SOFTWARE ENGINEERING PROJECT

SG Hassett – 2016117240
TM Mochaki – 2016212264
RG Phuthi – 2016392520

This document outlines the deliverables for the Requirements, Analysis and Design workflows of the Unified Process for the UFS COVID-19 Screening System. The UFS COVID-19 Screening System serves to automate the process of screening campus staff, student and visitors for COVID-19 symptoms before accessing the campus.

Table of Contents

Table of Figures

Chapter 1: Project Proposal

Project description

The current system for checking for COVID symptoms at a University of the Free State gate is manual and time-consuming. With Level 1 lockdown upcoming, a faster better system is needed. Design a new system that can quickly check a visitor's condition before entering campus. Important: assume that your development team will have to work under Level 1 Lockdown conditions and plan accordingly.

The system must be able to do the following:

- Determine a visitor's temperature
- Determine that a visitor has a mask
- Determine whether a visitor has come into contact with a COVID positive individual
- Determine the health of the visitor (cough, runny nose, and other COVID symptoms)
- Determine whether a visitor has a valid pass to come onto campus
- Determine a visitor's identification

The system must then generate a token that will allow a visitor to access the campus. The application must be easily usable by staff and students. Security will check whether a visitor's token is valid upon entering campus.

The approach you wish to utilise (smartphone, laptop, other devices) is completely up to your discretion but you will at least require a user interface, an application, and a database. You will design the complete system up until the implementation phase.

System description

Before arriving at any of the University of the Free State (UFS) campuses, staff, students, and visitors (from henceforth referred to as a user or users) will be required to complete a screening questionnaire. When a user accesses the screening questionnaire, the system will perform a permit check to ensure that the user has a valid permit. Only users with valid permits will be able to access the screening questionnaire. If a user is a student, the user will be required to provide a reason for their campus visit based on a selection in the screening questionnaire. Some

reasons will require an OTP. Upon arriving at one of the campus gates, a user will engage with a COVID-19 screening gate device. When the user scans their University of the Free State (UFS) access/identification card or a visitor scans their visitor barcode, the screening process will commence. During this process, the user is identified, and the system verifies that the user has completed and passed the UFS COVID-19 screening questionnaire. The system will thereafter check the user's temperature (thermal imaging) to ensure that it is within the normal range (35.1 °C - 37.8 °C), verify that the user has a mask on (image recognition), and activate an access token for the user which will be valid for only the day on which it was activated. In other words, it is valid until 23:59 on the day it was created. All tokens, questionnaire results, mask check results, and temperature readings will be reset/deactivated/cleared at 23:59 every day.

Components and features

System Database

A database will be created and used to store the following information needed to implement the campus gate security check:

- Staff and students' identification information
    - Full name(s) and surname
    - Staff/student number
    - National ID Number
    - Access card unique identifier
    - Permit status
- Visitors
    - Full name(s) and surname
    - Reference code
    - Visitor's barcode
- General security check data
    - Type of user (staff, student, and visitor)
    - Daily questionnaire completion status, validity, and start and end dates
    - Temperature reading

- Mask check results
- Access token

Security Notification Application

The Security Notification Application is a UFS COVID-19 Check System application that notifies security personnel at campus gates about problems at the gate screening apparatus such as inconclusive temperature measurements, high temperatures, abnormally high or low temperatures, failed mask-recognition results, and inconclusive mask-recognition results. The application is compatible with mobile smartphones. The application will run on UFS-owned mobile smartphones.

University of the Free State (UFS) Screening Questionnaire

Staff and students will be required to have completed a compulsory screening questionnaire indicating if they have been in contact with an individual who has been diagnosed with COVID-19. The questionnaire will require the user to provide information about their current health status as it pertains to COVID-19 symptoms. If the user successfully passes the questionnaire, this result will be entered into the user's database record, indicating that the user has completed and passed the questionnaire. The validity of the questionnaire result will be valid for only the day on which it was activated. In other words, it is valid until 23:59 on the day it was created. All screening questionnaire results will be reset at midnight. If a user is a student, the user will be required to provide a reason for their campus visit based on a selection provided in the screening questionnaire. Some reasons will require an OTP.

Visitors

The system will only make provision for visitors who have been granted temporary access permits by the University of the Free State. Each temporary permit has a unique reference code. The reference code (e.g. their National ID number) given on a visitor's access permit will be used to give the visitor access to the UFS Screening Questionnaire. In other words, this reference code will function as the

visitor's login credentials. The results of the screening questionnaire will be linked to this reference code. If the visitor successfully passes their health check questionnaire, a barcode will be generated and sent to their mobile device. This barcode will be used to update a visitor user's database record. This barcode will be valid for the duration of the visitor's access permit and will be linked to a visitor's unique reference code. This barcode will function in the same way as the barcode on student/staff access cards. A visitor will be prompted to scan this barcode at the gate ("Welcome. Please scan your access card or visitor's barcode"). When scanned at the gate, the barcode will initiate the same system and database triggers as is initiated by student/staff access cards.

Campus gate device

The device will comprise a steel display box attached to the end of a steel arm mounted on a steel column. The display box will contain the following:

An LCD monitor to provide a user interface e.g. instructions, messages, the visitor's identification information, screening results (temperature and mask recognition results), and screening process progress.

- An infrared camera to measure a user's temperature.
- A web camera for image recognition.
- A scanner for staff and student access cards and visitor barcodes.
- The display box on the security check structure will have a swivel mechanism to conduct a security check on both front and backseat passengers. The steel arm will have a mechanism to lower or raise the arm to accommodate cars of different heights.

The first phase (complete screening questionnaire, check for a valid permit, check for valid reason for campus visit if the user is a student) of the UFS COVID-19 Screening Process is initiated when the user accesses the UFS Screening Questionnaire. The second phase of the UFS COVID-19 Screening Process (temperature and mask checks, access token activation) will be initiated when a user scans their access card. The user will be prompted to do so using a welcome message ("Welcome. Please scan your access card/visitors barcode.") that will be displayed on the screen of the display box.

Complete UFS Screening Questionnaire

A user accesses the current UFS Screening Questionnaire and is prompted for their user type and login credentials.

- **Permit check**

  Once the user has provided their login credentials, the permit check process will be initiated. The system verifies that the user has a valid permit assigned to them. If the user has a valid permit, the user is permitted to complete the screening questionnaire

- **Reason for campus visit (student user)**

  A user is required to specify their user type (student, staff, visitor) when accessing the UFS Screening Questionnaire. If the user is a student, the user will be required to provide a reason for visiting the campus while completing the questionnaire. The user will be required to select a reason from a provided list of reasons. Some reasons will require the student to enter an OTP.

User identification and screening questionnaire completion verification

When the user swipes their card, the user will automatically be identified using the unique identifier embedded in staff and student access cards and visitor barcodes. The system will thereafter confirm that the user has completed the UFS screening questionnaire for that date and verify a negative (no symptoms) result.

Temperature check

Thermal imaging technology will be used to check visitors' check temperature. An infrared camera will be used to check the user's temperature.

Mask check

After a user's temperature has been measured and determined to be within the normal range (35.1 °C - 37.8 °C), the system will verify that the user has on. Image recognition technology will be used to verify that the user has a mask on. If the image recognition result is a fail (i.e. the system detects that the user is not wearing a mask), the user will have an option to retry. If the second attempt is a fail, a security guard will be notified through the Security Notification Application.

Access token

If the user has successfully passed all their checks, an entry will be made on their record in the database activating a token linked to their campus access/ID card/visitor barcode. The token will be valid for only the day on which it was activated. In other words, it is valid until 23:59 on the day it was created. All tokens will be reset at midnight. For a user to regain access to the campus if the user has left the campus during the token validity period, the user will simply have to go through the temperature and mask check processes.

Features (requirements) that are not achievable to implement.

Determining the current health of a visitor

Alternative

It will not be achievable to create a system feature that determines a visitor's current health at a campus gate by checking for symptoms of COVID-19 e.g. a cough, a runny nose, etc. While the temperature check feature will suffice as the first indicator of a possible infection, the University of the Free State's (UFS) current health check questionnaire will be used to determine if a visitor has any COVID-19 symptoms. The visitor will have to provide this information. This questionnaire must be completed by a visitor before coming to one of the UFS campuses. The result of the questionnaire will be sent to the security check database to update the visitor's record.

Additional features to be included as part of the requirements

How will non-staff/students with temporary permits (e.g. a guest lecturer) gain entry to the campus?

The system will only make provision for visitors who have been granted temporary access permits by the University of the Free State. Each temporary permit has a unique reference code. The reference code (e.g. their National ID number) given

on a visitor's access permit will be used to give the visitor access to the health check questionnaire. In other words, this reference code will function as the visitor's login credentials. The results of the health check questionnaire will be linked to this reference code. If the visitor successfully passes their health check questionnaire, a barcode will be generated and sent to their mobile device. The barcode will thereafter be updated each time a new health check questionnaire is completed. This barcode will be valid for the duration of the visitor's access permit and will be linked to a visitor's unique reference code. This barcode will function in the same way as the barcode on student/staff access cards. A visitor will be prompted to scan this barcode at the gate ("Welcome. Please scan your access card or visitor's barcode"). When scanned at the gate, the barcode will initiate the same system and database triggers as is initiated by student/staff access cards.

How would the system deal with a full vehicle (more than three passengers)?
Human intervention - if the device detects more than three people in a car, the system will set off an alert sound to alert the security personnel on duty at the gate to handle the situation as per UFS policy. The system will also display a message to the vehicle occupants to indicate that the vehicle occupants exceed the number of occupants allowed in a private vehicle as per UFS policy.
System description, components, and features

Chapter 2: Specification Document

University of the Free State (UFS) COVID-19 Screening System

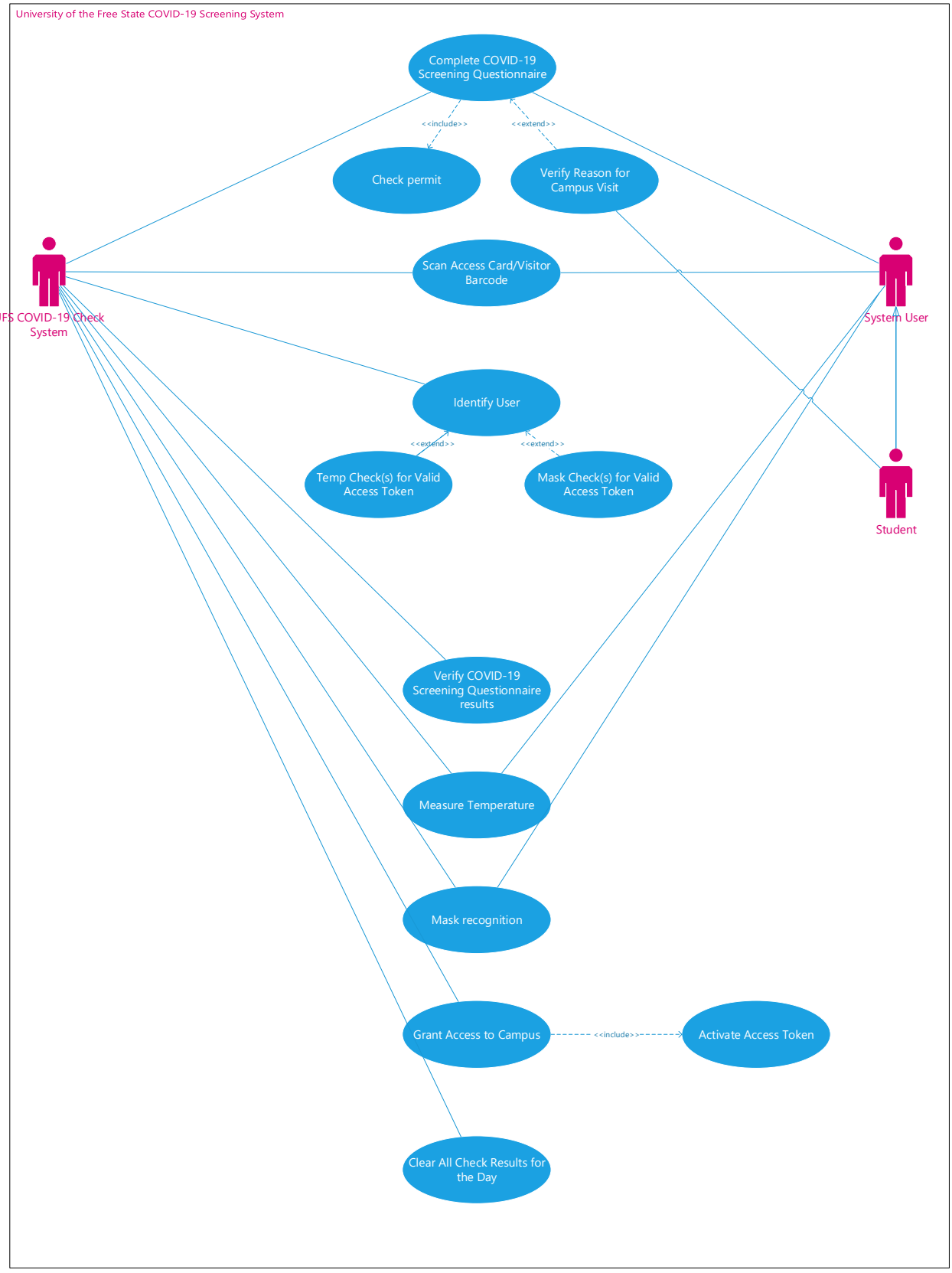


Figure 1 - University of the Free State (UFS) COVID-19 Screening System

Glossary

Access card: a UFS staff or student card

COVID-19 gate screening process (aka gate screening process): the portion of the UFS COVID-19 Check System that takes place at the campus gates. This includes identifying users, verifying questionnaire statuses and results, measuring temperatures, mask recognition, and granting access tokens.

COVID-19 screening gate apparatus: the device that performs provides input for the COVID-19 gate screening process. The device will be situated at each campus gate.

Display box: the part of the gate screening apparatus that contains the LED monitor for the user interface, a thermal camera (for measuring temperature), a digital camera (for mask recognition) and a scanner for access cards/visitor barcodes.

Security notification application: a UFS COVID-19 Check System application that notifies security personnel at campus gates about problems at the gate screening apparatus such as inconclusive temperature measurements and mask-recognition results. The application is compatible with mobile smartphones. The application will run on UFS-owned mobile smartphones.

System user (aka user): University of the Free State (UFS) staff, students, and visitors.

System: the overall system responsible for implementing each functionality of the UFS COVID-19 Check System, including updating the UFS COVID-19 Check System database and interfacing with the system users.

UFS COVID-19 Check System database record (aka database record): an individual system user's record in the UFS COVID-19 Check System database.

UFS COVID-19 Check System database: the database containing staff, students' and visitors' personal identification information, UFS COVID-19 screening questionnaire results, temperature readings, mask recognition results, and access tokens.

UFS COVID-19 screening questionnaire: a questionnaire used to screen system users for COVID-19 symptoms before the users arrive at the campus gate.

Unique identifier: an identifying mechanism linked to a system user's access card/visitor's barcode that is used to access the specific user's database record to for UFS COVID-19 Check System updates.

Visitor's barcode: an access barcode that is allocated to a visitor with a valid UFS campus permit. The barcode is generated and allocated the first time the visitor completes the UFS COVID-19 screening questionnaire. This barcode is only generated for visitors (e.g. guest lecturer, vendor) with valid campus permits and reference codes generated when a permit is issued.

Completed Before Going to Campus

Complete COVID-19 Screening Questionnaire

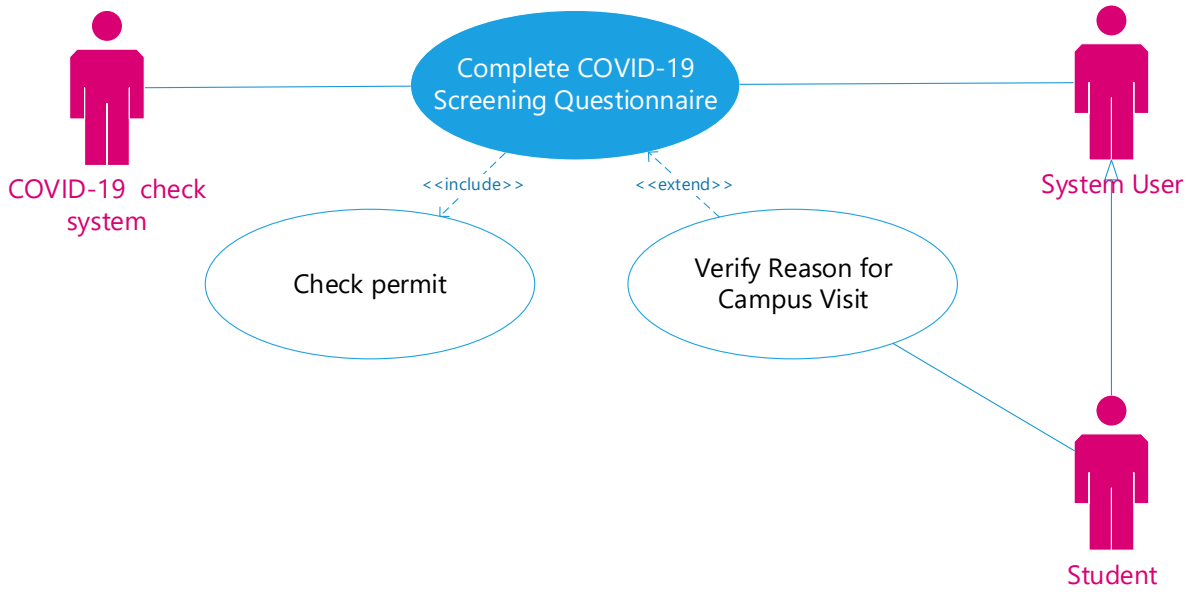


Figure 2 - Complete COVID-19 Screening Questionnaire Use Case

Pre-Condition:
The Permit Check use case is a pre-requisite as a user will not be able to access the questionnaire unless the system has verified that they have been issued a valid permit.

Brief Description:
The *Complete COVID-19 Screening Questionnaire* use case requires a system user to fill in the University of the Free State (UFS) COVID-19 Screening Questionnaire, that screens for COVID-19 symptoms, and processes the user's answers.

Step by Step Description:

1. The system allows the staff member/student/visitor access to the questionnaire.
2. The staff member/student/visitor fills in the questionnaire.
3. The system processes the questionnaire answers.
4. The system yields a result based on questionnaire answers.
5. The system updates the staff member/student/visitor's database record with the questionnaire result.
6. The questionnaire status is changed to "Completed".
7. The staff member/student/visitor is notified of the result of the questionnaire.
8. The staff member/student/visitor signs out of the system.

