Secure Door Lock

Software Requirements Specification

Team Members

Luke Bucher	lbucher2017@my.fit.edu
Christopher Kiefer	ckiefer2019@my.fit.edu
James Pabisz	jpabisz2020@my.fit.edu
Warren Smith	wsmith2019@my.fit.edu

Faculty Advisor

Dr. Marius Silaghi
Graduate Professor at Florida Institute of Technology
msilaghi@fit.edu

Client

Dr. Marius Silaghi
Graduate Professor at Florida Institute of Technology
msilaghi@fit.edu

Notes: 3

1. Introduction 4
1.1 Purpose 4
1.2 Document Conventions 4
1.3 Product Scope 4

2. Overall Description 5
2.1 Product Perspective 5
2.2 Product Functions 5
2.3 User Classes and Characteristics 5
2.4 Operating Environment 5
2.4.1 Mobile Application 5
2.4.2 Server 6
2.4.3 Firmware 6
2.5 Design and Implementation Constraints 6
2.6 Assumptions and Dependencies 6

3. External Interface Requirements 6
3.1 User Interfaces 7
3.2 Hardware Interfaces 7
3.3 Software Interfaces 7
3.4 Communications Interfaces 8
3.4.1 Firmware 8
3.4.2 Server 8
3.4.3 Mobile Application 8

4. System Features 8
4.1 Mobile Application 8
4.1.1 Remote Unlock/Lock 8
4.1.2 Remote Video View 9
4.1.3 Door Lock Adoption 11
4.1.4 User Authentication 12
4.1.5 Door Lock Adoption 14
4.1.6 Recognized Visitors 15
4.2 Server 17
4.2.1 Authentication 17
4.2.2 IOT Persistent Endpoint 18
4.2.3 Device Database 19
4.2.4 Facial Recognition 19
4.3 Firmware 20
4.3.1 Unlock/Lock 21

4.3.2 Video Stream 21
4.3.3 IOT Adoption 23

5. Other Nonfunctional Requirements 24
5.1 Performance Requirements 24
5.2 Safety Requirements 24
5.3 Security Requirements 24
5.4 Software Quality Attributes 24

6. Other Requirements 25

Revision History

Name	Date	Reason For Changes	Version
Luke Bucher	09/14/2022	Initial Creation	1.0
Luke Bucher	10/01/2022	Additional Requirement Additions	1.1
Luke Bucher	10/04/2022	Presentation Revision	1.2

Notes:

- Meeting Notes 9/14

1. Introduction

1.1 Purpose

This document is intended to outline the goals and objectives of the Secure Door Lock project. This document will serve as a high level overview of the needs of each system and discuss important metrics that should be maintained for the project. The project intends on establishing a robust and secure method of locking an entryway through a multi-factor method of authentication. Through the companion mobile application users will be able to remotely interface with the lock through the internet and control the physical lock.

1.2 Document Conventions

The following table below outlines any continuous use acronym that the reader may come across within this document. First uses of any specific acronym will be accompanied by its reference number to correlate its position on the following table. Within the table the definition of the abbreviation will be provided as well as a short description of its usage within this document.

Reference #	Abbreviation	Definition	Description
1.2.1	IOT	Internet of Things	https://en.wikipedia.org/wiki/Internet_of_things

1.3 Product Scope

The software system will encompass the IOT (1.2.1) device being developed, the server infrastructure being developed, and the companion mobile application. References to the software system as a whole will be attributed to these three systems in coordination. For clarity and simplicity features and perspectives will be split amongst these three categories into subsystems.

These subsystems will be split as such: Firmware Side, Server Side, and Application Side. While further abstractions can be established within these subsystems any abstraction will be handled by the umbrella term.

2. Overall Description

2.1 Product Perspective

Our Secure Door lock system is a replacement for commercially available off the shelf solutions. We aim to increase the responsiveness of the system by generating a secure low level system that is tightly integrated with its corresponding app to give the user additional control not directly available from off the shelf products. The three layer system consists of the door locking hardware, our API service, and the mobile application (see figure 2.1.1).

By simplifying the integration process and increasing the responsiveness of the mobile application our system intends on strengthening the overall security of the average individual by eliminating the potential headaches that are involved with upgraded security systems. Primarily focusing on the responsiveness and ease of use from the mobile application is how our system is designed to account for these improvements.

2.2 Product Functions

Below are the current high level functions that the software system will meet.

