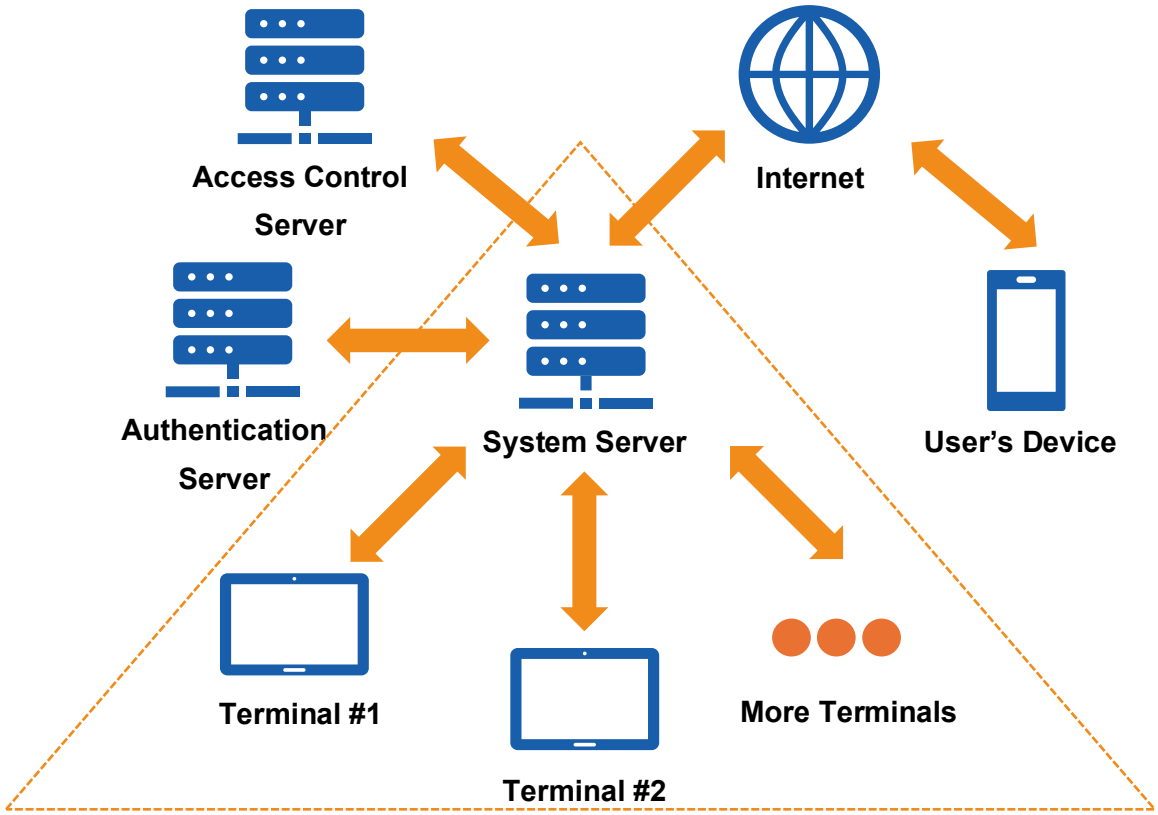

Fig. 1 Solution Structure

The terminal will support two ways of authentication. The first way is through a QR code which user obtains through online authentication as shown in Fig. 2.1 below.