Scenarios:

- Best case scenario – User answers all questions on the questionnaire
    - The system allows João to access the questionnaire
    - João fills in the questionnaire.
    - The system processes João's questionnaire answers.
    - The system verifies that João's questionnaire answers comply with university policy on who is permitted on campus
    - The system updates João's database record with his questionnaire result.

    - João's questionnaire status is changed to "Completed".
    - The system notifies João that he is allowed on campus.
    - João signs out of the system.

- Worst case – Allowed on campus even after "failing" questionnaire
    - The system allows João to access the questionnaire
    - João fills in the questionnaire.
    - The system processes João's questionnaire answers.
    - The system verifies that João's questionnaire answers comply with university policy on who is permitted on campus
    - The system notifies João that is he NOT allowed on campus based on his answers.
    - The system updates João's database record with the questionnaire result.
    - João signs out of the system.
    - João signs into the system again.
    - The system allows João to complete the questionnaire again.
    - João fills in the questionnaire to get a satisfactory result.
    - The system updates João's database record with his questionnaire result.
    - João's questionnaire status is changed to "Completed".
    - The system notifies João that he is allowed on campus.
    - João signs out of the system.

- Alternative 1 – The questionnaire is not completed but user is able to continue
    - The system allows João to access the questionnaire
    - The system fails to load the questionnaire
    - João cannot proceed with the questionnaire.
    - João takes a chance and goes to the campus.
    - João arrives at the gate.
    - The system allows João to proceed with the COVID-19 gate screening process.

- Alternative 2 – The visitor is unable to continue with the COVID-19 gate screening process despite having completed the questionnaire.
    - The system allows João to access the questionnaire
    - João fills in the questionnaire.
    - The system processes João's questionnaire answers.
    - The system verifies that João's questionnaire answers comply with university policy on who is permitted on campus
    - The system fails to update João's database record with his questionnaire result.
    - João's questionnaire status is changed to "Completed".
    - The system notifies João that he is allowed on campus.
    - João signs out of the system.
    - João arrives at the campus gate.
    - The system does not allow João to continue.

- Alternative 3 – The questionnaire is half-completed
    - The system allows João to access the questionnaire

    - João fills in the questionnaire.
    - The system malfunctions.
    - Joao is unable to complete the questionnaire.
    - The system abruptly ends the questionnaire.
    - The system does not allow João to access the questionnaire again.

- Alternative 4 – An empty questionnaire is submitted
    - The system allows João to access the questionnaire
    - João reads through the screening questions before answering.
    - João accidentally submits an empty questionnaire.
    - The system tries to process the questionnaire.
    - The system is not able to process the empty questionnaire.

- Alternative 5 – A semi-completed questionnaire is submitted
    - The system allows João to access the questionnaire
    - João reads through the screening questions.
    - João answers some of the screening questions as he scans through the questionnaire.
    - João accidentally submits a semi-completed questionnaire.
    - The system tries to process the questionnaire.
    - The system is not able to process the empty questionnaire.

- Alternative 6 – A semi-completed questionnaire is submitted (second outcome)
    - The system allows João to access the questionnaire
    - João reads through the screening questions.
    - João answers some of the screening questions as he scans through the questionnaire.
    - João accidentally submits a semi-completed questionnaire.
    - The system tries to process the questionnaire.
    - The system misinterprets empty fields.
    - The system notifies João that is he NOT allowed on campus based on his answers.
    - The system updates João's database record with the questionnaire result.
    - João is not able to retake the questionnaire.

Check Permit

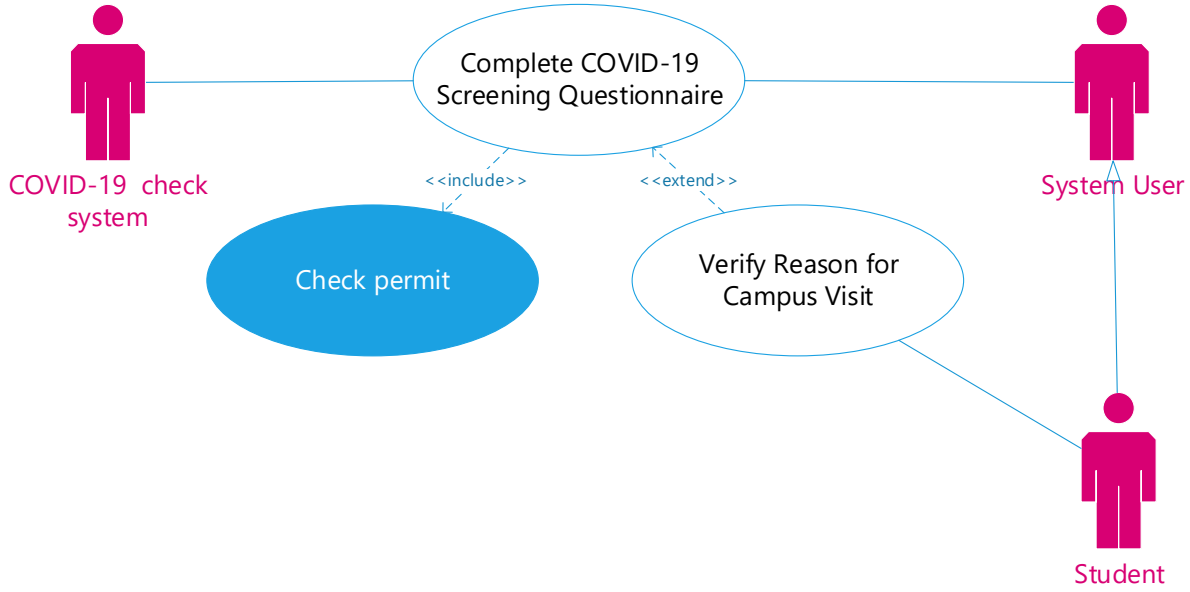


Figure 3 - Check Permit Use Case

Pre-condition
The *Complete COVID-19 Screening Questionnaire* use case must have been initiated.

Brief Description:
Before a user can access the COVID-19 Screening Questionnaire, the *Check Permit* use case determines whether the staff member/student/visitor has been issued a valid access permit.

Step-by-step:
1. Prompt the staff member/student/visitor for login information.
2. Determine if there is a valid permit issued to the staff member/student/visitor.
3. Link the valid permit to the staff member's/student's/visitor's unique identifier.
4. Grant the staff member/student/visitor access to the COVID-19 Screening Questionnaire.

Scenarios
- Best-case – the user has a valid permit that is linked to the database record and access card.
    - Sarah accesses the COVID-19 Screening questionnaire.
    - The system prompts Sarah for her user type (Staff member/student/visitor).
    - Sarah selects her user type.
    - The system prompts Sarah for her login info.
    - Sarah inputs her login credentials.
    - The system processes Sarah's info.
    - The system determines if Sarah has a valid permit issued to her.
    - The system updates Sarah's database record.
    - Sarah's permit is linked to her access card/barcode unique identifier.
    - The system allows her to continue to the COVID-19 Screening questionnaire.

- Worst-case – user repeatedly inputs invalid login credentials
    - Sarah accesses the COVID-19 Screening questionnaire.
    - The system prompts Sarah for her user type (Staff member/student/visitor).
    - Sarah selects her user type.
    - The system prompts Sarah for her login info.
    - Sarah inputs her login credentials in uppercase letters (CAPSLOCK).
    - The system processes the invalid credentials.
    - The system cannot identify Sarah in the database.
    - Sarah repeatedly inputs her credentials in uppercase letters.
    - The system crashes.
    - Sarah is not able to access the COVID-19 Screening Questionnaire login page.

- Alternative 1 – user enters incorrect login credentials
    - Sarah accesses the COVID-19 Screening questionnaire.
    - The system prompts Sarah for her user type (Staff member/student/visitor).
    - Sarah selects her user type.
    - The system prompts Sarah for her login info.
    - Sarah inputs the incorrect info into the system.
    - The system processes Sarah's info.
    - The system cannot identify Sarah in the database.
    - The system cannot check for a permit.
    - Sarah is denied access to the COVID-19 Screening Questionnaire.

- Alternative 2 – the system does not update the user's database record
    - Sarah accesses the COVID-19 Screening questionnaire.
    - The system prompts Sarah for her user type (Staff member/student/visitor).
    - Sarah selects her user type.
    - The system prompts Sarah for her login info.
    - Sarah inputs her login credentials.
    - The system processes Sarah's info.
    - The system determines if she has a valid permit issued to her.
    - The system updates Sarah's database record.
    - The system update fails.
    - Sarah's permit status is not linked to her access card/barcode.
    - Sarah denied access to the COVID-19 Screening Questionnaire.

- Alternative 3 – system updates the incorrect database record and unique identifier
    - Sarah accesses the COVID-19 Screening questionnaire.
    - The system prompts Sarah for her user type (Staff member/student/visitor).
    - Sarah selects her user type.
    - The system prompts Sarah for her login info.
    - Sarah inputs her login credentials.
    - The system processes Sarah's info.
    - The system determines if she has a valid permit issued to her.
    - The system then updates the incorrect database record.
    - Sarah's permit is linked to the incorrect access card/barcode.

    - The system allows her to continue to the COVID-19 Screening questionnaire.
    - Sarah arrives at the campus gate.
    - Sarah scans her access card/visitor barcode.
    - The system has no record of a permit linked to Sarah's access card/visitor's barcode.
    - Sarah is not able to continue with the gate screening process.

- Alternative 4 – the system does not link updated permit status to the user's unique identifier
    - Sarah accesses the COVID-19 Screening questionnaire.
    - The system prompts Sarah for her user type (Staff member/student/visitor)
    - The system asks Sarah for her login info.
    - Sarah inputs her login credentials.
    - The system processes Sarah's info.
    - The system determines if she has a valid permit issued to her.
    - The system then updates Sarah's database record.
    - Sarah's permit status is not linked to her access card/barcode.
    - The system allows her to continue to the COVID-19 Screening questionnaire.
    - Sarah arrives at the campus gate.
    - Sarah scans her access card/visitor barcode.
    - The system has no record of a permit linked to Sarah's access card/visitor's barcode.
    - Sarah is not able to continue with the gate screening process.

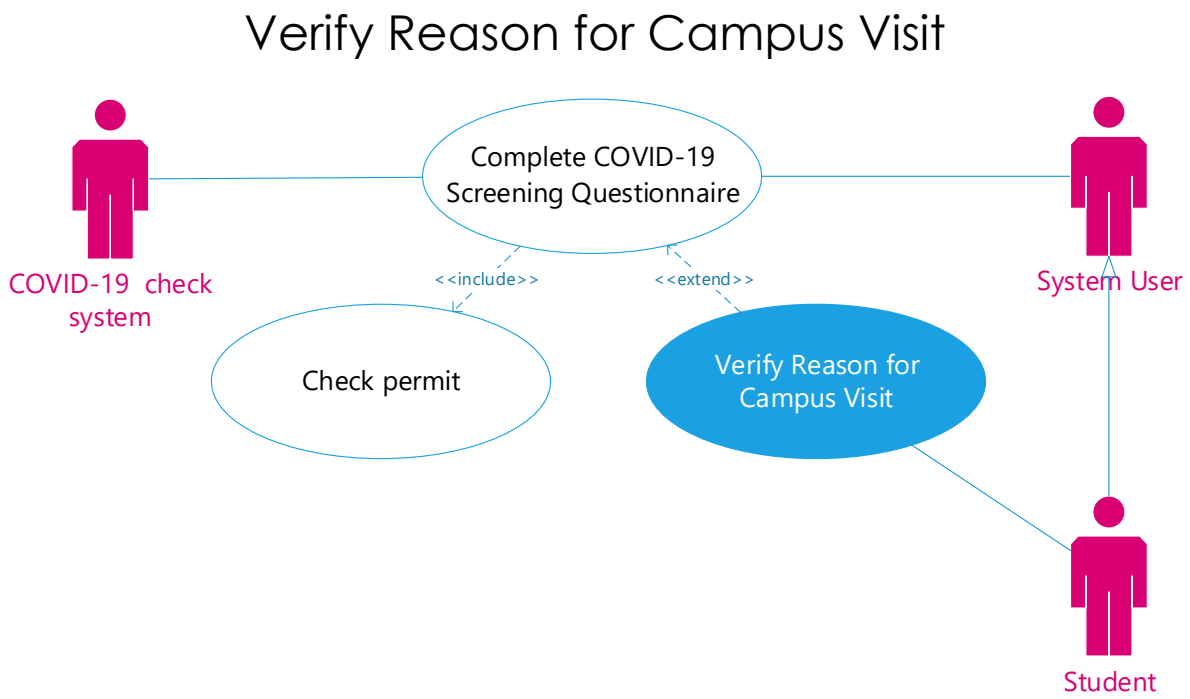


Figure 4 – Verify Reason for Campus Visit Use Case

Pre-condition

The *Complete COVID-19 Screening Questionnaire* and *Check Permit* use cases are a prerequisite as a user who has been identified as a student must specify a reason for their campus visit while completing the screening questionnaire.

Brief Description

The *Verify Reason for Campus Visit* use case requires students to specify a reason for their campus visit and provide verification thereof. A student will not be able to complete the screening questionnaire if a valid reason is not provided and verified.

Step-by-step Description

1. For each user identified as a student, prompt the student to select a reason for their campus visit from a list.
    - 1.1. Attending a class
    - 1.2. Meeting with a lecturer
    - 1.3. I live on campus
    - 1.4. Lab Access
2. Process student's selection
    - 2.1. If 'Attending a Class', access the UFS student class timetable database to verify.
    - 2.2. If 'Meeting with a lecturer', prompt student to enter OTP generated by the lecturer.
    - 2.3. If 'I live on campus', prompt the student to enter OTP generated by residence RA.
    - 2.4. If 'Lab Access', prompt student to enter OTP generated after booking lab slot.
3. Process and verify
    - 3.1. If the user is attending class on that day, verify from the UFS class timetable database, OR
    - 3.2. Process and verify OTP provided

4. Terminate session if the student does not have a class or inputs an invalid OTP, otherwise continue with COVID-19 Screening Questionnaire

Scenarios

- Best-case scenario – the user is coming to campus for a class/valid reason
    - Sarah accesses the screening questionnaire.
    - The system prompts Sarah to select her user type (Staff member/student/visitor).
    - Sarah selects "Student".
    - The system prompts Sarah for her login credentials.
    - Sarah logins in.
    - Sarah passes her permit check.
    - The system allows Sarah to complete the screening questionnaire.
    - Sarah completes the questionnaire.
    - The system prompts Sarah for the reason for her campus visit.
    - Sarah selects 'Attending a class'.
    - The system accesses the UFS student class timetable database.
    - The system verifies that Sarah has a class.
    - Sarah submits the questionnaire.

    OR

    - The system prompts Sarah for the reason for her campus visit.
    - Sarah selects Meeting with a Lecturer/I live on campus/Lab Access
    - The system prompts Sarah for an OTP.
    - Sarah provides her OTP.
    - The system verifies Sarah's OTP.
    - Sarah submits the questionnaire.

- Worst-case scenario – the user is unaware of providing a reason for the campus visit
    - Sarah accesses the screening questionnaire.
    - The system prompts Sarah to select her user type (Staff member/student/visitor).
    - Sarah selects "Student".
    - The system prompts Sarah for her login credentials.
    - Sarah logins in.
    - Sarah passes her permit check.
    - The system allows Sarah to complete the screening questionnaire.
    - Sarah completes the questionnaire.
    - Sarah does not know that she needs to provide a reason for her campus visit.
    - Sarah submits the questionnaire.
    - The system cannot process the semi-completed questionnaire

- Alternative 1 – The system cannot access the timetable database
    - Sarah accesses the screening questionnaire.
    - The system prompts Sarah to select her user type (Staff member/student/visitor).
    - Sarah selects "Student".
    - The system prompts Sarah for her login credentials.
    - Sarah logins in.
    - Sarah passes her permit check.
    - The system allows Sarah to complete the screening questionnaire.

- Sarah completes the questionnaire.
- The system prompts Sarah for the reason for her campus visit.
- Sarah selects 'Attending a class'.
- The system tries to access the UFS student class timetable database.
- The system cannot access the timetable database (database failure).
- The system does not allow Sarah to submit her questionnaire.

- Alternative 2 – miscommunication between the UFS COVID-19 Check System and the timetable database
  - Sarah accesses the screening questionnaire.
  - The system prompts Sarah to select her user type (Staff member/student/visitor).
  - Sarah selects "Student".
  - The system prompts Sarah for her login credentials.
  - Sarah logins in.
  - Sarah passes her permit check.
  - The system allows Sarah to complete the screening questionnaire.
  - Sarah completes the questionnaire.
  - The system prompts Sarah for the reason for her campus visit.
  - Sarah selects 'Attending a class'.
  - The system accesses the UFS student class timetable database.
  - The system waits for a response from the timetable database.
  - The system does not receive a response from the timetable database.
  - The system waits for a response indefinitely.

- Alternative 3 – Incorrect timetable accessed
  - Sarah accesses the screening questionnaire.
  - The system prompts Sarah to select her user type (Staff member/student/visitor).
  - Sarah selects "Student".
  - The system prompts Sarah for her login credentials.
  - Sarah logins in.
  - Sarah passes her permit check.
  - The system allows Sarah to complete the screening questionnaire.
  - Sarah completes the questionnaire.
  - The system prompts Sarah for the reason for her campus visit.
  - Sarah selects 'Attending a class'.
  - The system accesses the incorrect student's timetable.
  - The system verifies that the student does not have a class.
  - The system does not allow Sarah to submit her questionnaire.

- Alternative 4 – OTP mix-up
  - Sarah accesses the screening questionnaire.
  - The system prompts Sarah to select her user type (Staff member/student/visitor).
  - Sarah selects "Student".
  - The system prompts Sarah for her login credentials.
  - Sarah logins in.
  - Sarah passes her permit check.
  - The system allows Sarah to complete the screening questionnaire.
  - Sarah completes the questionnaire.

- The system prompts Sarah for the reason for her campus visit.
- Sarah has both a meeting with a lecturer and is going to the UFS labs.
- Sarah selects "Lab Access".
- The system prompts Sarah for an OTP.
- Sarah provides the OTP generated for her meeting with her lecturer.
- The system cannot match the OTP to a lab booking.
- The system prompts Sarah for an OTP.
- Sarah enters the same OTP.