- Status Monitoring
- Live Video Feed
- Remote Lock
- Facial Authentication

2.3 Operating Environment

2.3.1 Mobile Application

The mobile application will support mobile devices that operate on the android operating system. The assumption is that these mobile devices will include bluetooth capabilities, a network connection, and the ability to interact with the application through standard touch events.

2.3.2 Server

The main intermediary server will consist of an AWS instance that can facilitate communication from the mobile application to the proper IOT device. As well the server will act as the primary IOT host for any device that is connected to the Secure Door Lock network.

2.3.3 Firmware

The main door lock will integrate with a raspberry pi running Ubuntu Core on the device. This device will support the encoding for video feed from the integrated camera and handle interactions between the door solenoid and any other sensors that are connected to the Raspberry Pi.

2.4 Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

2.5 Assumptions and Dependencies

The largest assumption that we assume is that the end user has a network connected device that can connect to a device via bluetooth. Secondarily we assume the end user has a network connection located near the IOT device. We assume that the end user as well has access to a valid email address and can access the email for registration issues.

3. External Interface Requirements

3.1 User Interfaces

A common navigation bar will be present at all times at the bottom of the screen. As well A profile button will be accessible at the top of the screen for users to view their own profile. The bottom navigation buttons will include a main menu, device listing, and facial recognition profiles.

Further specifications can be found within the SDS document.

3.2 Hardware Interfaces

The device will connect to hardware components through the physical interface available on the Raspberry PI. Data will be translated through RTSPS to translate the image device into a JSON format for processing further upstream. Raw PWM data will be available for additional hardware interfaces,

3.3 Software Interfaces

Devices are connected over HTTP and receive connections via open API's on the devices to receive incoming requests via the Put and Get methodology. Currently devices will use a RESTful nature to produce data when requested and send data to an open endpoint. IOT devices not connected to a valid network connection will broadcast a Bluetooth connection that can be connected with the accompanying mobile application.

3.4 Communications Interfaces

3.4.1 Firmware

The camera module system transmits image data over serial connection directly with the implemented Raspberry PI installed in the physical lock. The raspberry pi will communicate with the local network in the area over Wi-Fi to the upstream IOT controller.

3.4.2 Server

The upstream IOT controller will use HTTP standards to facilitate the communication between the authentication database, the mobile application, and the secure door. These requests will be handled internally by the server and route appropriate communication to the correct sub-system. A RESTful architecture will be used to ensure that a responsive environment is promoted and allow for simultaneous and secure communication.

3.4.3 Mobile Application

Leveraging the RESTful nature of the primary server, the mobile application will communicate over HTTP. Using our REST server as an intermediary the app will open requests with the IOT device to facilitate the remote features that are available to the secure door lock.

4. System Features

System features will be separated by their accompanying sub-systems. Each feature will focus on the feature exclusively within that sub-system and if needed will have a data stream out to another sub-system.

4.1 Mobile Application

4.1.1 Remote Unlock/Lock

4.1.1.1 Description and Priority

The application will be able to submit a request to a specified lock that is associated with the authenticated user to be able to lock/unlock the door lock from within the mobile application. This is the highest priority functionality that must operate.

4.1.1.2 Stimulus/Response Sequences

User Selects the door lock from their adopted list

User is prompted with a button to unlock/lock the door or view the live video feed

If User selects the unlock/lock button, a request packet is generated and sent to the listening service

On acknowledgement of the request packet the user is notified that the action has been sent

If the action was completed successfully the user is notified that the door status has been updated to locked/unlocked

If the action was not successful the user is notified that an error has occurred and any additional information that can be determined is sent to them.

4.1.1.3 Itemized Requirements

REQ-1	The system shall display an unlock/lock button when a door lock is selected.
REQ-2	The system shall send a request packet to the listening service to communicate with the locking device

REQ-3	The system shall notify the user that the request has been sent and acknowledged
REQ-4	The system shall notify the user of errors that occur on failure of acknowledgement

4.1.2 Remote Video View

4.1.2.1 Description and Priority

The application will be able to submit a request to a specified lock that is associated with the authenticated user to be able to view the camera feed that is available on the lock. This is a secondary priority to the other features available.

4.1.2.2 Stimulus/Response Sequences

User Selects the door lock from their adopted list

User is prompted with a button to unlock/lock the door or view the live video feed

If the User selects the view live video feed the user is transitioned to a new window

A request packet is sent to the listening server to request a video feed.