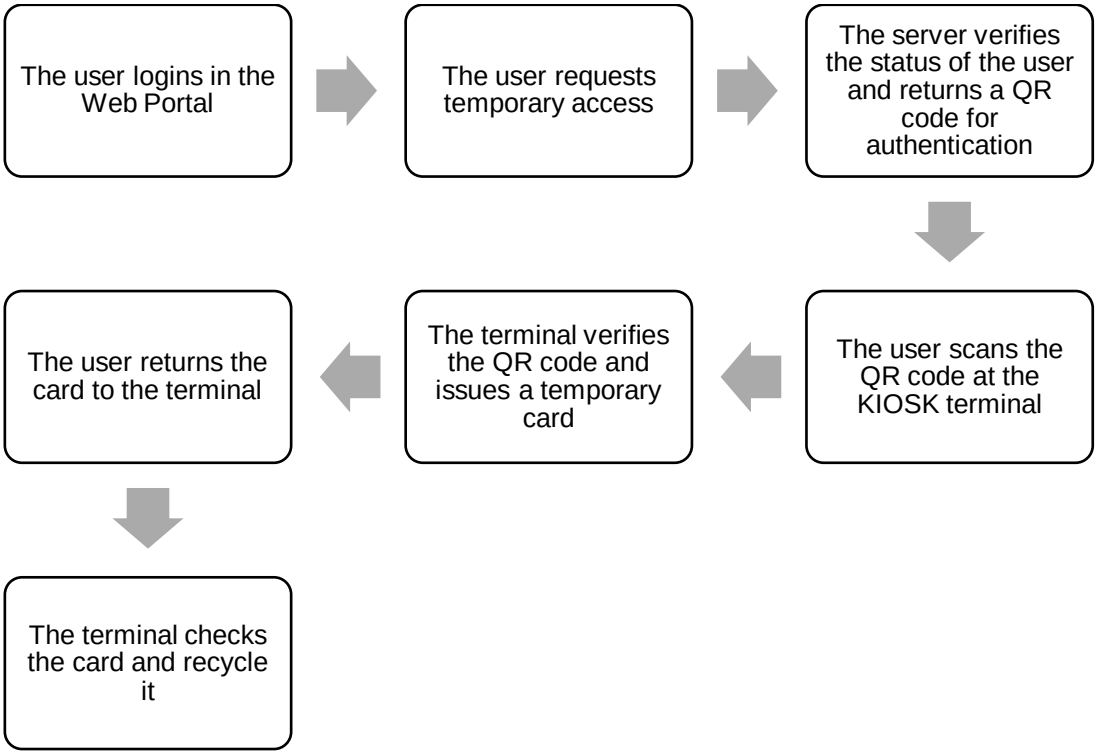

Fig. 2.1 Workflow of the System (Web-QR Code Authentication)

The other method is through facial recognition shown in Fig. 2.2.

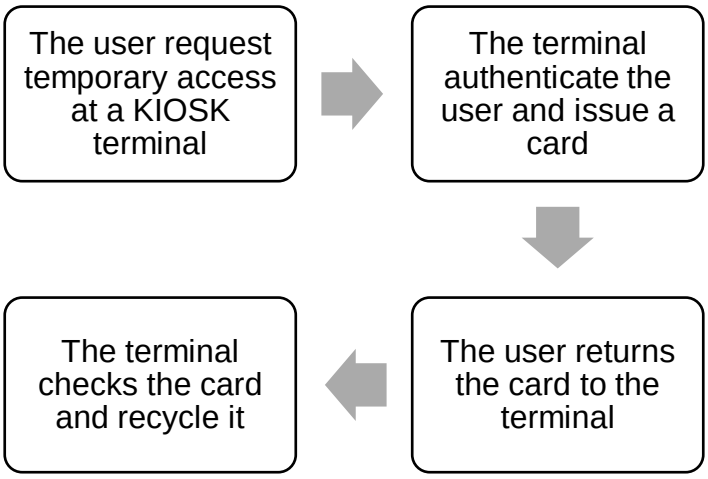

Fig. 2.2 Workflow of the System (Facial Recognition Authentication)

For safety concerns, the system will not directly interface with the existing offline access control systems in the Residential College. So, a user must register

themselves to the service before they can use it. The registration process is illustrated in Fig. 3.

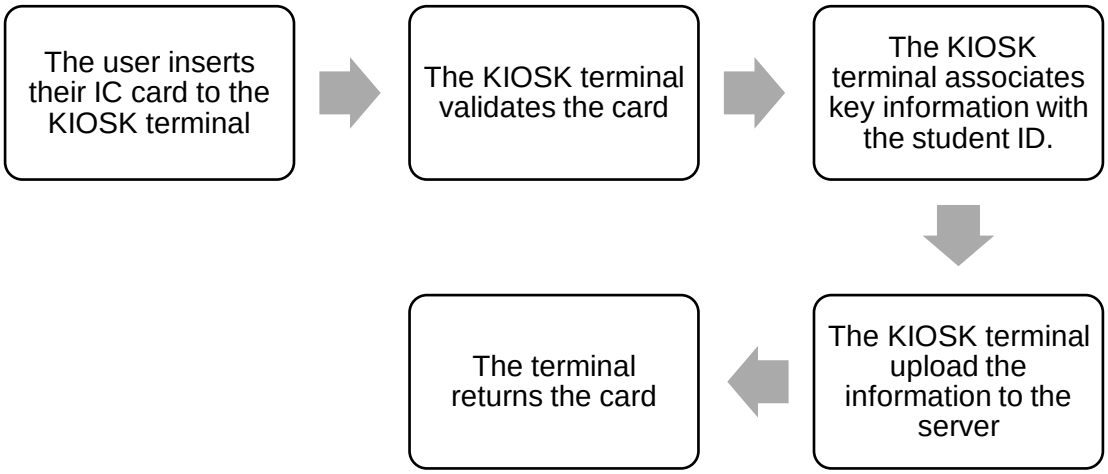

Fig. 3 Workflow of the System (Facial Recognition Authentication)

1.3 List of High Level Requirements

To solve the problem, the following high-level requirements should be met.