- Alternative 5 – Invalid OTP (previously-used OTP)
  - Sarah accesses the screening questionnaire.
  - The system prompts Sarah to select her user type (Staff member/student/visitor).
  - Sarah selects "Student".
  - The system prompts Sarah for her login credentials.
  - Sarah logins in.
  - Sarah passes her permit check.
  - The system allows Sarah to complete the screening questionnaire.
  - Sarah completes the questionnaire.
  - The system prompts Sarah for the reason for her campus visit.
  - Sarah does not have a reason for her campus visit.
  - Sarah selects "Lab Access".
  - The system prompts Sarah for an OTP.
  - Sarah tries to use her previous OTP.
  - The system cannot match the OTP to a lab booking.

Completed at Campus Gates

Scan Access Card/Visitor Barcode

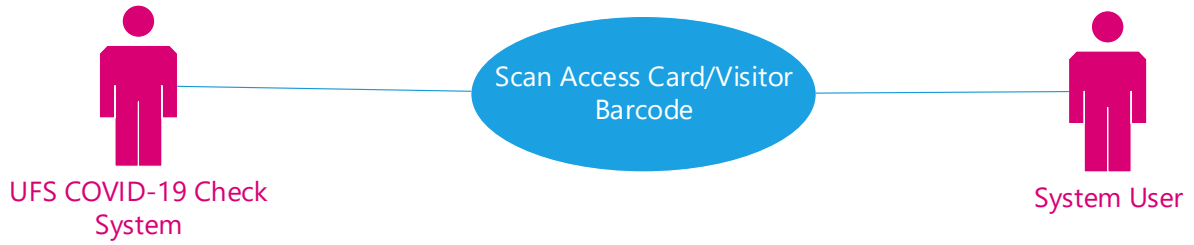


Figure 5 – Scan Access Card/Visitor Barcode Use Case

Pre-condition

None. The *Scan Access Card/Visitor Barcode* use case initiates the UFS COVID-19 Check System at the campus gates.

Brief Description

The *Scan Access Card/Visitor Barcode* use case scans a staff member's/student's access card or visitor's (e.g. Guest Lecturer) barcode at the campus gates and stores and sends the staff member's/student's/visitor's unique identifier to the *Identify User* use case.

Step by Step Description

1. Staff member/student/visitor approaches the scanner.
    a. Adjust (raise/lower) apparatus arm to car height, if necessary.
2. The system prompts staff member/student/visitor to scan their access card/visitor's barcode
3. Staff member/student/visitor scans their access card or provided barcode.
4. The system captures the staff member's/student's/visitor's unique identifier.

Scenarios:

- Best case scenario – User scans successfully with no problems.
    - Aurélio arrives at the gate and approaches the COVID-19 gate screening apparatus.
    - The system prompts Aurélio to scan his access card/provided barcode.
    - Aurélio scans his card/barcode.
    - The system captures Aurélio's unique identifier.
    - The system passes Aurélio's unique identifier to the next use case to identify Aurélio.

- Worst case scenario – User does not scan their access card.
    - Aurélio arrives at the gate and approaches the COVID-19 gate screening apparatus.
    - The system prompts Aurélio to scan his access card.
    - Aurélio fails to read or, comply with, the system prompt to scan his card.
    - Aurélio does not scan the card/barcode.
    - The system is unable to continue with the COVID-19 gate screening.

- Alternative 1 – the user might try to scan too fast.
    - Aurélio arrives at the gate and approaches the UFS COVID-19 check apparatus.

    - The system prompts Aurélio to scan his access card/provided barcode
    - Aurélio quickly flashes his access card/provided barcode.
    - The system is unable to scan the card/barcode.
    - The system is unable to capture the unique identifier.

- Alternative 2 – The user tries to scan too far from the COVID-19 gate screening apparatus.
    - Aurélio arrives at the gate and approaches the COVID-19 gate screening apparatus.
    - The system prompts Aurélio to scan his access card/provided barcode
    - Aurélio scans his card at a considerable distance (1m) from the scanner.
    - The system is unable to scan the card.
    - The system is unable to capture the unique identifier.

- Alternative 3 – Access card/Barcode might be unreadable.
    - Aurélio arrives at the gate and approaches the COVID-19 gate screening apparatus.
    - The system prompts Aurélio to scan his access card/provided barcode
    - Aurélio scans his card/barcode.
    - The access card is damaged and is not readable.
    - The system is unable to capture the unique identifier.
    - The system continues to prompt Aurélio to scan his card.

- Alternative 4 – User is unable to continue with the COVID-19 gate screening after a successful scan
    - Aurélio arrives at the gate and approaches the COVID-19 gate screening apparatus.
    - The system prompts Aurélio to scan his access card/provided barcode
    - Aurélio scans his card/barcode.
    - The system scans the card/barcode.
    - The system captures the unique identifier.
    - The system fails to pass on the identifier to the next use case to identify Aurélio.

- Alternative 5 – User scans card through a closed car window
    - Aurélio arrives at the gate and approaches the COVID-19 gate screening apparatus.
    - The system prompts Aurélio to scan his access card/provided barcode.
    - Aurélio has his car window up.
    - Aurélio scans his card/barcode.
    - The system cannot scan the card through the car window.

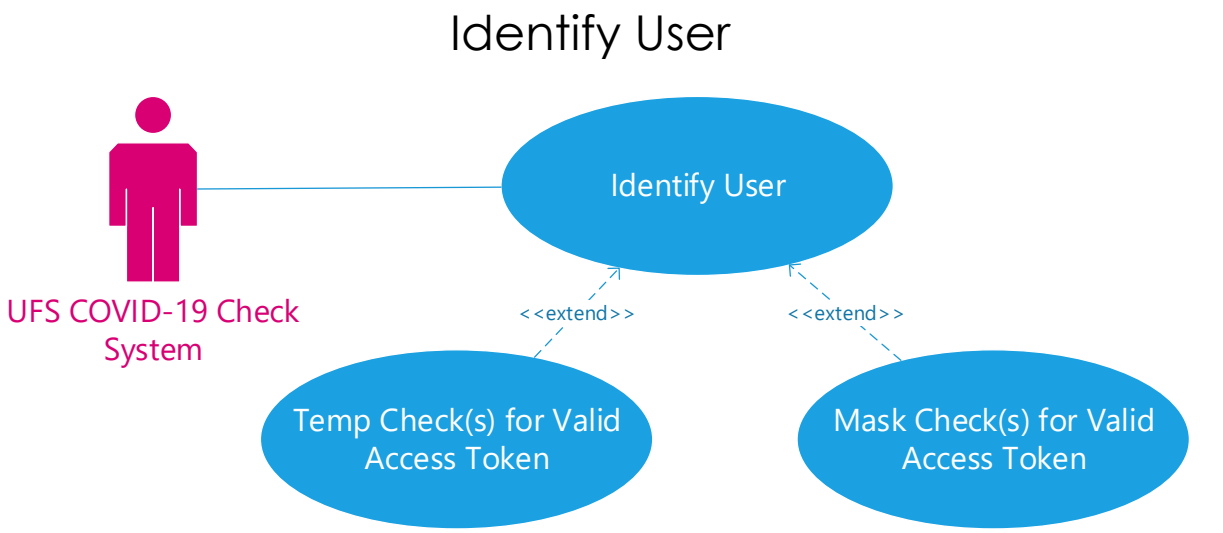


Figure 6 - Identify User Use Case

Pre-Condition:
The Scan Access Card/Visitor Barcode use case is a pre-requisite because the system will not be able to identify a user if it does not receive a unique identifier.

Brief Description:
The *Identify User* use case enables the UFS COVID-19 Check System to identify a user and access their UFS COVID-19 Check System database record using the unique identifier received from the *Scan Access Card/Visitor Barcode* use case. When the system identifies the user and detects that a user already has a valid token (has already completed the entire COVID-19 gate screening process that day), the system will only check the user's temperature [*Temp Check(s) for Valid Access Token i.e. Measure Temperature]* and verify that the user is wearing a mask [*Mask Check(s) for Valid Access Token i.e. Mask recognition]* before granting access.

Step by Step Description:

1. Receive the user's unique identifier from *Scan Access Card/Barcode* use case.
2. Use the unique identifier to find the user's database record in the UFS COVID-19 Check System database.
3. Access the user's UFS COVID-19 Check System database record
4. Check if the user has an existing, valid token for the day.
5. Skip to temperature and mask checking if the user already has a valid access token.
6. Continue with the original check system if the user does not have a valid access token.

Scenarios:

- Best case scenario – the user is successfully identified in the UFS COVID-19 Check System database.
    - Concessão's unique identifier is captured by the system.
    - The system uses the captured unique identifier to identify Concessão.
    - The system successfully matches Concessão's unique identifier to Concessão's database record.

- Worst case – user overloads system with multiple scans
    - Concessão scans her access card more than once.
    - The system captures multiple unique identifiers.
    - The system cannot process multiple unique identifiers and overloads.
    - The system crashes due to a system overload.

- Alternative 1 – user already has a valid access token, but must repeat the COVID-19 gate screening process
    - Concessão's unique identifier is captured by the system.
    - The system uses the captured unique identifier to identify Concessão in the UFS COVID-19 Check System database.
    - The system accesses Concessão's database record.
    - The system determines that Concessão already has an access token.
    - The system fails to proceed to the temperature and mask checks.
    - Concessão must repeat the entire COVID-19 gate screening process.

- Alternative 2 – user can skip the COVID-19 gate screening process without a valid access token
    - Concessão's unique identifier is captured by the system.
    - The system uses the captured unique identifier to identify Concessão in the UFS COVID-19 Check System database.
    - The system accesses Concessão's database record.
    - Concessão does not have an access token.
    - The system skips to the temperature and mask checks.
    - The system grants Concessão access to the campus.
    - Concessão does not complete the entire COVID-19 gate screening process.

- Alternative 3 – the system cannot identify the user and allows the user to continue with the COVID-19 gate screening process
    - Concessão's unique identifier is captured by the system.
    - The system uses the identifier to identify Concessão in the UFS COVID-19 Check System database.
    - The system fails to identify Concessão in the UFS COVID-19 Check System database.
    - The system continues with the COVID-19 gate screening process without identifying Concessão.

- Alternative 4 – system encounters multiple records with the same unique identifier
    - Concessão's unique identifier is captured by the system.
    - The system uses the captured unique identifier to identify Concessão in the UFS COVID-19 Check System database.
    - The system encounters multiple records on the database with the same unique identifier.
    - The system cannot identify Concessão.
    - The system decides to nullify the query.

- Alternative 5 – the user gains access to the campus without completing the COVID-19 gate screening due to incorrect record access
    - Concessão's unique identifier is captured by the system.
    - The system uses the captured unique identifier to identify Concessão in the UFS COVID-19 Check System database.
    - The system identifies/accesses the incorrect database record.
    - The system determines that the selected database record, which is not Concessão's, already has a valid access token.
    - The system allows Concessão to skip to the temperature and mask checks.
    - Concessão does not complete the COVID-19 gate screening process.

Verify COVID-19 Screening Questionnaire Results

Figure 7 - Verify COVID-19 Screening Questionnaire Results Use Case

Pre-condition
The *Complete COVID-19 Screening Questionnaire* and *Identify User* use cases are pre-requisites as the user must have been identified to access the user's database record for the results received from the *Complete COVID-19 Screening Questionnaire* use case.

Brief Description
The *Verify COVID-19 Screening Questionnaire Results* use case verifies that the identified user has completed the UFS COVID-19 screening questionnaire and accesses the results thereof to verify that the user has no COVID-19 symptoms.

Step-by-step Description
1. The system has identified the staff member/student/visitor and accessed their UFS COVID-19 Check System database record.
2. Verify that the staff member/student/visitor has completed the COVID-19 screening questionnaire.
3. Verify that the results of the questionnaire indicate that the staff member/student/visitor has no COVID-19 symptoms.

Scenarios
- Best-case scenario – the user is identified, and the system verifies questionnaire results
    - The system has identified João.
    - The system has accessed João's database record.
    - The system verifies that João has completed the questionnaire.
    - The system verifies that the questionnaire result indicates that João has no COVID-19 symptoms.

- Worst-case scenario – the user is identified, and the system identifies the incorrect questionnaire results
    - The system has identified João.
    - The system has accessed the incorrect database record.
    - The system verifies that "João" has completed the questionnaire.
    - The system determines that the questionnaire results indicate that "João" has COVID-19 symptoms.
    - João is not allowed to continue with the COVID-19 gate screening process.
    - The system terminates the session.

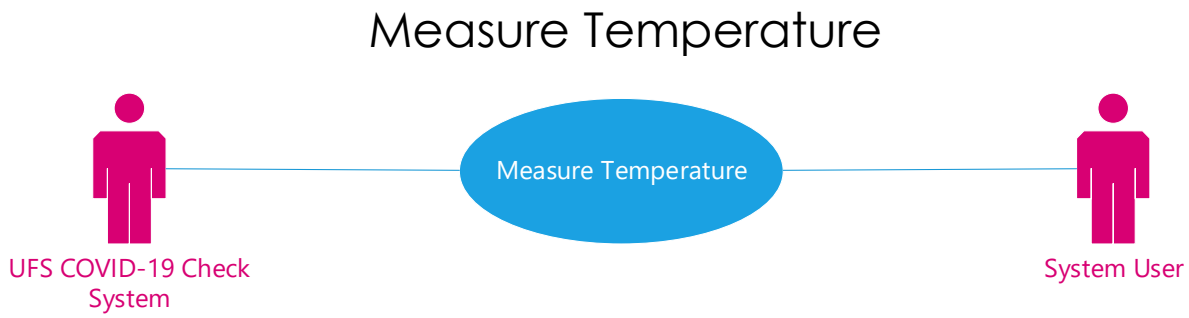


Figure 8 – Measure Temperature Use Case

Pre-condition
The *Identify User* use case is a pre-requisite use case because the system must update the user's database record with the user's most current temperature reading.

Brief Description:
The *Measure Temperature* use case measures a student's/staff member's/visitor's temperature to determine if it falls within a normal range (35.1 °C - 37.8 °C). Temperature measuring and mask recognition occur simultaneously.

Step-by-step Description
1. Switch on the green camera light in the display box.
2. Prompt staff member/student/visitor to look at the light.
    a. Align user's face in camera frame by swivelling display box, if necessary.
3. Measure the staff member's/student's/visitor's core temperature with the infrared/thermal camera.
4. Verify that the temperature reading is within the normal range (35.1 °C - 37.8 °C).
5. Update staff member's/student's/visitor's database record.
6. Display temperature reading to staff member/student/visitor.

Scenarios:
- Best-case scenario – the system accurately measures the user's temperature
    - The system switches on the green camera light.
    - The system prompts Sarah to look at the green light.
    - Sarah faces the display box and looks at the green light.
    - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
    - The system measures Sarah's temperature.
    - The system verifies that Sarah's temperature is within the normal range (35.1 °C - 37.8 °C).
    - The system updates Sarah's database record.
    - The system displays Sarah's temperature on the display box monitor.

- Worst-case scenario – inaccurate measurement due to external elements and thermal imaging limitations
    - The system switches on the green camera light.
    - The system prompts Sarah to look at the green light.
    - Sarah has her car window up (windows reflect heat).
    - Sarah faces the display box and looks at the green light.

    - The system checks if Sarah's face is close enough to the camera.
    - The system cannot distinguish between Sarah's temperature and the heat reflecting off the window.
    - The system cannot get an accurate reading.

- Alternative 1 – inconclusive measurement due to external elements and thermal imaging limitations
    - The system switches on the green camera light.
    - The system prompts Sarah to look at the green light.
    - Sarah has her car window up (windows reflect heat).
    - Sarah faces the display box and looks at the green light.
    - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
    - The system cannot distinguish between Sarah's temperature and the heat reflecting off the window.
    - The system cannot get an accurate reading.
    - The system yields an inconclusive reading.
    - The system alerts a security guard (Thando) via the system notification application.
    - Thando manually measures Sarah's temperature with a thermometer.
    - Sarah's temperature is within the normal range
    - Thando inserts Sarah's temperature via the system notification application.
    - The system updates Sarah's database record.

- Alternative 2 – the user can proceed with the gate screening process despite having a high temperature
    - The system switches on the green camera light.
    - The system prompts Sarah to look at the green light.
    - Sarah faces the display box and looks at the green light.
    - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
    - The system measures Sarah's temperature.
    - Sarah's temperature is over 37.8 C°.
    - Sarah's database record is updated.
    - The system displays Sarah's temperature on the display box monitor.
    - The system allows Sarah to continue with the COVID-19 gate screening process.

- Alternative 3 – security personnel are notified about a user with high temperature
    - The system switches on the green camera light.
    - The system prompts Sarah to look at the green light.
    - Sarah faces the display box and looks at the green light.
    - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
    - The system measures Sarah's temperature.
    - Sarah's temperature is over 37.8 C°.
    - Sarah's database record is updated.
    - The system notifies a security guard via the system notification application.

- The system displays Sarah's temperature.
- The system prompt informs Sarah that she is not allowed on campus due to a high temperature.
- The COVID-19 gate screening process is then terminated.

- Alternative 4 – system fails to take a temperature reading
  - The system switches on the green camera light.
  - The system prompts Sarah to look at the green light.
  - Sarah faces the display box and looks at the green light.
  - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
  - The system prompts Sarah to move closer to the display box.
  - The system checks if Sarah's face is close enough to the camera.
  - The system prompts Sarah to come closer to the camera.
  - Sarah is already as close as possible.
  - The system fails to measure Sarah's temperature.
  - The system alerts a security guard (Thando) via the system notification application.
  - Thando manually measures Sarah's temperature with a thermometer.
  - Sarah's temperature is within the normal range
  - Thando inserts Sarah's temperature via the system notification application.
  - The system updates Sarah's database record.

- Alternative 5 – system fails to update database record
  - The system switches on the green camera light.
  - The system prompts Sarah to look at the green light.
  - Sarah faces the display box and looks at the green light.
  - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
  - The system measures Sarah's temperature.
  - The system verifies that Sarah's temperature is within the normal range (35.1 °C - 37.8 °C).
  - The system fails to update Sarah's database record.
  - The system displays Sarah's temperature on the display box monitor.
  - The system allows Sarah to continue with the COVID-19 gate screening process.