If the camera view is available the window will display the current view that is seen by the door lock

If the camera view is unavailable the window will display a feed not available notification.

4.1.2.3 Itemized Requirements

REQ-5	The system shall display a view live feed button.
REQ-6	The system shall send a request to the listening service to communicate with the locking device to establish a video feed.
REQ-7	The system shall contain a separate window for the viewing of the camera feed.

REQ-8	The system shall automatically transition the user to the new window for viewing
REQ-9	The system shall display a live feed from the selected camera
REQ-10	The system shall display a “feed unavailable” notification if the feed cannot be obtained
REQ-11	The system shall maintain a minimum framerate of the live feed of 1 frame per second

4.1.3 Door Lock Adoption

4.1.3.1 Description and Priority

The application will display a list of currently adopted devices that are available to the current authenticated user. On the list display the user will be able to add additional devices. This is a tertiary priority and will be implemented if time is available.

4.1.3.2 Stimulus/Response Sequences

The authenticated user selected the “My Devices” window

The user is transitioned to the “My Devices” window

The new window lists all connected devices to the account and displays relevant information about the door lock.

The user selected the “Add Device” button within the window

The user is prompted to add the device to their list through identification or bluetooth

The user is notified that the device has been added to their listing

If a failure of addition occurs the user is notified that the device has not been added to their listing, providing and possible errors

4.1.3.3 Itemized Requirements

REQ-12	The system shall display a “My Devices” window
REQ-13	The system shall list all devices that are connected to the authenticated user account
REQ-14	The system shall display an “Add Device” button within the “My Devices” window
REQ-15	The system shall display the current status of each connected device: Lock Status, Door Open Status, Camera Feed Status, Battery Status
REQ-16	The system shall display an error on the list entry if it cannot communicate with the device
REQ-17	The system shall display an “Add Device” window when the “Add Device” button is clicked
REQ-18	The system shall display two options for adding a device to the authenticated users device listing
REQ-19	The system shall notify the user when a devices is added successfully to their device listing
REQ-20	The system shall notify the user when a device is unsuccessful when attempting to add to the listing

4.1.4 User Authentication

4.1.4.1 Description and Priority

The application will be able to ingest a username and password from the user and attempt to validate the credentials and login to the specified user account. The application will be able to reset passwords through a forgotten password section. On completion of the login process the user will be transferred to their device listing of their adopted devices. The user will also be able to select a “Remember Me” function and automatically login when the app is opened from the same device.

4.1.4.2 Stimulus/Response Sequences

User opens the mobile application for the first time on their device.

User is transitioned to the login screen.

User is prompted for a Username and Password

The system will also display “Forgot Password?” and “Sign Up” buttons below the credential fields

The user enters their credentials into the field

The user selects the “Login” button

The credentials are sent to the listening service to be authenticated

The user is notified that their credentials are accepted if the credentials are correct

If the credentials are not correct the user is notified that the authentication failed

The user selects the “Forgot Password” button

An email is sent to the registered email address associated with the username with a temporary replacement password

The user selects the “Sign Up” button

The user is transferred to a Sign up window

The user is prompted to enter a email address and username

An email is sent to the entered email address with a temporary password

If the user logs in with a temporary password the user is requested to update their password

If the user completes a log in with their normal credentials they are transitioned to the home page of the application.

If a User has selected the “Remember Me” option upon opening the application the user will be authenticated automatically and transitioned to their home page

4.1.4.3 Itemized Requirements

REQ-21	The system shall display a window for logging into the application that contains: Credential Fields (User Name, Password), Buttons (Login, Forgot Password, Sign Up), and a Checkbox (Remember Me)
REQ-22	The system shall notify the user on successful login to their account and transition them to their home page
REQ-23	The system shall notify the user on unsuccessful login attempt
REQ-24	The system shall allow the user to reset their password by sending a “Request Password Change” to the listening service
REQ-25	The system shall allow the user to sign up by sending a “New User” to the listening service
REQ-26	The system will display a secondary window that will prompt for a username and email from the user

4.1.5 Recognized Visitors

4.1.5.1 Description and Priority

Recognized visitors are determined by the user registry based on stored facial recognition. Authenticated users will be able to view stored users and the last recognized name and photo of them. Users will be able to add dynamic names to the profile as well as customize the photo for the profile. Users will be able to remove and combine profile data to increase the accuracy of the recognition system. This requirement maintains the lowest priority among the mobile application requirements and can be seen as a further extension of usability.