- Reliable, robust, and convenient authentication methods should be adopted to keep the issuance process secure. The system should be invulnerable to conventional cyber-attacks. A successful issuance process should take less than 60 seconds.
- The mechanical card dispenser should have failure rate less than 1/500.
- The system should support multiple languages and can be easily maintained and managed.

2 Design

The entire project consists of two major parts: KIOSK terminal and server.

2.1 Block Diagram

2.1.1 KIOSK Terminal Block Diagram

The KIOSK Terminal consists of two subsystems: the user interaction subsystem and the mechanical subsystem. The composition of the KIOSK terminal is shown in Fig.

4.

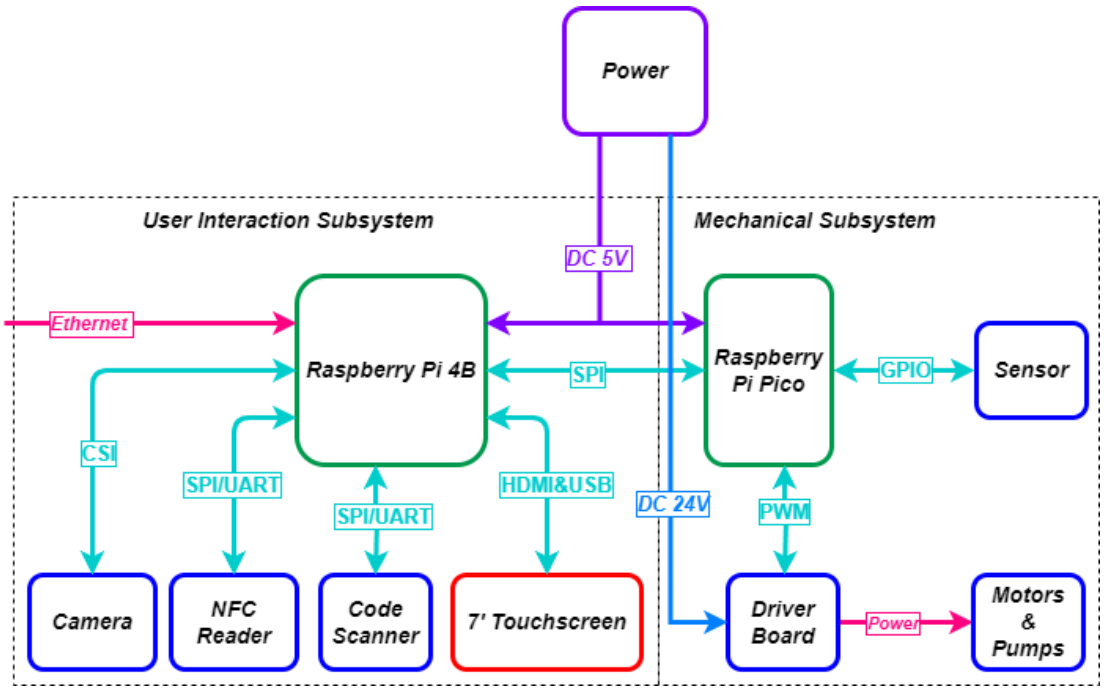

Fig. 3 Block Diagram of KIOSK Terminal Hardware

The client-side software will run on Raspberry Pi 4B which interact with the user and control the mechanical subsystem. The hierarchy of the client-side software is shown in Fig. 5.

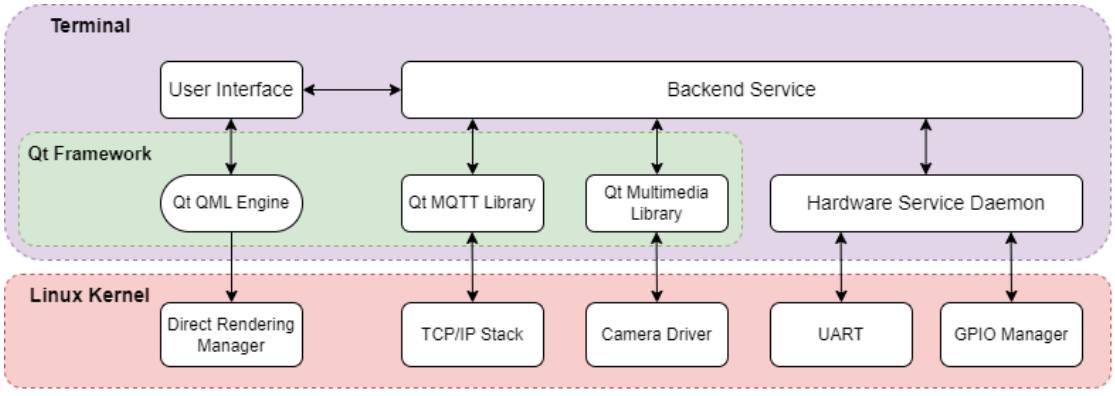

Fig. 4 Block Diagram of Client-side Software

2.1.2 Server-side Software Block Diagram

The server-side software will process requests and manage the web service. The two software subsystems will communicate via the LAN on campus. The composition of the software is shown in Fig. 6.