- Alternative 6 – the system takes an abnormally high temperature reading
  - The system switches on the green camera light.
  - The system prompts Sarah to look at the green light.
  - Sarah faces the display box and looks at the green light.
  - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
  - The system measures Sarah's temperature.
  - The system measures an abnormally high temperature (>40 °C).
  - The system alerts a security guard (Thando) via the system notification application.
  - Thando manually measures Sarah's temperature with a thermometer.

- Sarah's temperature is within the normal range
- Thando inserts Sarah's temperature via the system notification application.
- The system prompts Thando to verify that Sarah's temperature is within the normal range.
- Thando verifies Sarah's temperature.
- The system alerts system development team and relevant UFS personnel of system error.

- Alternative 7 – the system takes an abnormally high temperature reading
  - The system switches on the green camera light.
  - The system prompts Sarah to look at the green light.
  - Sarah faces the display box and looks at the green light.
  - The system checks if Sarah's face is close enough to the camera and swivels display box, if necessary.
  - The system measures Sarah's temperature.
  - The system measures an abnormally low temperature (<33 °C).
  - The system alerts a security guard (Thando) via the system notification application.
  - Thando manually measures Sarah's temperature with a thermometer.
  - Sarah's temperature is within the normal range
  - Thando inserts Sarah's temperature via the system notification application.
  - The system prompts Thando to verify that Sarah's temperature is within the normal range.
  - Thando verifies Sarah's temperature.
  - The system alerts system development team and relevant UFS personnel of system error.

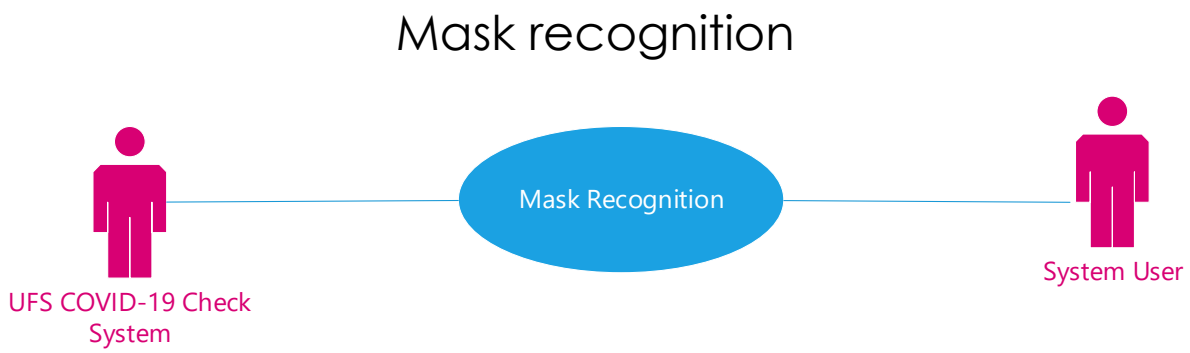


Figure 9 - Mask Recognition Use Case

Pre-condition
The *Identify User* use case is a pre-requisite because the staff member/student/visitor must have been identified to allow the system to update the correct database record.

Brief Description
The *Mask Recognition* use case enables the system to verify that a staff member/student/visitor is wearing a mask. Mask recognition and temperature measuring occur simultaneously.

Step-by-step Description
1. Switch on the green camera light in the display box.
2. Prompt staff member/student/visitor to look at the light.
3. Align staff member/student/visitor face in the camera frame.
    a. Swivel display box, if necessary
4. Prompt the user to remain still until the light goes off.
5. Image recognition software analyses camera feed.
6. Image recognition software verifies if staff member/student/visitor is wearing a mask.
7. Staff member's/student's/visitor's database record is updated with the mask check result.
8. Switch off the green camera light.
9. Indicate mask check result on screen in the display box.

Scenarios
- Best-case scenario – the user follows prompts correctly and mask recognition is completed successfully
    - The system switches on the green camera light.
    - The system prompts Joelle to look at the green light.
    - Joelle faces the display box to look at the green light.
    - The system successfully detects Joelle's face.
    - The system prompts Joelle to remain still until the green light goes off.
    - The image recognition software analyses the camera feed.
    - The image recognition software verifies that Joelle is wearing a mask.
    - The system updates Joelle's database record with the mask check result.
    - The system switches off the green camera light.
    - The system displays Joelle's mask check result on the display box screen.

- Worst-case scenario – the user does not follow prompts correctly

- The system switches on the green camera light.
- The system prompts Joelle to look at the green light.
- Joelle looks at the green light but does not face the camera.
- The system detects Joelle's facial profile.
- The image recognition software analyses the camera feed.
- The image recognition software yields an inconclusive mask check result.
- The system transmits the camera feed and a notification to a security guard (Thando) via the system notification application.
- Thando checks the camera feed via the system notification application on his mobile device.
- Thando verifies that Joelle is wearing a mask via the system notification application.
- The system updates Joelle's database record with the mask check result.
- The system switches off the green camera light.
- The system displays Joelle's mask check result on the display box screen.

- Alternative 1 – the system does not detect that the camera light is not working
  - The system switches on the green camera light.
  - The green camera light does not go on.
  - The system prompts Joelle to look at the green light.
  - Joelle does not see a green light.
  - The system indefinitely continues to prompt Joelle to look at the green light.

- Alternative 2 – the system determines that the user is not wearing a mask
  - The system switches on the green camera light.
  - The system prompts Joelle to look at the green light.
  - The system detects Joelle's face.
  - The image recognition software analyses the camera feed.
  - The image recognition software determines that Joelle is not wearing a mask.
  - The system displays a message prompting Joelle to put on a mask.
  - Joelle puts on a mask.
  - The system repeats the mask check.
  - The system verifies that Joelle is wearing a mask.
  - The system updates Joelle's database record with the mask check result.
  - The system switches off the green camera light.
  - The system displays Joelle's mask check result on the display box screen.

- Alternative 3 – inconclusive mask recognition result
  - The system switches on the green camera light.
  - The system prompts Joelle to look at the green light.
  - The system successfully detects Joelle's face.
  - The image recognition software analyses the camera feed.
  - The image recognition software yields an inconclusive mask check result.
  - The system transmits the camera feed and a notification is sent to security personnel (Thando) via the system notification application.
  - Thando checks the camera feed on the system notification application on his mobile device.

    - Thando indicates that Joelle is not wearing a mask.
    - The system displays a message prompting Joelle to put on a mask.
    - Joelle puts on a mask.
    - The system repeats the mask check.
    - The system prompts Thando to verify if Joelle is wearing a mask.
    - Thando verifies that Joelle is wearing a mask.
    - The system updates Joelle's database record with the mask check result.
    - The system switches off the green camera light.
    - The system displays Joelle's mask check result on the display box monitor.

- Alternative 4 – the system yields a false negative after the user has put their mask on.
    - The system switches on the green camera light.
    - The system prompts Joelle to look at the green light.
    - The system detects Joelle's face.
    - The image recognition software analyses the camera feed.
    - The image recognition software determines that Joelle is not wearing a mask.
    - The system prompts Joelle to put on a mask.
    - Joelle puts on a mask.
    - The system repeats the mask check.
    - The system detects that Joelle is not wearing a mask (false negative).
    - The system notifies security personnel (Thando) via the system notification application about Joelle's two failed mask checks.

- Alternative 5 – the system inaccurately determines that the user is not wearing a mask (false negative) and terminates session
    - The system switches on the green camera light.
    - The system prompts Joelle to look at the green light.
    - The system detects Joelle's face.
    - The image recognition software analyses the camera feed.
    - The image recognition software determines that Joelle is not wearing a mask.
    - The system prompts Joelle to put on a mask.
    - Joelle puts on a mask.
    - The system repeats the mask check.
    - The system detects that Joelle is not wearing a mask (false negative).
    - The system does not notify security personnel.
    - The system terminates the session.

- Alternative 6 – the system sends a notification to all security personnel via the system notification application
    - The system switches on the green camera light.
    - The system prompts Joelle to look at the green light.
    - The system detects Joelle's face.
    - The image recognition software analyses the camera feed.
    - The image recognition software yields an inconclusive mask check result.

- The system transmits the camera feed and a notification to all security personnel (Thando, Anna, and JP) at the campus gate via the system notification application.
- Thando, Anna, and JP all reply to the notification.
- The system is overloaded.
- The system terminates the session.

Grant Access to Campus (Access Token)

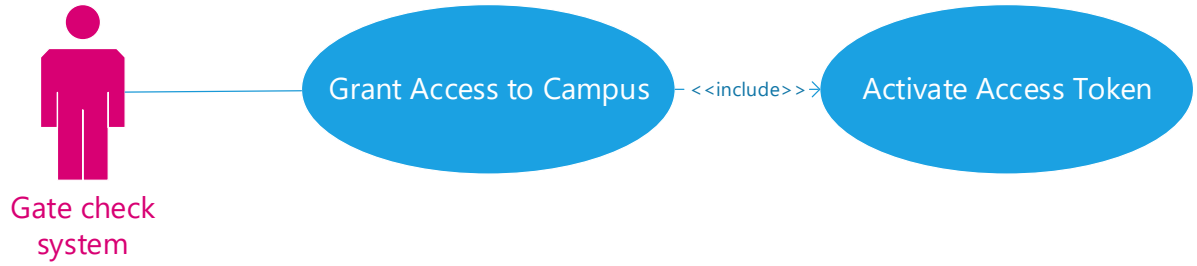


Figure 10 – Grant Access to Campus Use Case

Pre-condition
The *Complete COVID-19 Screening Questionnaire* (and related use cases), *Scan Access Card/Visitor Barcode*, *Identify User*, *Check Temperature*, and *Mask Recognition* use cases must all have been completed.

Brief Description
The *Grant Access to Campus* use case activates an access token that is linked to a UFS staff member's/student's/visitor's campus access card/barcode that provides the staff member/student/visitor with access to their specified campus for the day on which the token was activated.

Step-by-step Description
1. Activate access token for staff member/student/visitor
2. Update the staff member's/student's/visitor's database record with the access token.
3. Link access token to staff member's/student's/visitor's access card/visitor's barcode.
4. Inform the staff member/student/visitor that the access token has been activated.
5. Open the campus gate.
6. Sign out staff member/student/visitor from COVID-19 gate screening.

Scenarios
- Best-case scenario – an access token is activated for the user
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system activates a valid access token for Joelle.
    - The system updates Joelle's database record with the access token.
    - The system links the access token to Joelle's campus access card.
    - The system grants Joelle access.
    - The system opens the campus gate.
    - The system signs Joelle out of the UFS COVID-19 check.

- Worst-case scenario – System malfunction that deactivates all access tokens
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system activates the access token.
    - The system malfunctions and deactivates Joelle's access token and all other access tokens.
    - The system will not activate any more access tokens.

- Alternative 1 – the system does not update the user's database record and user is not granted access to the campus
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system activates a valid access token for Joelle.
    - Joelle's database record is not updated with the access token.
    - The system does not grant Joelle access.
    - The system signs Joelle out of the COVID-19 gate screening.
    - Joelle must repeat the COVID-19 gate screening process.

- Alternative 2 – the system does not update the user's database record and access card
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system activates a valid access token for Joelle.
    - Joelle's database record is not updated with access token.
    - The system does not link the access token to Joelle's access card.
    - The system opens the campus gate.
    - The system signs Joelle out of the COVID-19 gate screening process.
    - Joelle leaves the campus.
    - Joelle comes back later and scans her access card.
    - The system does not detect Joelle's access token.
    - Joelle must repeat the COVID-19 gate screening process.

- Alternative 3 – the system activates an invalid token for the user
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system activates an invalid access token for Joelle.
    - The system updates Joelle's database record with the invalid access token.
    - The system links the invalid access token to Joelle's campus access card.
    - The system opens the campus gate.
    - Joelle leaves the campus.
    - Joelle comes back later and scans her access card.
    - The system detects that Joelle's access token is invalid.
    - The system does not allow Joelle access to the campus.
    - Joelle must repeat the COVID-19 gate screening process.

- Alternative 4 – the system does not sign out user
    - The system did not sign out the previous staff member/student/visitor.
    - Joelle is granted access to the campus, based on the checks of the previous staff member/student/visitor.

- Alternative 5 – system malfunction that deletes the user's latest check results
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system malfunctions and loses record of Joelle's completed checks.
    - The system does not activate an access token for Joelle.
    - The system signs Joelle out of the UFS COVID-19 check.

- Alternative 6 – the system malfunctions and does not activate an access token
    - Joelle passes her questionnaire, permit, temperature, and mask checks.
    - The system does not activate an access token.

- Joelle cannot access the campus.
- The system signs Joelle out.
- Joelle must repeat the COVID-19 gate screening process.

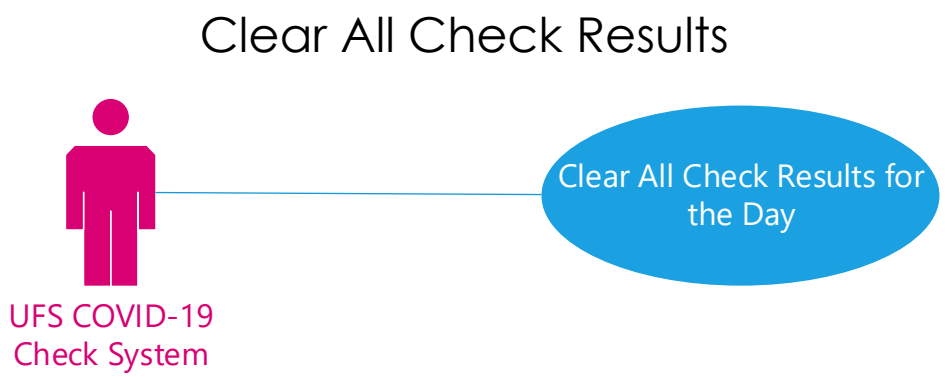


Figure 11 - Clear All Check Results for the Day Use Case

Pre-condition
None

Brief Description
The *Clear All Check Results for the Day* use case clears all questionnaire results, temperature readings, and access tokens at the end of each day thereby preventing campus access.

Step-by-step Description
1. Delete all questionnaire results.
2. Change all questionnaire statuses to "Not completed".
3. Delete all temperature readings.
4. Deactivate all access tokens.

Scenarios
- Best-case scenario
  - At 23:59, the system clears all check results for the day.
    - The system deletes all questionnaire results.
    - The system changes all questionnaire statuses to "Not completed".
    - The system deletes all temperature readings.
    - The system invalidates all access tokens.

- Worst-case scenario – the system does not deactivate access tokens
  - The system does not invalidate access tokens.
  - Joelle can access the campus the next day without passing through the UFS COVID-19 check.

Chapter 3: Software Project Management Plan

1. OVERVIEW

1.1. Project summary

1.1.1. Purpose, scope, and objectives

The objective of this project is to develop a system to streamline and automate the COVID-19 symptom checking process at all University of the Free State (UFS) campus entrances.

The project scope comprises two phases:

- Phase 1 (carried out before the user arrives at a UFS campus): the product will require the user to complete a screening questionnaire, verify a reason for accessing the campus (if the user is a student), and will verify that the user has a valid permit.
- Phase 2 (carried out at the campus gates): the product will authenticate the user, measure the user's temperature, verify that the user is wearing a mask, and activate an access token for the user to gain campus entry throughout the day.

At the end of each day (23:59), all data gathered during the day (i.e. temperature readings, mask-recognition results, access tokens, etc.) will be deleted and/or deactivated.

1.1.2. Assumptions and constraints

The deadline must be met.

The product must be developed within the given budget.

The product must reliably check for COVID-19 symptoms.

The product must be open to adding functionalities as the regulations and information around COVID-19 change.

The product must be user-friendly.

The product must make provision for human intervention to handle inconsistencies, false negatives, false positives, and results that indicate that a user may be infected.

1.1.3. Project deliverables

The complete product must be completed 1,019 hours [3 months] after the project commences, that is:

- the hardware apparatus for the campus gates,
- the system software,
- the system application for security personnel,
- system user database and database management system, and
- user manuals.

1.1.4. Schedule and budget summary

Requirements workflow

- Three team members, 21 hours at R 3,780.00

Analysis workflow

- Three team members, 46 hours at R8,280.00

Design workflow

- Three team members, 52 hours at R9,360.00

Implementation workflow

- Software: Three team members, 675 hours at R121,500.00
- Gate-check apparatus
  - R 48,675.30 per single-armed apparatus x10
  - R 56,890.00 per double-armed apparatus x5
  - R19,500.00 per apparatus for pedestrian gates x6

Testing workflow

- Three team members, 225 hours at R40,500.00

Total development time: 1,019 hours

Total cost

R183,420.00 (software development)
+ R888,203.00 (mechanical apparatuses)

R1 071 623

1.2. Evolution of the project management plan

All changes in the software project management plan must be agreed to by Shayne Hassett, Tshegofatso Mochaki, and Reba Phuthi. The software product management plan must be updated accordingly by Reba Phuthi (project manager). All changes must be documented by Reba Phuthi to keep the software project management plan correct, complete, and up to date. The specifications document and software product management plan will be reviewed after any requirements, specifications, or artifact changes.

2. REFERENCE MATERIALS

Schach, S.R., 2011. *Object-Oriented and Classical Software Engineering*. 8th ed. New York: McGraw-Hill.

3. DEFINITIONS AND ACRONYMS

Client – The University of the Free State

Mechanical gate apparatus – the mechanical device that houses the system hardware components to carry out symptom checks. The device will be situated at campus entrances.

Screening questionnaire – a questionnaire used to screen system users for COVID-19 symptoms before the users arrive at the campus gate. The screening questionnaire is accessible via the client's website.

Security notification application software – the software responsible for notifying security personnel at campus gates of problems with the gate apparatus that require human intervention, and of user metrics that indicate that the user may be infected with COVID-19.

System software – the software responsible for executing the overall software product, including:

- capturing, tracking, and storing user data and metrics,

- interfacing with hardware components,
- capturing data from hardware components,
- interacting and interfacing with system users, and
- interfacing with security personnel through the security notification application software.

System user database – this is the database that will be used to store a user's personal data that pertains to the COVID-19 symptom checking process. The database will also store the following:

- Screening questionnaire results and status
- Permit check status
- Reason provided for campus visit (student)
- Temperature readings
- Mask-recognition results
- Access tokens

UFS – The University of the Free State (Main Campus, and South Campus)

4. PROJECT ORGANISATION

4.1. External interfaces

All work relating to the software components, i.e. the system software and security notification application software, will be performed by Shayne Hassett, Tshegofatso Mochaki, and Reba Phuthi.

All work relating to the building of the mechanical gate apparatus will be carried out by *Techno Engineering*.