4.1.5.2 Stimulus/Response Sequences

Face is recognized at the door Authenticated user is notified via push notification: Push notification contains the potential recognized face profile and a last seen image. User is prompted for an approve or deny within the mobile application If approved device commences the Remote Unlock process Else A Deny request is sent to the server
A list of recognized profiles is displayed A user selects a profile Within the profile the option to change customize the profiles name and image is presented User selects a new name and profile and is updated within the registry A user selects another profile The option to combine or delete profiles is presented A user selects delete profile The profile is deleted from the registry A user selects combine profile The selected profile is removed and underlying data is combined with the selected

combination.

4.1.5.3 Itemized Requirements

REQ-55	The system shall display a unique window for “Recognized Visitors”
REQ-56	The system shall display facial profile data related to the authenticated user
REQ-57	The system shall display buttons to delete or combine profiles
REQ-58	The system shall allow for customization for facial profile data: Names and Profile images can be customized per the users request
REQ-59	The system shall push a notification to the authenticated user from a server request
REQ-60	The system shall contain an approved or denied request within the push notification to begin or stop the remote Lock/Unlock process
REQ-61	The system shall display an image of the detected face within the mobile application and the push notification

4.2 Server

4.2.1 Authentication

4.2.1.1 Description and Priority

The server will maintain a database of user information as well as accept requests for login attempts from devices and verify against stored information. As well the system will allow for user account creation and reset password credentials on request. The priority of this falls below the IOT Persistent Endpoint however it is required to verify accounts.

4.2.1.2 Stimulus/Response Sequences

4.2.1.3 Itemized Requirements

REQ-27	The system shall accept login requests from the mobile application and verify against the user database
REQ-28	The system shall store user information within a secure and protected database
REQ-29	The system shall accept forgotten password requests and send validation emails to the requesting user account
REQ-30	The system shall generate new user accounts from requests within the mobile application

4.2.2 IOT Persistent Endpoint

4.2.2.1 Description and Priority

The system shall maintain a persistent connection between the IOT devices and authenticated user accounts. The system will accept video stream and sensor notifications from the IOT device and notify the authenticated user. These requirements shall maintain the highest level of priority within our sub-system.

4.2.2.2 Stimulus/Response Sequences

4.2.2.3 Itemized Requirements

REQ-31	The system shall accept video data directly from the IOT device
REQ-32	The system shall accept sensor data from the IOT device regarding: Door Open Status, Lock Status, and Battery Status
REQ-33	The system shall stream video data to an authenticated user
REQ-34	The system shall notify authenticated users of relevant sensor data from the IOT device

4.2.3 Device Database

4.2.3.1 Description and Priority

The server shall maintain a database of connected devices and authenticated users that have the required access to them. The system shall allow for registration of new devices to authenticated users and approved secondary owners. This is the lowest priority system that can be delayed based on time needs of other requirements.

4.2.3.2 Stimulus/Response Sequences

4.2.3.3 Itemized Requirements

REQ-35	The system shall maintain a database of registered IOT devices as well as user ownership
REQ-36	The system shall accept registration of new IOT devices and attach new owners from authenticated users

4.2.4 Facial Recognition

4.2.4.1 Description and Priority

The server shall receive incoming images from a registered IOT device. The server will scan images and recognize potential faces. Once a face is recognized a registry of faces will be assigned to the owner account of the registered IOT device. Faces within the registry will be able to be assigned names. This is a secondary requirement that is needed to be completed on project completion.

4.2.4.2 Stimulus/Response Sequences

Image is received from the Persistent Endpoint.

Image is scanned for potential faces within the image

Face is detected within the image.

Facial data is analyzed and compared with user registry

If similar data is found relevant named Face data is pushed to the registered account

If no data is found or if an error occurred an unknown face notification is pushed to the registered account

4.2.4.3 Itemized Requirements

REQ-37	The system shall scan image data from the IOT device for possible faces
REQ-38	The system shall maintain a registry of facial data
REQ-39	The system shall add facial data as a new entry when no data is currently present
REQ-40	The system shall allow the registered account to add data to existing entries

REQ-41	The system shall push a notification to the registered user(s) that a face has been detected at the door: A named entity if found in the registry or an unknown visitor if not found.