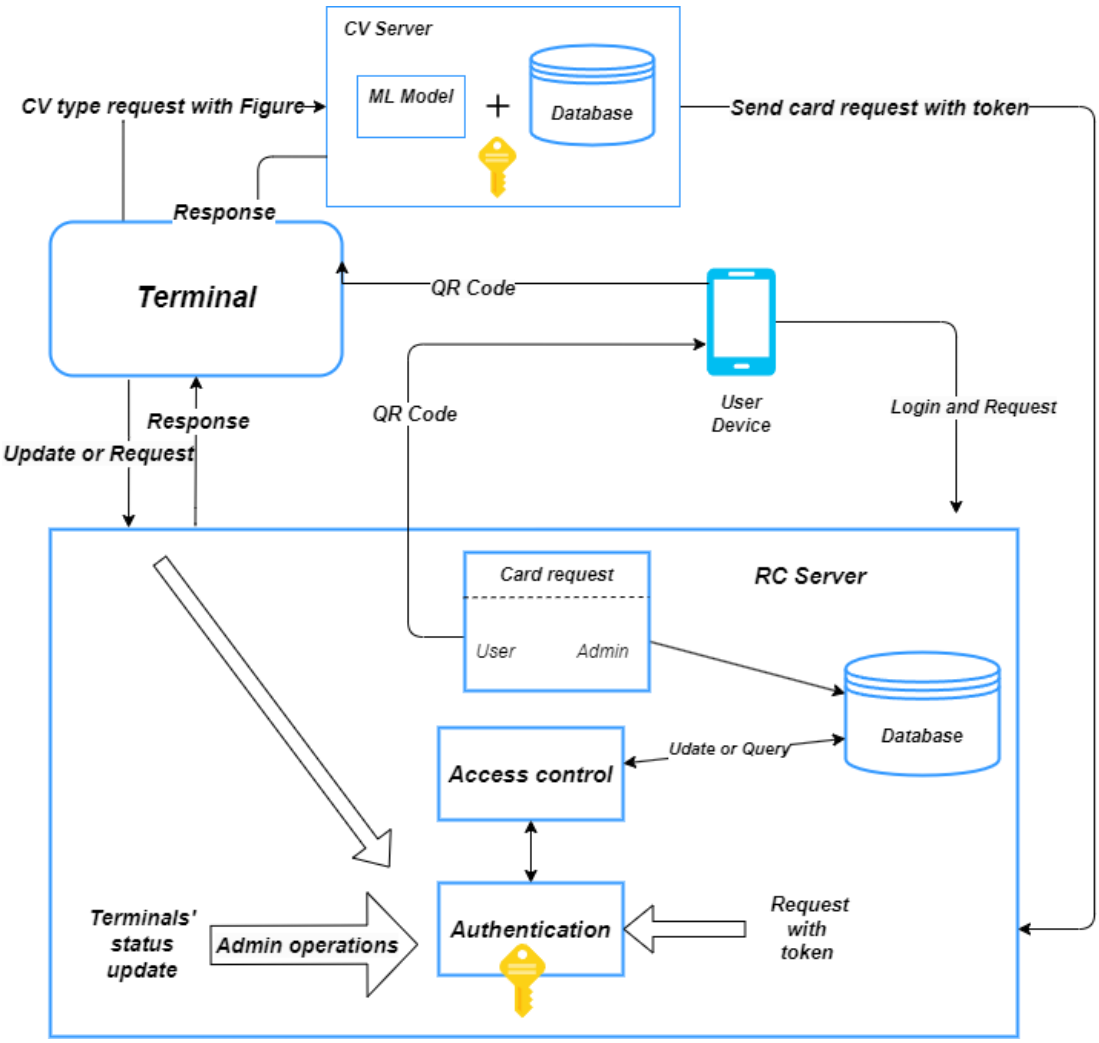

Fig. 6 Server-side Software Block Diagram

2.2 Physical Design

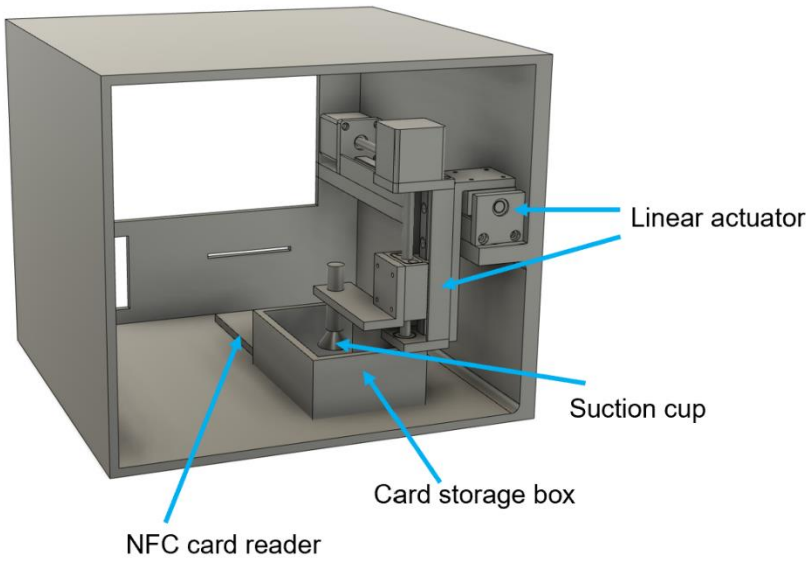

Fig. 7 CAD model of the card dispenser

Fig. 7 shows the CAD model of the card dispenser, especially the design of the mechanical subsystem. The mechanical subsystem mainly consists of a suction cup and two linear actuators. The suction cup is used to pick up IC cards from the card storage box. And the two linear actuators are adopted to move the cards along two orthogonal axes.

2.3 KIOSK Terminal

2.3.1 User Interaction Subsystem Overview & Requirements

Overview

The user interaction subsystem consists of a Raspberry Pi 4B computer and peripheral devices including a camera for facial recognition, an NFC reader for IC card communication, a code scanner for QR code input and a 7-inch touchscreen for user to interact with. The mechanical subsystem will consist of a microcontroller, Raspberry Pi Pico, several sensors for monitoring status of cards or transmission system control, and driver boards. The user interaction system will communicate with mechanical subsystem through SPI protocol to issue or recycle the cards.

Requirements

The User Interaction subsystem should be capable of handling user input/output properly and efficiently. The following requirements should be met before the software for KIOSK can function.

Requirements	Verification
1. Medium rendering capabilities 2. Medium computational power 3. Capable of collecting visual information and QR code information 4. Compatible with the mechanical control subsystem	1. Run standard GPU benchmark. 2. Run standard processor benchmark. 3. Run camera testing software. 4. Run GPIO testing software

2.3.2 Terminal-side Software Overview & Requirements

Overview

The terminal-side software is responsible for collecting information from the user, including information required for the authentication process, e.g., the QR code and facial information. The software should also update the status of the machine to the RC server and allow simple remote maintenance. Most importantly, the software should be able to interact with users effectively.