The integration of the hardware components into the mechanical gate apparatus will be handled collectively by Shayne Hassett, Tshegofatso Mochaki, Reba Phuthi, and *Techno Engineering*. Integration of software components with the hardware components where necessary will be handled collectively by Shayne Hassett, Tshegofatso Mochaki, Reba Phuthi.

Shayne Hassett, Tshegofatso Mochaki, and Reba Phuthi will meet with the client representative(s) on a bi-weekly basis to report on the software product and discuss possible changes and modifications.

4.2. Internal structure

The development team consists of Reba Phuthi (project manager), Shayne Hassett, Tshegofatso Mochaki, and Rouxan Fouché (client representative).

4.3. Roles and responsibilities

Shayne Hassett, Tshegofatso Mochaki, and Reba Phuthi will perform the design, implementation, and testing workflows.

- Shayne Hassett will be responsible for constructing the artifacts that integrate the results of the screening questionnaire, permit check, and related functionalities into the database management system and that verify screening questionnaire results at the campus gates.
- Tshegofatso Mochaki will be responsible for constructing the artifacts to handle temperature measurement and mask recognition.
- Reba Phuthi will be responsible for constructing the artifacts that pertain to authenticating system users, activating access tokens, and clearing all check results at the end of each day.

Each team member is responsible for the quality of the software components he or she produces. Integration testing will be performed collectively by all team members.

5. MANAGERIAL PROCESS PLANS

5.1. Start-up plan

5.1.1. Estimation plan

The total development time is estimated at 1,019 hours with a total internal cost of R1 071 623. The figures were obtained using the Bottom-up Approach where the development effort of each artifact was quantified to obtain the cost of the artifact. The cost of the artifacts for each workflow were thereafter combined to provide an overall figure for the workflow and, consequently, the project.

5.1.2. Staffing plan

Shayne Hassett, Tshegofatso Mochaki and Reba Phuthi are needed for the entire duration of the project development. Shayne Hassett, Tshegofatso Mochaki and Reba Phuthi be required as system analysts and designers for the requirements, analysis and design workflows. Thereafter, Shayne Hassett, Tshegofatso Mochaki and Reba Phuthi will be required as programmers and testers for the implementation and testing workflows. Reba Phuthi will be required as the project manager for the entire development process.

5.1.3. Resource acquisition plan

Hardware:

The hardware necessary to measure temperature (infrared camera) and to conduct mask recognition (digital web camera) will be purchased from our supplier partner, *Hikvision*. The client already has the monitors needed for the user interface monitors and access card scanners available. The gate-check mechanical apparatus, which will house all the hardware components, will be constructed by *Techno Engineering*.

Software:

The following software will be constructed by the development team:

- System software
- Security notification application software

- System user database and database management system using Microsoft SQL Server.

The client already has the following software/database systems available:

- Screening questionnaire software
- UFS timetable database

The following software will be purchased and integrated into the system software:

- Infrared technology software. This technology comes embedded in the infrared camera. *Hikvision* will provide the integration method (API key).
- Image recognition software

The product will be delivered as a mechanical apparatus with all the hardware components integrated. The system software will run on a system computer located in the security office at each campus gate. The mechanical apparatus will be installed at each campus gate at all the UFS campuses. The security notification application will be installed on handheld devices to be used by security personnel.

5.1.4. Project staff training plan

No additional staff training is necessary.

5.2. Work plan

5.2.1.– 5.2.2. Work activities and Schedule allocation

Requirements workflow (21 hours) – development team met with the client, obtained client requirements, determined and compiled requirements artifacts, and reviewed requirements artifacts.

- **Status:** Complete

Analysis workflow (46 hours) – development team produced and reviewed analysis artifacts. The artifacts (specifications document) were shown to the client for approval. Once approved, the software product management plan was compiled and reviewed.

- **Status:** Complete

Design workflow (52 hours) – each member of the development will produce his/her assigned design artifacts. The team will collectively review design artifacts.

- **Status**: In progress

Implementation and testing workflows (900 hours) – implementation and inspection of artifacts. Each team member will be responsible for unit testing as it pertains to their assigned artifacts. The team members will also perform unit testing on one another's artifacts. The team will collectively perform integration and product testing. Documentation will be reviewed.

- **Status**: In progress

5.2.3. Resource allocation

Each team member will work separately on their assigned artifacts as far as possible. Team members will work collectively on specific artifacts as the need arises. Reba Phuthi will be responsible for tracking daily and overall project progress (scheduling), adherence to budget constraints, and subcontractor progress. Team members will meet daily at the end of the day to discuss any problems with, and progress of, their assigned artifacts. Formal meetings between Shayne Hassett, Tshegofatso Mochaki, Reba Phuthi, and the client representative(s) will be held on a bi-weekly basis to report on the software product progress and discuss possible changes and modifications.

Team members will collectively perform risk management on each artifact. This will include carrying out unit testing, integration testing, and product testing. Reba Phuthi will primarily be responsible for ensuring adherence to the specifications document, the software product management plan, and the constraints outlined therein. Reba Phuthi has overall responsibility for all project documentation and ensuring that the project documentation remains correct, complete, and up to date.

5.2.4. Budget allocation

Requirements workflow	R3,780.00
Analysis workflow	R8,280.00
Design workflow	R9,360.00
Implementation workflow	R1 009 703.00
Testing workflow	R40,500.00
Total	**R1 071 623**

5.3. Control plan

Any changes to requirements, specifications, schedule, or budget must be agreed upon by all development team members. In other words, consensus must be reached between team members. Once consensus has been reached, the team will inform the client of the extension(s) for approval. If approval is obtained, Reba Phuthi will document the change in requirements and update the specifications document.

No external quality assurance group will be involved. The team members will test each other's work, and where necessary, employ the pair programming approach.

Reba Phuthi will be responsible for tracking project progress, expenditure, and quality reviews. This will be accomplished through daily, team meetings. At each meeting, Hassett, Mochaki, and Phuthi will report on the day's progress and any problems encountered. Using this information, Phuthi will determine if the team is progressing as expected and whether

their development tasks adhere to the specifications document and software product management plan. Any major problems will be reported during the daily meetings and the necessary steps to resolve the problem will be carried out in time for the next day's meeting.

5.4. Risk management plan

No similar software product exists to use as a model for risk management. Therefore, extensive testing carried out by all development team members will be used as a mechanism to mitigate potential risks.

A software component will undergo unit testing during and after it has been completed. Unit testing will initially be carried out by the team member to whom the artifact is assigned. Following the completion of the artifact, another programmer will test the artifact. The members of the development team will collectively perform integration testing and product testing. Reba Phuthi will be responsible for monitoring adherence to the specifications document, software product management plan, and the constraints outlined therein.

Any risk related to hardware failure will be mitigated as per the service agreement with *Techno Engineering* and warranties outlined by *Hikvision*.

5.5. Project close-out plan

Only applicable once the client discontinues the use of the software product.

6. TECHNICAL PROCESS PLANS

6.2. Process model

The Unified Process methodology and Iterative-and-Incremental Life-cycle model will be used.

6.3. Methods, tools, and techniques

The workflows of the Unified Process will be applied. The product will be implemented in Java and C#. The database and database management system will be implemented using Microsoft SQL Server.

6.4. Infrastructure plan

The product will be developed using IntelliJ and Visual Studio running under Linux on personal computers. The system software will run on Linux OS on a computer located in the security office at each campus gate. The user interface of the product will be cast to all gate screening mechanical apparatuses at the campus gates. The security notification application to be used by security personnel will be installed on handheld devices and managed by the system software.

6.5. Product acceptance plan

Acceptance of the product by the client will be achieved by following the steps of the Unified Process.

7. SUPPORTING PROCESS PLANS

7.2. Configuration management plan

Git will be used throughout for all artifacts.

7.3. Testing plan

Testing will be conducted according to the activities of the testing workflow of the Unified Process.

7.4. Documentation plan

Project documentation will be produced as specified in the Unified Process.

7.5. Quality assurance plan

Shayne Hassett, Tshegofatso Mochaki, and Reba Phuthi will each test their assigned artifacts and each other's code (unit testing). Integration and product testing will be conducted by all three development team members. Reba Phuthi will also oversee the integration and product testing to ensure adherence to the specifications document and compliance will the quality standards and constraints outline therein.

7.6. Reviews and audits plan

Not applicable.

7.7. Problem resolution plan

Any major problems will be reported during the daily meetings.

7.8. Subcontractor management plan

The sub-contractors, *Techno Engineering,* will be given a schedule to follow (as per a service agreement) to ensure adherence to the time constraints specified in the product management plan. The schedule will also be used as a means to keep the progress of the software implementation on track to integrate all hardware and software components of the product in a timely fashion.

7.9. Process improvement plan

Not applicable here.

8. Additional plans

Security: The credentials of a system administrator will be required to access the system user database.

Training: Security personnel will undergo training on how to use the security notification application system, and override aspects of the system software when necessary. Training will be performed by Shayne Hassett, Tshegofatso Mochaki, and Reba Phuthi when the product is delivered and has undergone acceptance testing. The training will be conducted over one workweek (5 days).

Maintenance: Corrective maintenance on the software components will be performed by the team at no cost for 6 months. Maintenance of hardware components will be performed as per the subcontractor service agreement.

Chapter 4: Quotation

HMP Solutions Ltd.
Phone: 051-447-1234
Email address: quotation@hmp.co.za

Date: 17 November 2020
Quotation # 1742
Customer ID

Quotation valid until:
01 December 2020

Quotation For:
Department of Information and Communications Technology
University of the Free State
205 Nelson Mandela, Park West
Bloemfontein
9301

051-401-2000

Description	Quantity	Price	Tax	Total
Requirements	1	ZAR 5,040	0 %	ZAR 5,040
Analysis	1	ZAR 11,040	0 %	ZAR 11,040
Design	1	ZAR 13,920	0 %	ZAR 13,920
Implementation (Software)	1	ZAR 121,500	15 %	ZAR 139,725
Single-arm gate screening appraratus	10	ZAR 48,675.30	15 %	ZAR 559,765.95
Double-arm gate screening apparatus	5	ZAR 56,890	15 %	ZAR 327,117.50
Pedestrian gate screening apparatus	6	ZAR 19,500	15 %	ZAR 134,550
Testing	1	ZAR 54,000	0 %	ZAR 54,000
Support Software	1	ZAR 6,956.52	15 %	ZAR 8,000
General Expenses	1	ZAR 12,228	0 %	ZAR 12,228
Other - additional client rep consult	1	ZAR 960	0 %	ZAR 960

Total **ZAR 1,266,346.45**

Overview of development effort

Requirements and Analysis Workflows

The initial project proposal developed through during the Requirements workflow has been substantially redefined using iteration and incrementation. Once the analysis workflow commenced, certain functionalities included in the project proposal were deemed beyond the scope of the software product, and thus, the functionalities were removed. The inherent nature of the iterative-and-incremental model was never more evident than during the analysis workflow. Each use case, step-by-step descriptions and the accompanying scenarios went through up to four iterations before the final, comprehensive use case diagram was compiled.

Design Workflow

The most iteration and incrementation was applied during the development and compilation of the Class-Responsibility-Collaboration (CRC) cards and class diagram, respectively. Noun extraction was used to identify possible classes to be represented by CRC cards. Each CRC card was thereafter populated with the responsibilities, collaborators, attributes and relationships for each class. This information was used to develop the class diagram. The best-case scenarios of each use case were depicted using sequence diagrams. The information thereof, together with the responsibilities indicated in the CRC cards, was then used to populate the class methods in the class diagram. Upon inspection of the relationship in the class diagram, modifications were made to the CRC cards depicting the changes in collaborators and relationships between the classes (iteration). Finally, the stereotypes of the classes in the class diagram were depicted using a stereotype diagram and the events which result in the changing states of the software product were depicted in a statechart diagram.

Total Hours spent

Requirements	21 hours
Specification / Analysis	46 hours
Design	58 hours
Implementation	675 hours
Other	Consulting with "client representative" outside of workflows (Mr Rouxan Fouché): • 4 hours
Total	**804 hours**

*Total man-hours indicated in SPMP = 1,019 hours (inclusive of Testing workflow)

Chapter 5: Design Documentation

Class diagram

Noun Extraction

Step 1: Describe the Software Product

The System Software reads a user's answers received from the UFS Screening Questionnaire and evaluates them to ensure that user does not have COVID-19 symptoms. The System Software stores the result in the System Database. The System software performs user permit checks to ensure that a user has a valid permit. If the user is a student, the system software verifies the reason for their campus visit. User identification is performed when the system scans a user's access card/visitor barcode at the campus gate. System verifies completion of screening questionnaire by the user and the results thereof. Measurement of user's temperature through an infrared camera and mask recognition through a digital camera is performed. System software activates access token and grants user access to the campus. System clears all check results of the day at 23:59.

Step 2: Identify the Nouns

The **System Software reads** a user's answers **received** from the **UFS Screening Questionnaire** and **evaluates** them to ensure that user does not have **COVID-19 symptoms**. The System Software **stores** the result in the **System Database**. The System software performs **user permit checks** to ensure that a user has a valid permit. If the user is a **student**, the system software verifies the **reason** for their **campus visit**. User **identification** is performed when the system **scans** a user's **access card/visitor barcode** at the campus gate. System **verifies completion** of screening questionnaire by the user and the results thereof. **Measurement** of user's **temperature** through an infrared camera and **mask recognition** through a digital camera is performed. System software **activates access token** and **grants** user **access** to the campus. System **clears** all **check results** of the day at 23:59.

Nouns	Abstract Nouns
1. System software 2. UFS Screening Questionnaire 3. COVID-19 symptoms 4. System Database?	1. Evaluates, verifies, reads, stores 2. Receives 3. Reads 4. Stores?

5. Permit 6. User 7. Student 8. Campus visit 9. Access card/visitor barcode 10. Temperature 11. Mask 12. Access tokens 13. Campus access 14. Check results (of the day)	5. Check 6. Identification 7. Reason (for campus visit) 8. Reason 9. Scan 10. Measure 11. Recognition 12. Activates 13. Grant 14. Clears

Class-responsibility-collaboration (CRC) Cards

Visitor Barcode Generator

Class Name: Visitor Barcode Generator	**ID:** 1	**Stereotype:** Control
Description: Generates a visitor's barcode for visitor user with a special, temporary permit.		**Associated Use Cases:** 0
Responsibilities		**Collaborators**
• System asks for reference code • System loads user details • Generate barcode • Update user database record		• Visitor
Attributes:		
• VisitorName (string) • VisitorSurname (string) • VisitorRefCode (string) • VisitorNationalIdNo (int) • VisitorBarcode (img)		
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	Visitor	
Other Associations:	N/A	

COVID-19 Symptoms

Class Name: COVID-19 Symptoms	**ID:** 2	**Stereotype:** Entity
Description: Stores and manages (adds, removes, edits) a list of COVID-19 symptoms		**Associated Use Cases:** 1
Responsibilities		**Collaborators**
• Add symptom • Remove symptom • Edit symptom Update database		• UFS Screening Questionnaire
Attributes:		
• Symptom ID (string) • Symptom Description (string) • SymptomsList (List<T>)		
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	N/A	
Other Associations:	N/A	

Screening Questions

Class Name: UFS Screening Questionnaire	**ID:** 3	**Stereotype:** Entity
Description: Stores and manages (adds, removes, edits) a list of screening questions.		**Associated Use Cases:** 1
Responsibilities		**Collaborators**
• Add question • Remove question • Edit question • Edit list of reasons • Update System Database		• UFS Screening Questionnaire
Attributes:		
• QuestionID (int) • QuestionDescription (string)		
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	UFS Screening Questionnaire	
Other Associations:	N/A	

UFS Screening Questionnaire

Class Name: UFS Screening Questionnaire	**ID:** 4	**Stereotype:** Boundary
Description: Accepts input from the online UFS COVID-19 Screening Questionnaire		**Associated Use Cases:** 1
Responsibilities		**Collaborators**
• Prompt users with screening questions • Verify user input (Error-Checking) • Prompts student users for campus visit reason • Prompt student user for campus-visit OTP • Save answers		• Validate Questionnaire
Attributes:		
• QuestionID (int) • QuestionAndAnswer (Dictionary<string, string>) • UserID (string)		• CampusVisitOTP (int) • CampusVisitReasons (Dictionary<string, string>)
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	Validate Questionnaire	
Other Associations:	N/A	

Permit Check

Class Name: Permit Check	**ID:** 5	**Stereotype:** Control
Description: Confirms if the user has a valid permit issued to them.		**Associated Use Cases:** 1

Responsibilities	**Collaborators**
• Verify permit status	• Validate Questionnaire
Attributes:	
• UserId (string) • PermitId (string) • isValidPermit (boolean) • PermitStatus (string)	
Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	Validate Questionnaire
Other Associations:	N/A

User

Class Name: User	**ID:** 6	**Stereotype:** Entity
Description: People that will be interacting with the system.		**Associated Use Cases:** 10

Responsibilities	**Collaborators**
• Display user details • Update database record	• UFS Screening questionnaire • Permit Check • Validate Questionnaire • Scan Access Card/Visitor Barcode • Identify User • User Interface • Measure Temperature • Mask Recognition • Activate Access Token
Attributes:	
• UserId (string) • UserName (string) • UserSurname (string) • UserNationalId (boolean) • UserType (string) • hasValidAccessToken (boolean)	• HasCompletedScreening (boolean) • hasValidPermit (boolean) • hasPassedTemperatureCheck (boolean) • hasMaskOn (boolean) • LatestTemperature (double)
Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	N/A
Other Associations:	N/A

Student

Class Name: Student	**ID:** 6.1	**Stereotype:** Entity
Description: People classified as students that will be interacting with the system.		**Associated Use Cases:** 1
Responsibilities		**Collaborators**
• Capture OTP • Capture valid reason for campus visit • Access student timetable database		• UFS Screening questionnaire • Reason for Campus Visit • Validate Questionnaire
Attributes:		
• StudentNumber (string) • CampusVisitReason(string) • OTP (int)		
Relationships:		
Generalization (a-kind-of):	User	
Aggregation (has-parts/is-part-of):	Validate Questionnaire	
Other Associations:	N/A	

Staff

Class Name: Staff	**ID:** 6.2	**Stereotype:** Entity
Description: People classified as staff that will be interacting with the system.		**Associated Use Cases:** 0
Responsibilities		**Collaborators**
		• UFS Screening Questionnaire
Attributes:		
StaffNumber(string)		
Relationships:		
Generalization (a-kind-of):	User	
Aggregation (has-parts/is-part-of):	N/A	
Other Associations:	N/A	