4.3 Firmware

4.3.1 Unlock/Lock

4.3.1.1 Description and Priority

As this is the primary focus of the door lock to unlock the IOT device from a network request this requirement maintains the highest priority. The IOT device will receive a request from the persistent endpoint to unlock the door. Upon receiving the request the raspberry pi controller will communicate with the hardware servos to complete the unlocking process. As well the system will communicate upstream updates with the hardware sensors about current activity within the system.

4.3.1.2 Stimulus/Response Sequences

Device receives request from endpoint

Signals are sent to hardware components to unlock the IOT device

Updates are received from hardware sensors to raspberry pi controller

Notifications are pushed upstream to persistent endpoint

4.3.1.3 Itemized Requirements

REQ-42	The system shall receive requests from a persistent endpoint to communicate with hardware servos
REQ-43	The system shall send requests related to hardware sensors to a connected persistent endpoint

REQ-44	The system shall communicate with hardware components using direct connection via the local raspberry pi controller

4.3.2 Video Stream

4.3.2.1 Description and Priority

The IOT device will accept image data from the onboard camera. Image data shall be sent upstream to the connected persistent endpoint. To avoid over use of the on board battery a dynamic poll rate of the video feed will be used. A minimum and maximum poll rate of the camera module will be set and will be adjusted by detected disturbances within the image data. After a cooldown period of inactivity, the poll rate is adjusted again back to the minimum poll rate. This poll rate can also be adjusted from view feed requests from the persistent endpoint. This priority is secondary to the other local functionality located on the IOT device.

4.3.2.2 Stimulus/Response Sequences

Device polls from hardware camera module for image data Activity is detected within the image data Poll rate is increased to stream additional frames upstream to the endpoint Cooldown begins After cooldown activity is no longer detected and minimum poll rate is set again
Device receives view feed request Poll rate is increased User exits view feed Device receives notification that view feed has completed Cooldown begins After cooldown poll rate is brought down again

4.3.2.3 Itemized Requirements

REQ-45	The system shall receive image data from the hardware camera via the raspberry pi controller
REQ-46	The system shall send image data upstream to a persistent endpoint
REQ-47	The system shall receive view feed requests
REQ-48	The system shall dynamically adjust polling rate of image data based upon: Activity Detection or Request “View Feed”. A minimum and maximum poll rate will be established as well as a cooldown rate for making rate adjustments

4.3.3 IOT Adoption

4.3.1.1 Description and Priority

The IOT device requires an owner in order to properly complete the listed requirements. In order to complete the adoption process a user will be required to begin the initial set up process of the device. The device will need to be accessed via a bluetooth connection in order to establish ownership of the device as well as connect the device to a network connection. In instances where the device cannot connect to a valid network the device will broadcast a bluetooth connection that allows the device to be configured with the appropriate network information.

4.3.1.2 Stimulus/Response Sequences

Device is powered on
Attempt to connect to a local network
No network is found or no connection can be made
Bluetooth broadcast begins
Connection from mobile application is authenticated
Configuration is received from application

Network connection is established Registration is communicated via configuration and registered with persistent endpoint

4.3.1.3 Itemized Requirements

REQ-62	The system will detect a valid connection to a local network
REQ-63	The system shall broadcast a bluetooth signal when a connection to a local network cannot be made
REQ-64	The system shall accept configuration files via bluetooth from a connected mobile application
REQ-65	The system shall maintain an unique identifier that is stored internally based upon unique hardware identifiers
REQ-66	The system shall store a registered owner locally

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Performance is a paramount consideration when it comes to IOT devices in coordination with a mobile application. Minimizing the time to action between user actions and server response is key to building an user friendly experience.

While our initial goal is to maintain functionality and security ensuring that an action can respond within 7-10 seconds.

5.2 Security Requirements

Our system will maintain a secure environment to not allow outside intrusion via user authentication and token authentication. Ensuring a secure environment where only authorized access is allowed should ensure this sentiment.

Users are only allowed to add “discoverable” devices that have not been adopted by other accounts. Devices that are adopted by an authenticated user are not able to be accessed by other accounts. Ownership can only be transferred via the original owner.

5.3 Software Quality Attributes

Since we intend to market our application to the general public we must ensure that the application minimizes complexity. Ensuring that applications maintain a maximum of 6 actions to complete will ensure that the standard user does not get confused during their usage of the app.

Ensuring as many actions are clearly labeled or context driven should ensure that users can follow any action pattern. User interactions should be limited to an unique window and any action regarding the unique window should be limited to the unique window.