Requirements

The requirements of client system can be summarized as follows:

Requirements	Verification
1. Effective Human-Computer Interaction: Simple user interface with multiple language support 2. Continuous Internet Connectivity Requirement: The client system must maintain an uninterrupted connection to the Internet to facilitate ongoing communication with the server, including the transmission of requests and the reception of responses. 3. Immediate Timeout Error Handling: In scenarios where the client system experiences a loss of Internet connectivity, it is imperative that a timeout error message is promptly communicated to the associated	1. Hire volunteers for pilot testing and collect feedback from them. 2. Monitor the terminal status for at least a week and record the Internet connection anomaly. 3. Test the terminal under unstable Internet condition and record any anomaly of the software. 4. Joint testing with the mechanical subsystem and test the efficiency of the relay.

hardware to signal the disruption in service. 4. Responsive Instruction Relay: Upon successfully receiving a card sending instruction from the server while connected to the Internet, the client system is obligated to efficiently relay these instructions to the designated hardware, ensuring timely execution of commands.	

2.3.3 Mechanical Subsystem Overview & Requirements

Overview

Mechanical Subsystem is responsible for issuing and recycling IC cards. Mechanical Subsystem's job is to circulate IC cards among three places, card storage box, NFC reader and the exit of the card dispenser. Apart from connecting with power supply, it receives the commands of dispensing cards from User Interaction Subsystem and report information like the number of cards in card storage box and whether there is a card at the exit to User Interaction Subsystem.