Visitor

Class Name: Visitor	**ID:** 6.3	**Stereotype:** Entity
Description: People classified as visitors that will be interacting with the system.		**Associated Use Cases:** 0

Responsibilities	Collaborators
	• UFS Screening Questionnaire • Visitor Barcode Generator
Attributes:	
VisitorBarcode(image)	
Relationships:	
Generalization (a-kind-of):	User
Aggregation (has-parts/is-part-of):	N/A
Other Associations:	N/A

Reason for Campus Visit

Class Name: Reason for campus visit	**ID:** 7	**Stereotype:** Entity
Description: Maintains a list for the valid reasons for a campus visit by a student.		**Associated Use Cases:** 1
Responsibilities		**Collaborators**
• Add valid reason • Remove reason • Edit list of reasons • Update System Database		• UFS Screening Questionnaire • Student
Attributes:		
ReasonId (int) ReasonDescription (string)		
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	UFS Screening Questionnaire	
Other Associations:	N/A	

Validate Questionnaire

Class Name: Validate Questionnaire	**ID:** 8	**Stereotype:** Control
Description: accepts input from the *UFS Screening Questionnaire, Permit Check,* and *Student* classes to compare questionnaire answers to a list of symptoms, verify that the user has a valid permit and verify a student user's campus-visit reason.		**Associated Use Cases:** 3
Responsibilities		**Collaborators**

• Evaluate Screening Questionnaire Answers • Validate permit status • Verify Reason for Campus Visit Update User Database Record	• UFS_Screening Questionnaire • COVID-19 Symptoms • Permit Check • User
Attributes:	
• UserId (string) • QuestionnaireAnswers (Dictionary <string, string>) • HasPassedSymptomCheck (boolean)	• HasValidPermit (boolean) • HasValidCampusVisitReason (Boolean)
Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	N/A
Other Associations:	N/A

User Interface

Class Name: User interface	**ID:** 9	**Stereotype:** Boundary
Description: Displays system prompts, error messages and results of the user's COVID-19 check process.		**Associated Use Cases:**
Responsibilities		**Collaborators**
• Prompt user to scan access card • Display User details • Prompt user to face screen for temperature and mask checks • Display check process progress • Display measured temperature • Display mask recognition results • Display access token granted message • Display error messages • Sign user out		• Scan Access Card • User
Attributes:		
• UserName (string) • UserSurname (string) • UserId (string) • Temperature (decimal) • HasMaskOn (boolean) • Error Message (string)		
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	N/A	

Other Associations:	N/A

Scan access card/visitor barcode

Class Name: Scan Access Card/Visitor Barcode	**ID:** 10	**Stereotype:** Boundary
Description: Scans a student's or staff member's access card, or a visitor's barcode		**Associated Use Cases:** 1

Responsibilities	**Collaborators**
• Scan access card/visitor barcode • Capture unique identifier	• Identify User
Attributes:	
• UniqueUserId (var)	
Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	Identify User
Other Associations:	N/A

Identify User

Class Name: Identify User	**ID:** 11	**Stereotype:** Control
Description: Scans a student's or staff member's access card, or a visitor's barcode		**Associated Use Cases:** 1

Responsibilities	**Collaborators**
• Access System Database • Identify user	• Identify User • User
Attributes:	
• UniqueUserId (var)	
Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	N/A
Other Associations:	N/A

Measure Temperature

Class Name: Measure Temperature	**ID:** 12	**Stereotype:** Control
Description: The system checks the temperature of the user.		**Associated Use Cases:** 1

Responsibilities	**Collaborators**
• Align user's face in camera frame • Take temperature reading using infrared cameras • Compares readings to min and max range • Generate the results • Override temperature • Update User class	• Security Notification Application • User • Control Mechanical Apparatus
Attributes:	
• IsLightOn (boolean) • MinTemp (boolean) • MaxTemp (boolean) • isTempTooHigh (boolean)	• isTempTooLow (boolean) • isValidTemperature (boolean) • HasTemperatureBeenOverriden (boolean) • UserId (string)
Relationships:	

Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	Control Mechanical Apparatus
Other Associations:	N/A

Mask Recognition

Class Name: Mask recognition	**ID:** 13	**Stereotype:** Control
Description: The system checks if the user has a mask on.		**Associated Use Cases:** 1

Responsibilities	**Collaborators**
• Switch on the facial recognition camera • Align user's face in camera frame • Scan user's face • Perform mask recognition • Update User class	• Security Notification Application • User • Control Mechanical Apparatus
Attributes:	
• IsLightOn (boolean) • FacialScan (img) • HasMaskOn (boolean)	• HasMaskCheckResultBeenOverriden (boolean) • UserId (int)
Relationships:	

Generalization (a-kind-of):	N/A

Aggregation (has-parts/is-part-of):	Control Mechanical Apparatus
Other Associations:	N/A

Activate Access Token

Class Name: Activate Access Token	**ID:** 14	**Stereotype:** Control
Description: Activates an access for a user once the user has passed the COVID-19 check process		**Associated Use Cases:** 2

Responsibilities	**Collaborators**
• Verify that user has passed check process • Activate access token • Update user database record • Link access token to access card/visitor barcode identifier	• User • Grant Campus Access

Attributes:	
• UserID (string) • AccessToken (var) • HasValidToken (boolean)	

Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	Grant Campus Access
Other Associations:	N/A

Grant Campus Access

Class Name: Grant Campus Access	**ID:** 15	**Stereotype:** Control
Description: Opens the campus gate if the user has been granted an access token		**Associated Use Cases:** 1

Responsibilities	Collaborators
• Open campus gate	• Activate Access Token

Attributes:	
• User Id • IsValidAccessToken (boolean)	

Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	N/A

Other Associations:	N/A

Security Notification System

Class Name: Security Notification Application	**ID:** 16	**Stereotype:** Boundary
Description: Displays error messages related to temperature measurements and mask recognition results that must be handled by security personnel at the campus gates.		**Associated Use Cases:** 2

Responsibilities	**Collaborators**
• Send message: unable to scan card • Send message: inconclusive temperature reading • Send message: temperature too low message • Send message: temperature is abnormally high • Send message: temperature reading failed • Send message: inconclusive mask recognition reading • Send message: mask recognition failed • Send message: mask not worn • Accept input to override temperature/mask recognition results	• Scan Access Card/Visitor barcode • Measure Temperature • Mask Recognition

Attributes:	
• UserID (string) • GateCheckApparatusID (int) • HasMaskOn (boolean) • IsTempTooLow (boolean)	• IsTempTooHigh(boolean) • OverrideTemperature (boolean) • Temperature (decimal)

Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	Measure Temperature, Mask Recognition
Other Associations:	N/A

Clear System Check

Class Name: Clear System Check Results	**ID:** 17	**Stereotype:** Control
Description: Clears all questionnaire results, questionnaire completion statuses, temperature readings, mask recognition results from the System Database and invalidates all access tokens at the end of each day.		**Associated Use Cases: 0**

Responsibilities	**Collaborators**
• Clear questionnaire results	

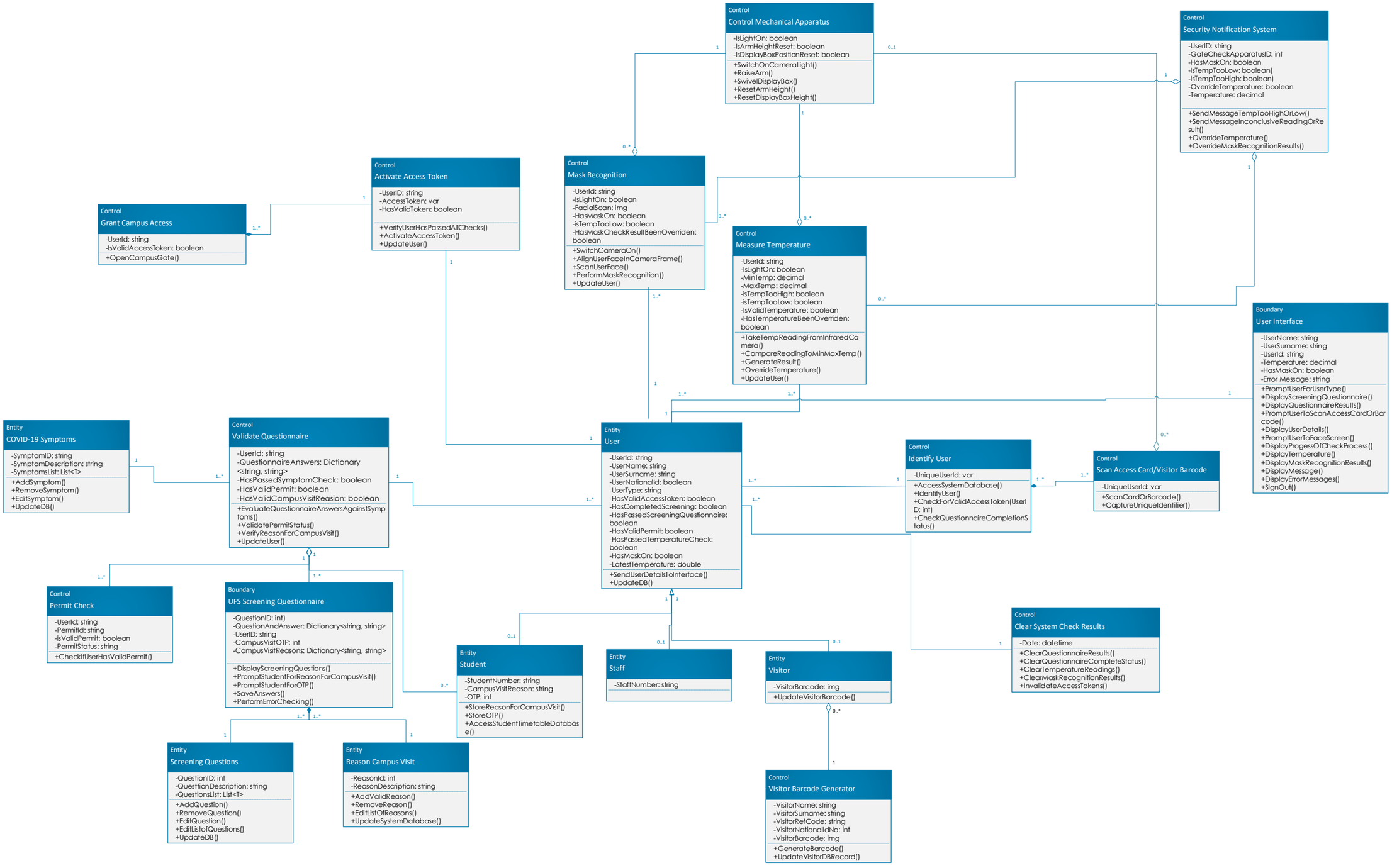
• Clear questionnaire completion statuses • Clear temperature readings • Clear mask recognition results • Invalidate access tokens	
Attributes:	
• Date (datetime)	
Relationships:	
Generalization (a-kind-of):	N/A
Aggregation (has-parts/is-part-of):	N/A
Other Associations:	N/A

Control Mechanical Apparatus

Class Name: Control Mechanical Apparatus	**ID:** 18	**Stereotype:** Control
Description: Controls the gate screening apparatus's motorised arm and display box swivel mechanism.		**Associated Use Cases:** 0
Responsibilities		**Collaborators**
• Switch on green light • Reset arm height • Raise arm • Reset display box head position • Swivel display box head		
Attributes:		
• IsLightOn (boolean) • IsArmHeightReset (boolean) • IsDisplayBoxPositionReset (boolean)		
Relationships:		
Generalization (a-kind-of):	N/A	
Aggregation (has-parts/is-part-of):	Scan Access Card, Measure Temperature, Mask Recognition	
Other Associations:	N/A	

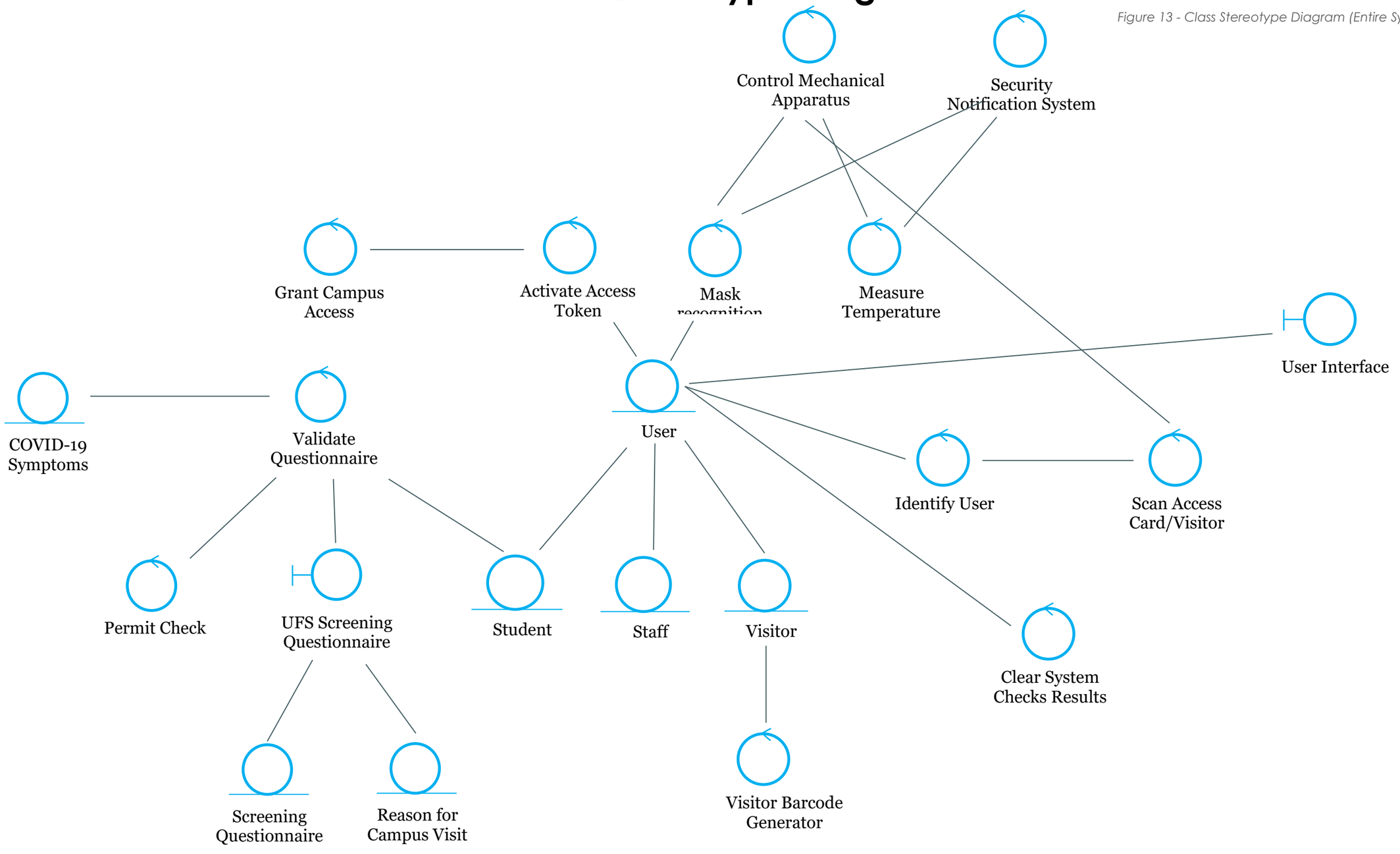
Class Diagram

Figure 12 - Class Diagram



Class Stereotype Diagram

Figure 13 - Class Stereotype Diagram (Entire System)



Pseudocode

Class: MeasureTemperature - Method: CompareReadingToMinMaxTemp

```
void CompareReadingToMinMaxTemp(decimal Temp)
{
     If(Temp is less than MinTemp)
     {
          IsTempTooLow is true;
     }
     Else if (Temp is greater than MaxTemp)
     {
          IsTempTooHigh is true;
     }
     Else
     {
          IsValidTemperature is true;
     }
}
```

Class: MeasureTemperature - Method: GenerateResult

```
string GenerateResult(decimal Temp)
{
     String result;
     Decimal UserTemperature

     CompareReadingToMinMaxTemp(Temp);

     If(IsTempTooLow is true)
     {
          Set result equal to "Your temperature is too low";
          Set UserTemperature equal to Temp
     }

     Else if(IsTempTooHigh is true)
     {
          Set result equal to "Your temperature is too high";
          Set UserTemperature equal to Temp
     }

     Else
     {
          Set result equal to "Your temperature is in the normal
range";
          Set UserTemperature equal to Temp;
     }
     return result;
}
```

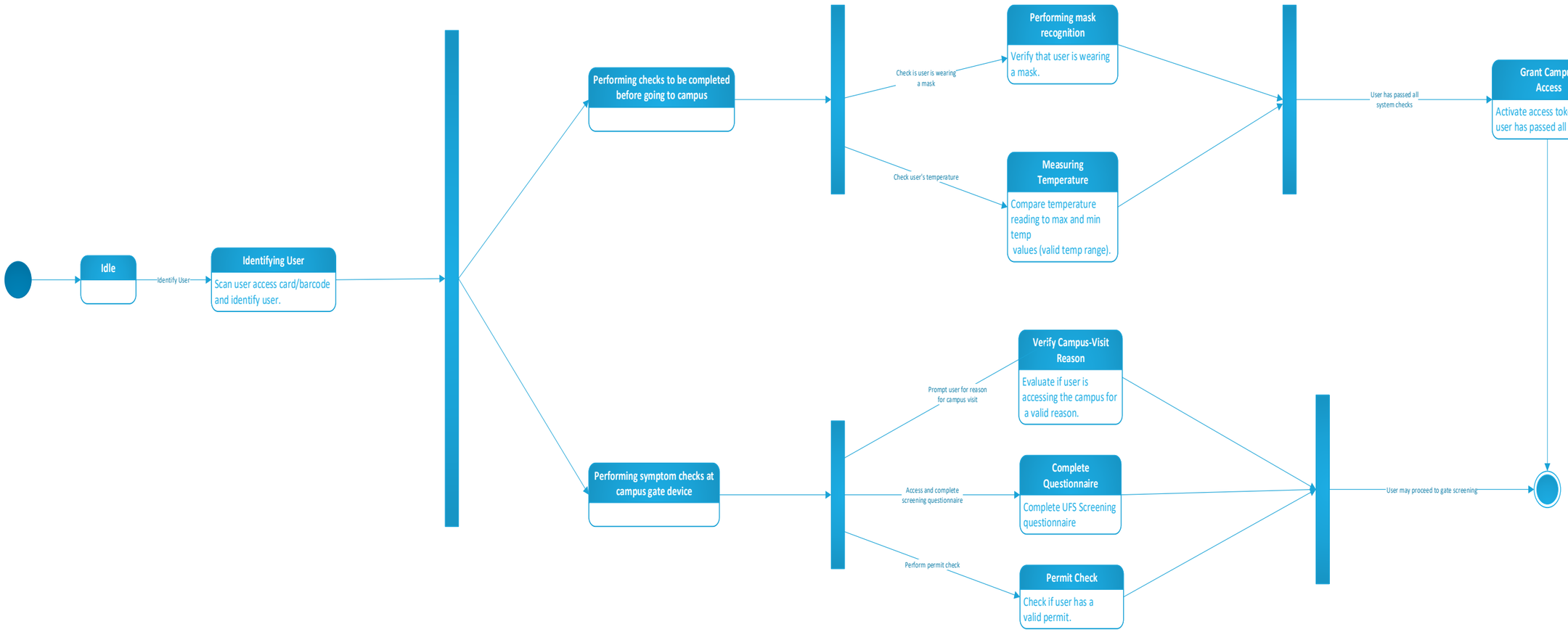
Class: ActivateAccessToken - Method: VerifyUserHasPassedAllChecks

```
boolean VerifyUserHasPassedAllChecks(mockuser of type User)
{
     If(mockuser.HasCompletedScreening is true
          AND mockuser.HasPassedScreeningQuestionnaire is true
          AND mockuser.HasPassedTemperatureCheck is true
          AND mocksuer.HaskMaskOn is true)
     {
          ActivateAccessToken();
          return true;

     }
     return false;
}
```