Requirements

Mechanical Subsystem is responsible for transferring IC cards to realize dispensing and recycling and monitoring the status of IC cards. It should satisfy the following requirements:

Requirements	Verification
1. Accurate card grabbing: Mechanical system should pick exactly ONE card from a stack of IC cards. 2. Transmission of cards: Mechanical system is supposed to take cards out of the storage box, send cards to the NFC reader for information processing, send cards to and accept cards from the users. 3. Accurate card alignments: The cards should not be stuck by the structure and the card reader can detect the cards.	1. Accurate card grabbing: a. Put a stack of cards in the card storage box, place the suction cup on the upper surface of the top card. b. Turn on the vacuum pump and lift the suction cup to ensure that it picks up only one card. 2. Transmission of cards & 3. Accurate card alignments: c. Move the card from the storage box to the NFC reader and let the card reader read the card. d. Then move the card from the card

	reader to the exit. e. Also let the suction cup grab the card from the exit to the card reader, let the card reader read the card, then move the card to the storage box. f. Repeat a to c 10 times, check if there is any case of card stuck, card fallen from the suction cup, or if card cannot be detected by the card reader

2.4 Server-side Software

2.4.1 Authentication Subsystem

The Authentication Subsystem is a critical component of our software system, designed to ensure the integrity and legitimacy of every request made to the system. This subsystem employs advanced cryptographic techniques, including SHA-256 for token validation, TLS (Transport Layer Security) for secure communication, and digital certificate management through a Public Key Infrastructure (PKI). Upon receiving a request, the subsystem verifies the associated token against a repository of known valid tokens stored within a secure server environment. This process is essential for preventing unauthorized access and safeguarding against potential system abuse. The Authentication Subsystem serves as the primary defense mechanism against external threats, interfacing directly with both the Hardware Subsystem and Access Control Subsystems to maintain a robust security framework. We are searching for a protocol that is suitable for our task now. MQTT [1] is probably a good choice.

Requirements	Verification
1. Cryptographic Validation: Must utilize a secure encryption method (e.g. SHA-256) to validate tokens with a processing time not exceeding 2 seconds per request, ensuring swift and secure access. 2. Digital Certificate Management: Requires a Public Key Infrastructure (PKI) to publish certificates for different terminals.	1. Cryptographic Validation: a. We built a simple QR code server and tested how long it will take to encrypt an information segment. The cost was about 0.01ms. b. The hardness of crack SHA-256 algorithm has been validated in SHA-512/256 paper[6]. 2. TTL Validation: a. We adopt HTTPS as the TTL protocol.

3. Transport Layer Security (TLS): May implement TLS for data transmissions, to ensure that the confidentiality and integrity of the communication are maintained. But this is not as necessary as secure encryption. 4. User login authentication: This part will be work on above ZJU intl authentication system. We need to know how the APIs work.	The transportation speed stays the same compared with HTTP during our test. 3. User login validation: a. This functionality works based on ZJU intl authentication system, so the verification is their(and Microsoft) responsibility, and we trust them.

2.4.2 Access Control Subsystem

The Access Control Subsystem functions as the regulatory framework within our software architecture, closely integrated with the system's database which archives user credentials and hardware metrics. Its chief role is to administer the issuance of cards to users, contingent upon card availability and the user's compliance with borrowing protocols. By interfacing with the Authentication Subsystem, it ensures legitimacy and adherence to administrative regulations, thereby upholding the system’s capacity and policy constraints.

Requirements	**Verification**
1. Request Handling: The subsystem must validate card availability against the database within a 1-second response threshold to guarantee prompt resource distribution. 2. User Status Check: It is essential for the subsystem to discern user eligibility for card issuance through database queries, factoring in their borrowing chronicle and current standing. 3. Database Interaction: Synchronization with the database must be instantaneous to inform Access Control decisions with real-time data. 4. Policy Adherence: Automated enforcement of borrowing regulations is mandated, with the system being capable of updating user records within 3 minutes after any policy	1. Request Verification: a. With the simple QR code server we built, we conducted timed trials to validate response times using a high-resolution timer. The result was pretty good with about less than 0.05ms response time. 2. Database Interaction Verification and Policy Adherence: a. We have deployed a MySQL server and write some simple rules to test. b. Currently, we have three columns: Name, ZJU ID, Card NO. and the test has successfully passed.

amendments.	

2.4.3 Facial Recognition Subsystem Overview & Requirements

Overview

The facial recognition server is responsible for receiving the face picture from the client system. It uses the ML model and face dataset to judge which student the face belongs to. Some famous ML models include Deepface [2], FaceNet [3] and VGGFace [4], etc. If the face is not matched to any faces in the dataset, the server needs to return error message to the client immediately, otherwise, it will send the request with user information to the RC server to request a temporary card for the student.

Requirements	Verification
1. Given a face image, the face recognition subsystem should recognize whose face it is, with error rate lower than 0.5% 2. The face recognition subsystem should have a good performance. The whole recognition period should be less than 0.5s. 3. The face recognition subsystem should safely store the information of users and prevent malicious network attacks.	1. Verification Stage 1: In stage 1, we should guarantee the functionality correctness. Given a face image, we should make sure the subsystem can return a correct result (user id). 2. Verification Stage 2: In stage 2, we should try to improve the subsystem's performance. Specifically, we should speed up the subsystem, and make the latency of the face recognition subsystem less than 0.5s for recognizing a face. 3. Verification Stage 3: In stage 3, we should make sure the safety of the subsystem. We can simulate the malicious network attacks, such as sending thousands of requests to the subsystem in a short time and test whether the subsystem will collapse.

2.5 Tolerance Analysis

2.5.1 Facial Recognition Subsystem Tolerance Analysis

Definitions in the Context of Facial Recognition:

True Positive (TP): The system correctly identifies a face that is in the dataset.

True Negative (TN) The system correctly determines that a face is not in the dataset (though in many facial recognition applications, this category is less commonly considered, as the focus is often on identifying known individuals rather than confirming unknowns).

False Positive (FP): The system incorrectly identifies an unknown face as someone in the dataset.

False Negative (FN): The system fails to identify a known face in the dataset, either by not recognizing the person or by misidentifying them as someone else.

Error Rate Calculation:

In this context, the error rate is primarily concerned with how often the system makes incorrect identifications (either by misidentifying individuals or by failing to identify them correctly). Therefore, we can calculate the error rate as the sum of false negatives and false positives divided by the total number of identification attempts:

$$ER = \frac{FP + FN}{TP + TN + FP + FN}$$

This formula gives us the proportion of all identification attempts that result in incorrect outcomes, whether those are misidentifications (FP) or failures to identify (FN).

Adjustment of Error Rate:

If the confidence score is lower than 90, the system will ask the client to take a photo again, and it will re-evaluate the picture.

To adjust the error rate calculation for a facial recognition system that requests a re-attempt when the confidence score is lower than 90, we need to consider how these re-attempts influence the probabilities of false positives (FP) and false negatives (FN).

A second photo may provide clearer evidence for correct identification, reducing instances where the system fails to recognize a person in the dataset.

While the primary goal is to reduce FNs, the quality of the new photo and the inherent uncertainty in any recognition attempt mean that FPs could also be impacted, though the direct intention is not necessarily to reduce FPs through this mechanism.

The adjusted error rate can then be considered as follows, taking into account the likelihood (L) that a re-attempt leads to a correct identification when the initial attempt was below the confidence threshold:

$$ER_{adjusted} = \frac{FP + \big(FN \times (1 - L)\big)}{TP + TN + FP + FN}$$

Where L represents the likelihood of correcting a FN on a re-attempt, which could be inferred from system performance data or empirical testing.

2.5.2 Card Numbers Tolerance Analysis

We have counted the students who borrow cards from the front desk in RC. We find that 287 cards were borrowed in 8 days, or about 36 cases of borrowing on average every day, during the office hour. We also notice that no more than 15 people can borrow cards at the same time. We assume the peak value of borrowed cards is 4 times the average number, so putting 20 cards in one automatic card dispenser would be enough for the requirement if 3 automatic dispensers are implemented at RC.

Based on observations, there are an average of 36 card requests every day. The peak concurrent requests observed is five. Assuming the peak borrow rate is four times the average, we would expect a maximum of 144 borrow requests per day.

A small server, with modest specifications, can typically handle hundreds to thousands of requests per minute. Given that our peak estimation is only 144 requests per day, this demand is well within the operational capabilities of a small server. Even we suppose our server is slow, which can handle 10 requests per minute, based on the calculations below, we can see the server can handle requests easily.

- *Peak request number per hour (assuming 8 active hours): 144 requests / 8 hours = 18 requests/hour*
- *Transactions per minute during peak hours: 18 requests / 60 minutes = 0.3 requests/minute*
- *Server load during peak = (Peak transactions per minute / Server capacity per minute) * 100*

- *Server load during peak = (0.3 / 10) * 100 = 3%*

Setting a safety margin, we can calculate the server's capacity to handle unexpected spikes. If a small server can handle 200 requests per minute, compared to our peak estimation of approximately 0.3 requests per minute (144 requests per day), the safety margin is substantial.

The current and projected peak volume of requests is significantly lower than the processing capacity of a small server. This indicates a high tolerance for request handling and suggests that even a small server can provide reliable service without risk of overload.