State Chart Diagram

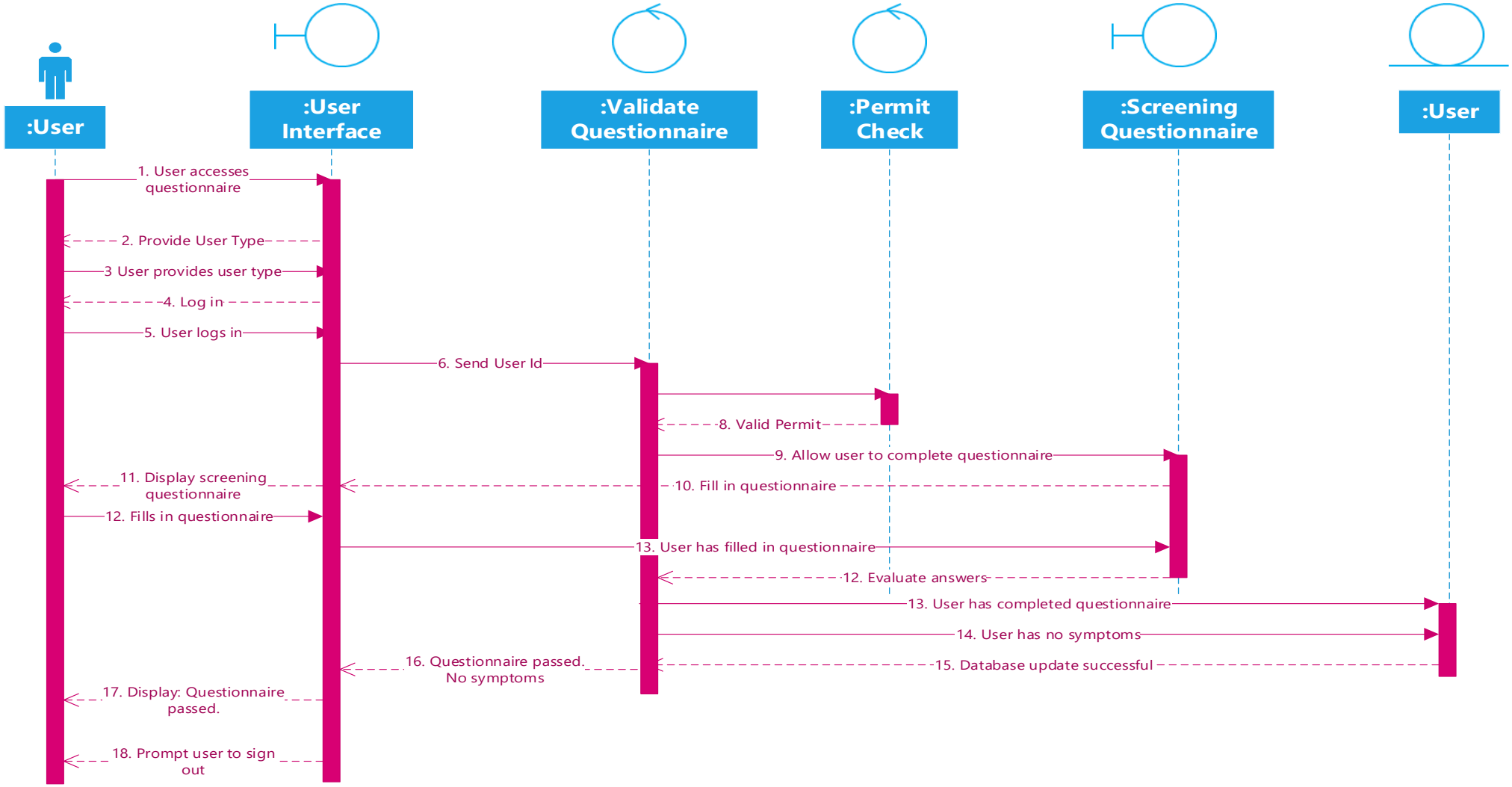
Figure 14 - Statechart Diagram (Entire System)



Sequence Diagrams

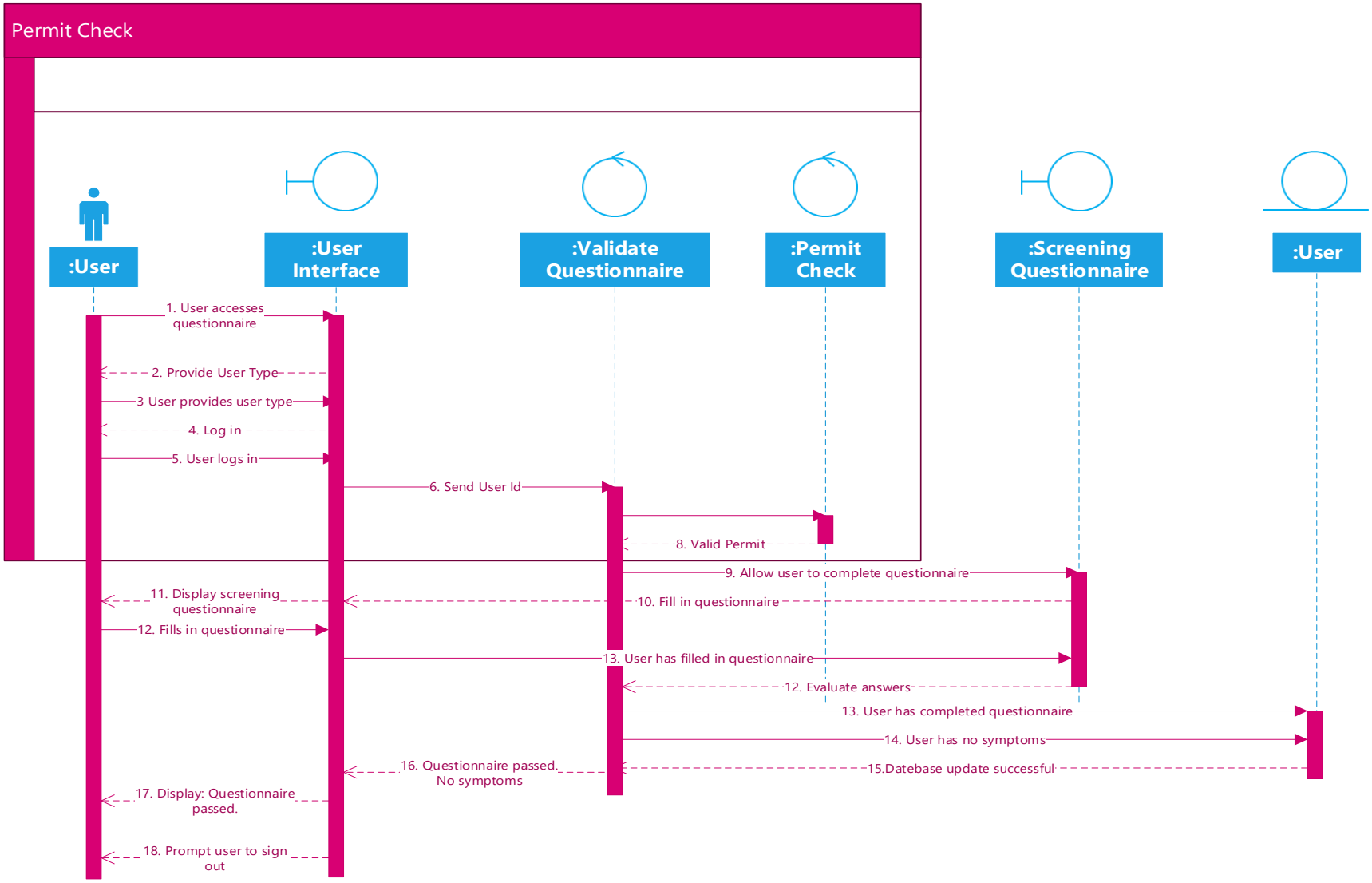
Complete COVID-19 Screening Questionnaire

Figure 15 - Complete COVID-19 Screening Questionnaire Sequence Diagram



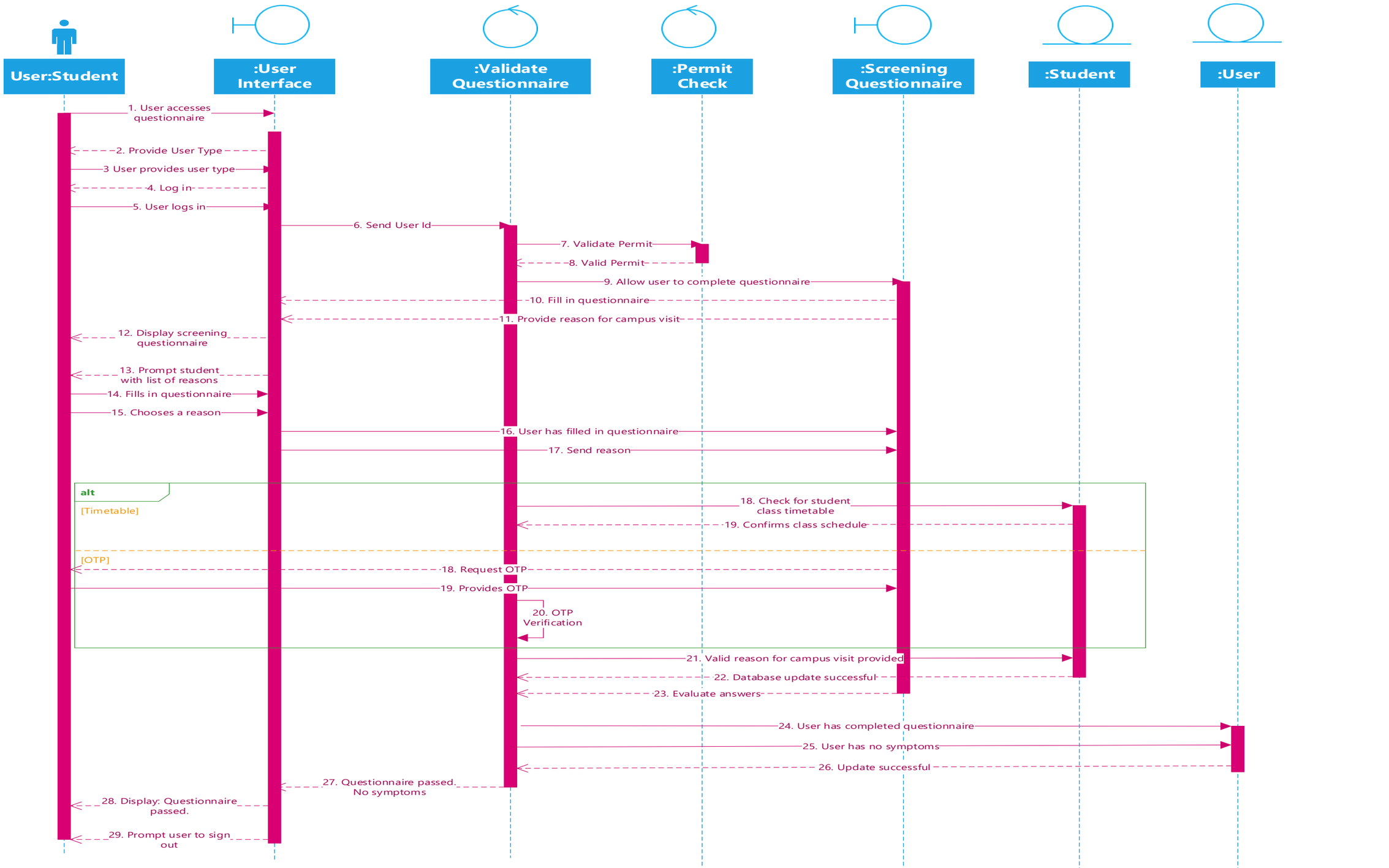
Check Permit

Figure 16 - Check Permit Sequence Diagram



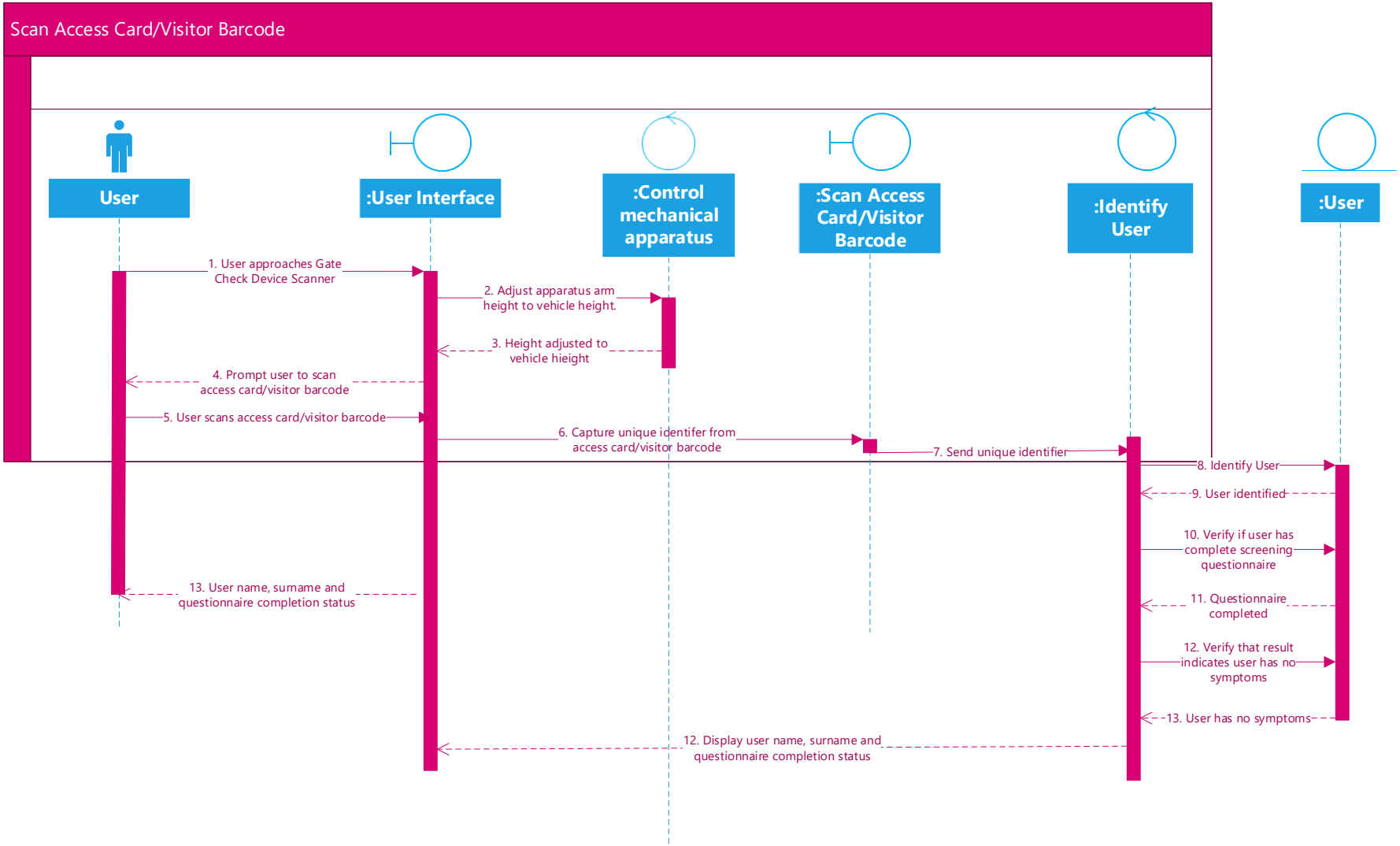
Verify Reason for Campus Visit

Figure 17 - Verify Reason for Campus Visit Sequence Diagram



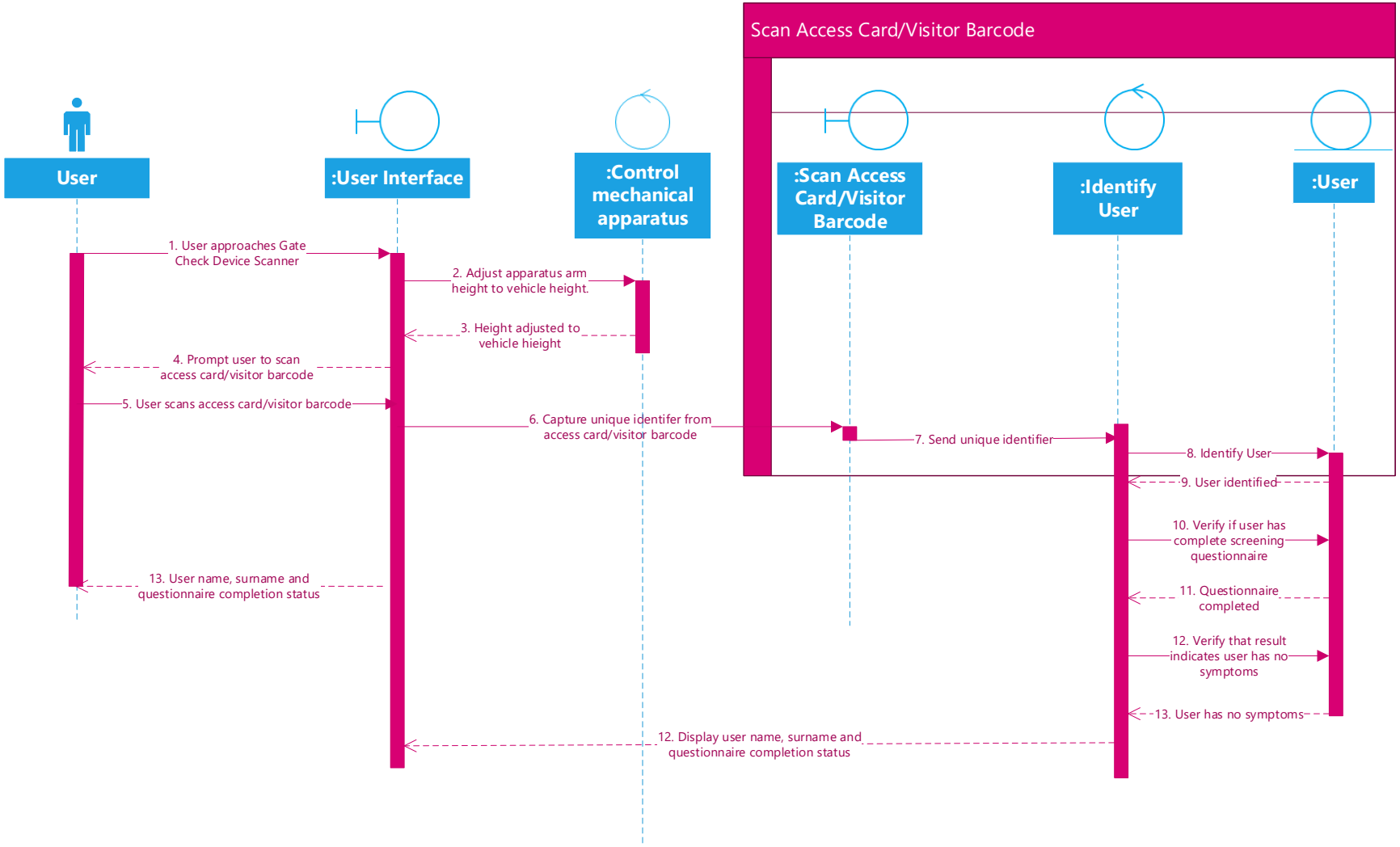
Scan Access Card/Visitor Barcode

Figure 18 - Scan Access Card/Visitor Barcode Sequence Diagram



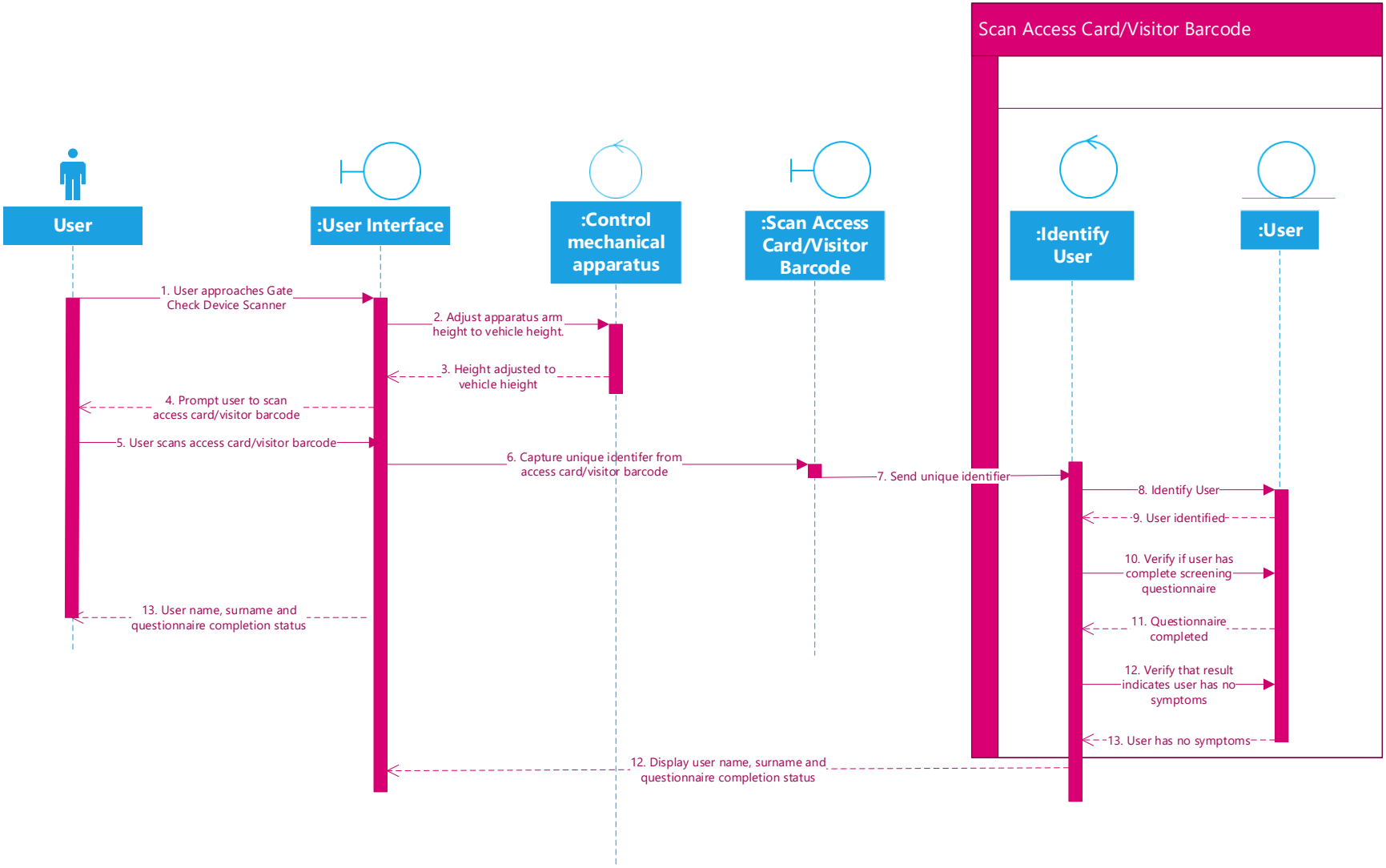
Identify User

Figure 19 - Identify User Sequence Diagram



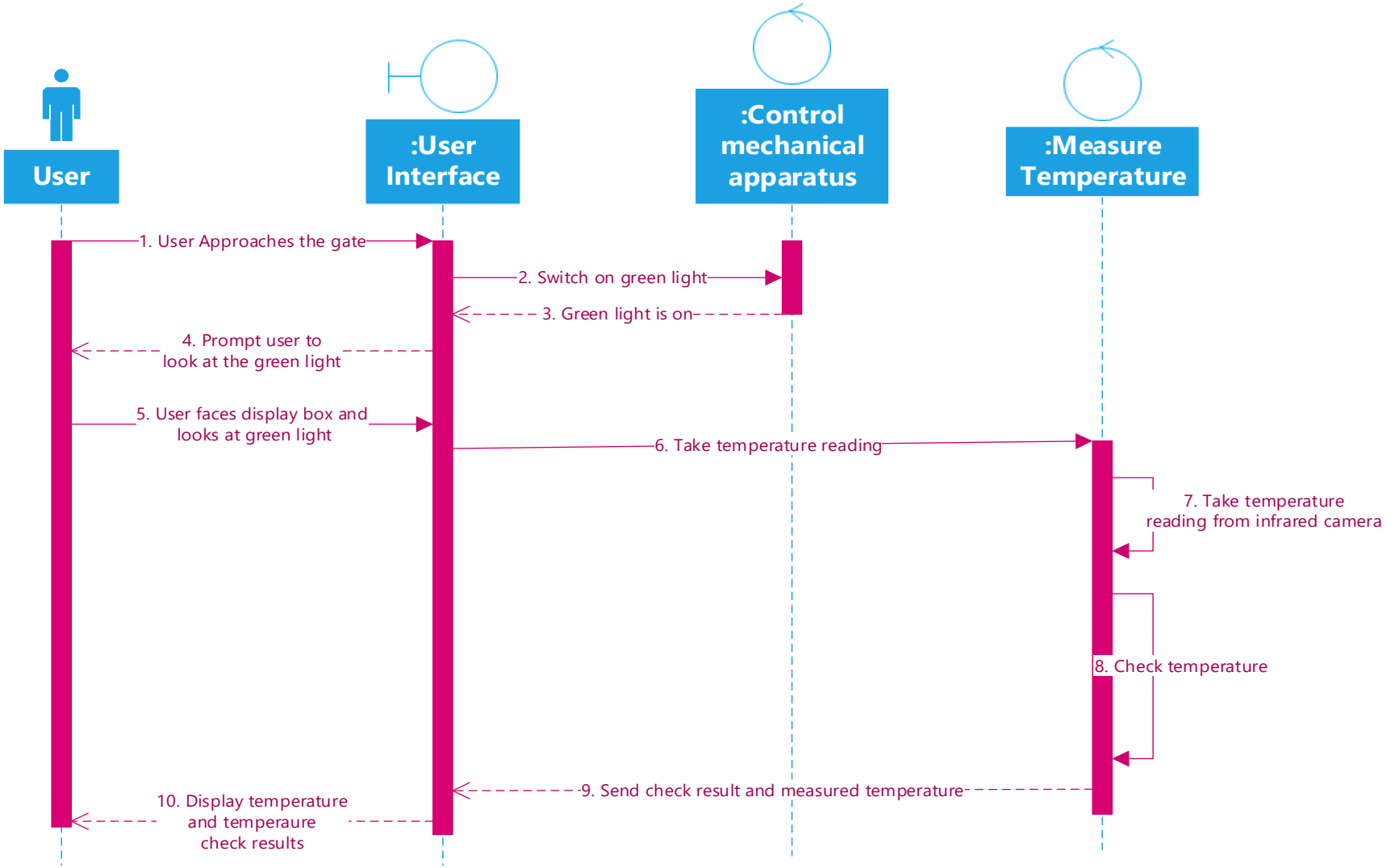
Verify COVID-19 Screening Questionnaire Results

Figure 20 - Verify COVID-19 Screening Questionnaire Results Sequence Diagram



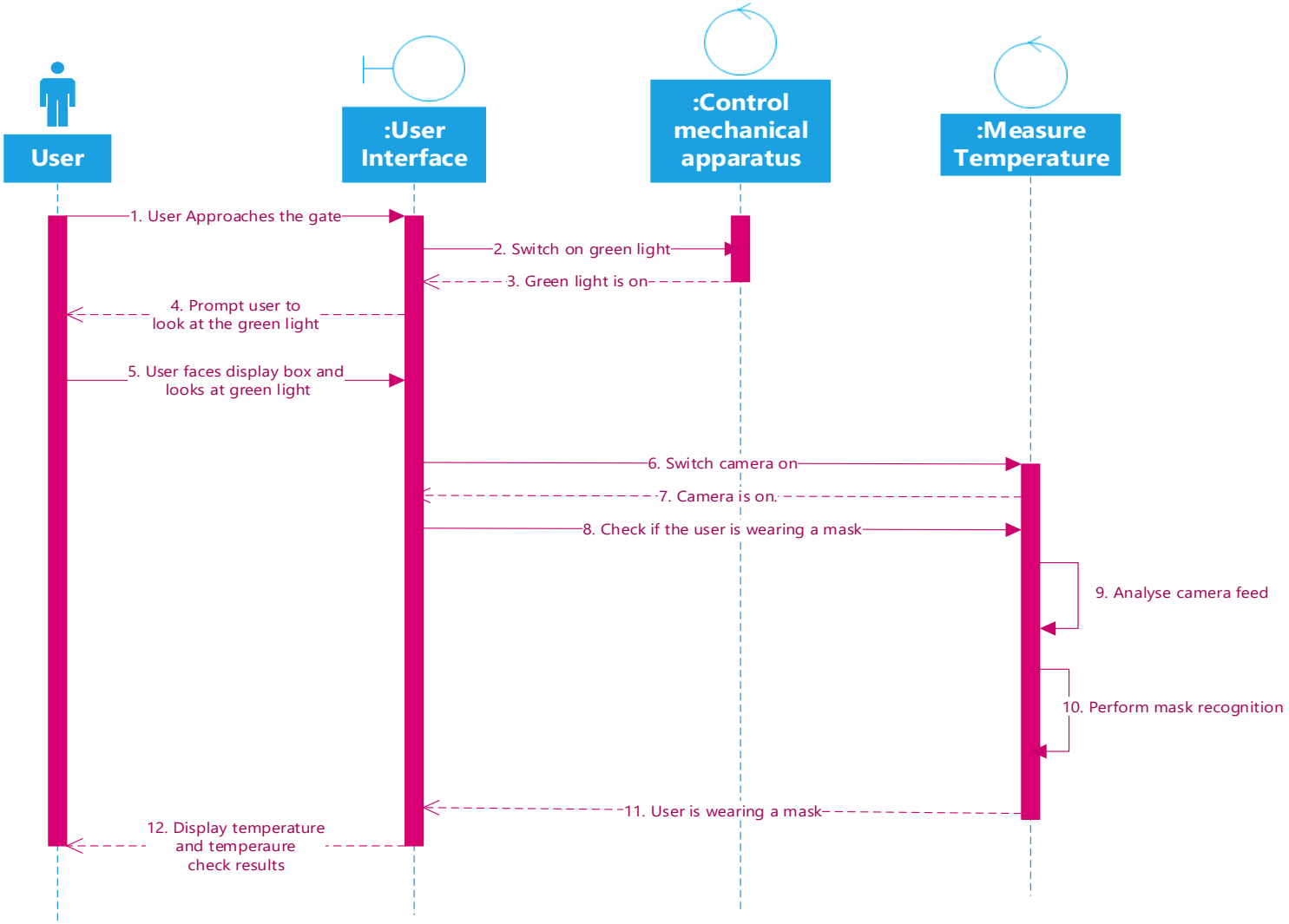
Measure Temperature

Figure 21 - Measure Temperature Sequence Diagram



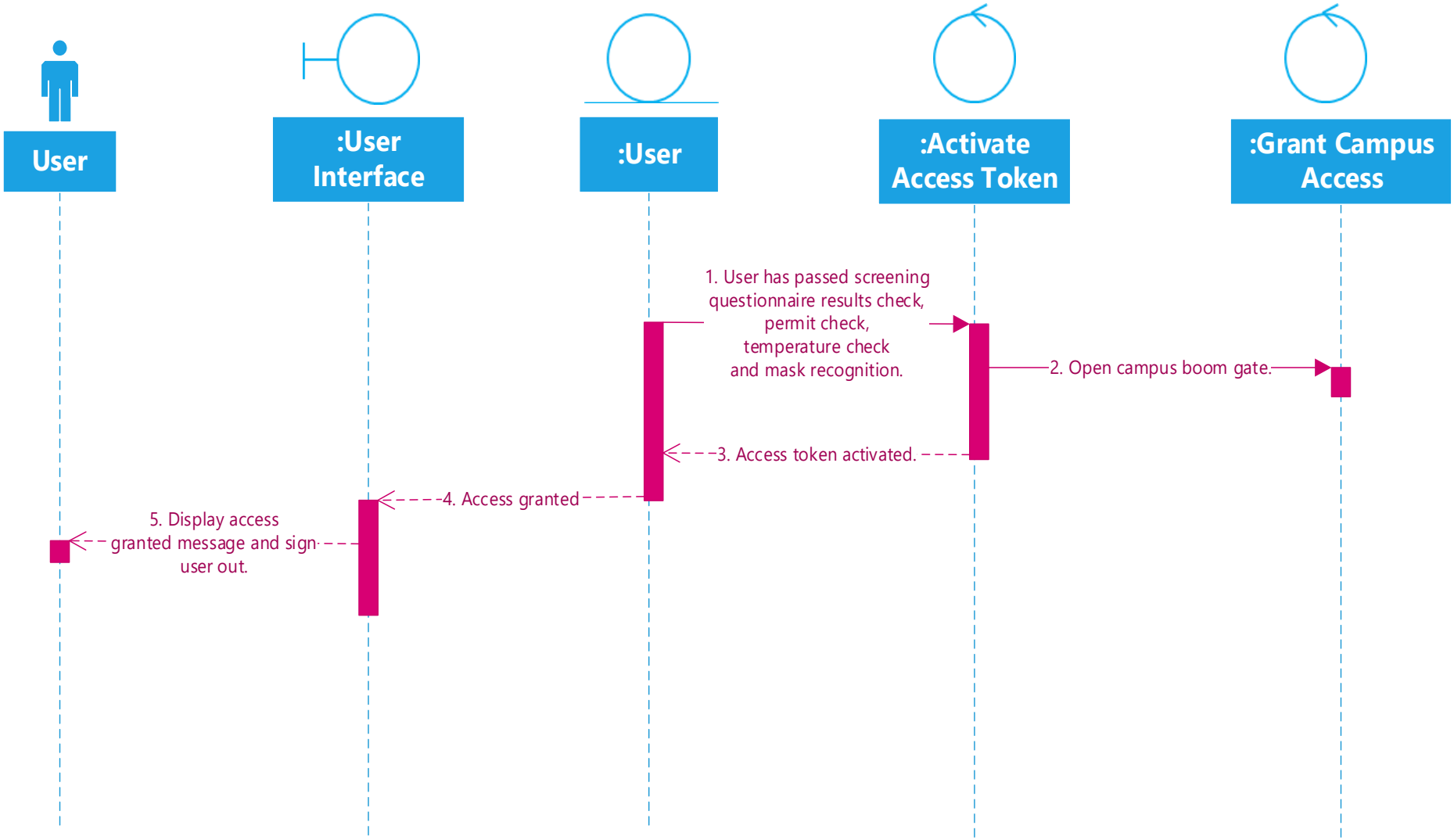
Mask recognition

Figure 22 - Mask Recognition Sequence Diagram



Grant Access to Campus (Access Token)

Figure 23 - Grant Access to Campus Sequence Diagram



Clear All Check Results

Figure 24 - Clear All Check Results Sequence Diagram