2.5.3 Suction Cup Tolerance Analysis

Mechanical Subsystem contains a suction cup powered by a small vacuum pump to pick up IC cards. It is crucial that the vacuum pump can provide sufficient suction force to keep the IC card stuck on the suction cup during transmission. Assume an IC card is m = 20 grams in mass, 50mmx100mm in size. We also assume that the suction cup used for the subsystem has a radius of r = 5 mm and the vacuum pump can produce a pressure of P = 45.45 x 10^3 Pa, which corresponds to the pressure provided by a vacuum pump with a vacuum degree of 45%. Given by the radius of the cup and the pressure of the vacuum pump, the suction cup can provide a force as follows,

$$F = P\pi r^2 = 3.57\,N$$

If the suction cup takes an IC card to move vertically upward with an acceleration of a = 0.05 m/s. It can be calculated that the force required is,

$$F' = m(g + a) = 0.1972N$$

where g is the acceleration of gravity. The value for the acceleration of gravity used here is 9.81 m/s^2. By comparing the force provided by the suction cup F and the required force F', the safety factor S can be calculated as below,

$$S = \frac{F}{F'} = 18.10$$

which indicates that suction force is strong enough to prevent IC cards from falling from the suction cup.

3 Cost and Schedule

3.1 Cost Analysis

Labor	**Quant.**	**Amount (CNY)**
Develop the face recognition subsystem	50 h	2000
KIOSK Software Development	75 h	3000
Build a QR code web server	40h	1200
Build database and import RC data	30h	1000
Build mechanical subsystem	40h	1200

Parts	**Quant.**	**Amount (CNY)**
Raspberry Pi 4B Development Kit	1	775.90
RFID Analyzing Tool	1	345.92
Vacuum pump and suction cup	1	30.39
XY actuator	1	480

3.2 Schedule

Mechanical Plan Test & Confirmation	2024/03/29
QR code Functional Test	2024/03/29

Facial Recognition Functional Test	2024/04/05
Microcontroller Software	2024/04/05
Mechanical Stability Test	2024/04/08
Network Comm Test	2024/04/08
KIOSK Terminal Test	2024/04/12
Web User Interface Functional Test	2024/04/12
KIOSK Terminal – Server Joint Test	2024/04/19
Pre–Demo Test	2024/05/06

4 Ethics and Safety

4.1 Safety of Authentication and Access control System

The safety concerns for the software system primarily relate to data protection, system reliability, and user privacy. Ensuring the safety of these aspects is critical, as breaches could lead to identity theft, unauthorized access, or service disruptions.

- User Privacy and Encryption: The system must maintain user privacy by ensuring that personal data is not exposed to unauthorized entities. This requires all data stored and transmitted will be encrypted using industry-standard cryptographic protocols to prevent unauthorized access. Therefore, we plan to adopt SHA-256, which is a relatively secure, difficult and costly encryption method in the current industry.
- System Reliability: The RC Server must offer high availability and fault tolerance to avoid service interruptions, which could lead to safety issues in systems that rely on constant connectivity for critical functions. Although the traffic and demand on campus may not be large, the stability and load capacity of the

server are also to be considered.

4.2 Reliability of Facial Recognition

It is guaranteed that the face dataset of students is safely stored in the facial recognition system. We will also make sure that when the camera take a picture of student, the picture will only be used for facial recognition, but not for other purpose.

4.3 Safety Concerns of Mechanical Systems

- Since mechanical system uses vacuum suction cup to grab IC cards, which involves the production and use of compressed air flow. It should be checked that the compressed air is processed properly so there will be no leak or explosions.
- It should be ensured that the mechanical subsystem will not hurt the users' hands when the user puts his or her hand at the exit of the card dispenser to pick up the card or return the card. When a user picks up the card, the user opens the door at the exit of the card dispenser to get the card. The suction cup and the linear actuators cannot move when the user opens the door.
- The mechanical subsystem should have the ability to recycle IC cards at the exit back if the users do not pick up the cards in 5 minutes after the cards are sent out. That function can prevent the risk that someone else taking away the cards, in order "to protect safety of others" [5].
- The mechanical subsystem should wear the card as little as possible to " comply with ethical design and sustainable development practices" [5] and reduce the frequency of replacement of cards. Suction cup is adopted since it will do little wear to the cards.

Reference

[1] "MQTT - the standard for IoT messaging." https://mqtt.org/

[2] Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[3] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A Unified

Embedding for Face Recognition and Clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[4] Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep Face Recognition. In British Machine Vision Conference (BMVC).

[5] IEEE, "IEEE Code of Ethics," ieee.org, Jun. 2020.
https://www.ieee.org/about/corporate/governance/p7-8.html

[6] National Institute of Standards and Technology, "Secure Hash Standard (SHS)," U.S. Department of Commerce, Aug. 2015. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf