



Requirements Document

Portcullis

November 26th, 2025

General Dynamics Mission Systems

Client: Benjamin Walker

Faculty Mentor: Bailey Hall

Team members: Christopher Son, Aiden Seay, Preston Smith, & Ryan Todd

Overview

The purpose of this document is to introduce the project and problems faced by the client and the requirements that the client has defined for a successful product. We will look at key aspects of the requirements and build out the team's solution vision, along with analyzing the functional and performance requirements.

Accepted as the baseline requirements for the project.

Client:

Team: *Christopher Son*

1.0 Introduction	3
2.0 Problem Statement	4
3.0 Solution Vision	5
4.0 Project Requirements	6
4.1 Domains	6
4.2 Functional Requirements	8
4.2.1 Must-Have (MHx) Functional Requirements	8
4.2.2 Should-Have (SHx) Functional Requirements	14
4.3 Performance Requirements	18
4.4 Environmental Requirements	19
5.0 Potential Risks	21
6.0 Project Plan	22
7.0 Conclusion	25

1.0 Introduction

The defense industry plays a critical role in supporting national security, providing advanced and secure communication, surveillance, and access control systems to government and military organizations. These systems must perform reliably in remote and harsh environments..

General Dynamics Mission Systems (GDMS) is a leading defense manufacturer that builds and delivers critical mission systems to defense, government, intelligence, & cybersecurity clients. As part of this broader mission, GDMS develops secure, resilient technologies that support federal agencies, including the U.S. Coast Guard. In our project, GDMS is developing a locking solution for the Coast Guard for remote sites.

The Coast Guard currently uses Commercial off-the-shelf (COTS) products (latches, software, etc.) to secure sensitive sites. While these products can secure these sites, there are multiple shortcomings that COTS products have, which make them less than optimal for securing Coast Guard installations. Chiefly, these systems must be able to operate without power safely and securely, while remaining up to date with modern security standards. These limitations create operational and logistical challenges for the Coast Guard, motivating GDMS to seek an in-house solution that provides full design control, enhanced security, and lower long-term maintenance costs.

2.0 Problem Statement

To understand the issues that GDMS faces with the current arrangement, we need to understand the current flow of accessing secured locations.

1. Personnel present an authentication token to the HID (an access control company) key reader.
2. The card reader reads data from the token and transmits the data to an access control panel.
3. The Access Control panel analyzes the data and determines whether the person requesting access is authorized to enter.
4. Access is granted or denied.

This current solution is fairly simple, where a preprogrammed card is issued to authorized site personnel to access specific locations. In this system, the Access Control Panel conducts the business logic, which decides whether the card, not the person, is authorized to access the location. This simplicity leads to several pitfalls in the system, which may be exacerbated by the dynamic environments of Coast Guard facilities.

Some of these pitfalls are:

- Loss of connection from the main server of Access Control Panel
- Authorized person loses access token
- Card reader loses connection to Access Control Panel
- An unauthorized person steals and uses an access token
- Simple access control logs

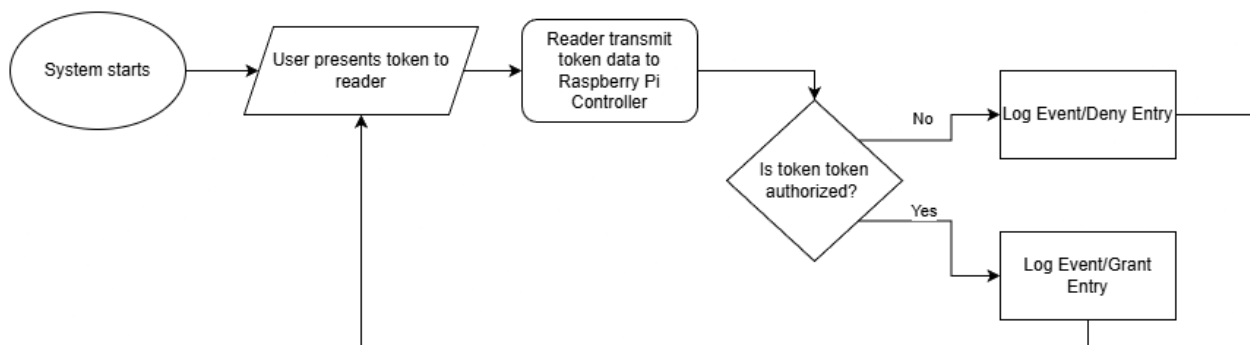
While this list is not exhaustive, many of these problems are prevalent in the current system, where, in chaotic settings, this can cause a site to become insecure or unresponsive. The Portcullis team envisions a new solution that can cover much of this and all of the Coast Guard's requirements. This new solution will be durable, responsive, redundant, and cost-effective.

3.0 Solution Vision

Our new solution takes advantage of new technologies and methodologies by implementing up-to-date protocols, security standards, event management and logging, data storage and retrieval, utilizing cost-effective hardware, and upgrading the administrator tools to be web-based. The following list describes the components.

- **Raspberry Pi:** Robust controllers, secure data communication, offline operability
- **Token Reader:** Use new protocols, allows for NFC authentication, and offline operability
- **Administrator Terminal/Application:** Tolerates high latency environments, allows for real-time notifications, digests and analyzes data from controllers
- **Database:** Archives data logs from controllers, allows for detailed logs of events, keeps data persistent, and acts as a single source of truth

We intend to keep the simplicity of the current authentication system for the user, making sure not to exceed four steps for allowing users to enter an area or not. The simplicity is illustrated below.



4.0 Project Requirements

The Coast Guard has been proactive and set out 15 requirements that Portcullis must meet for the new solution to be considered a success. In this section we will analyze and sort each requirement into their respective fields. Functional requirements, which are requirements that determine what the system does, and performance requirements, which specify how well the system or aspects should run. Environmental requirements are constraints which come from external entities.

4.1 Domains

The requirements from the Coast Guard can be divided into several domains, which in conjunction with each other, can help Portcullis design an up-to-date solution that keeps shelters safe, alert personnel of events, save data for auditing purposes, and enhance robustness.

- **D1 - Reliable Intrusion Detection and Alerting:** The system must detect intrusion attempts and provide immediate, reliable alerts to operators, even in degraded network environments.
- **D2 - Secure and Authorized Access Control:** Only authorized personnel should be able to access protected shelters. Unauthorized access attempts must be detected, logged, and reported.
- **D3 - Robust Operation Under High Latency:** Coast Guard installations often experience poor, high-latency network conditions. The system must remain fully functional—even at 400 ms latency.
- **D4 - Real-Time System Status Awareness:** Administrators require an accurate, real-time view of shelter access status and system health.
- **D5 - Persistent Event Logging and Auditability:** All access events, intrusion events, and system state changes must be logged and preserved for long-term compliance and auditing.

- **D6 - Secure, Modern, Encrypted Communication:** All system components must communicate using secure, industry-standard cryptographic protocols.
- **D7 - Recoverability and Fault Tolerance:** The system must automatically recover from outages and should avoid unrecoverable states during updates or communication failures.

These domains are fulfilled by the following functional (FRx), performance (PRx), and environmental (ERx) requirements.

Functional Requirements

Functional requirements will also be further classified using the MoSCoW prioritization method. The categories are Must-Have (MHx), Should-Have (SHx), Could-Have (CHx), and Won't-Have (WHx) where the smaller number marks the more important requirement of the classified requirements.

- **FR1(MH1)** - The system shall notify operators with a cautionary alert when there is a change in the state of an intrusion detector.
- **FR2 (MH2)** - The system shall notify operators with a cautionary alert when actions to enter controlled areas are detected.
- **FR3 (MH3)** - The status panel shall indicate access status to a shelter.
- **FR4 (MH4)** - The intrusion detector shall support three states: no access, authorized access, and unauthorized access.
- **FR5 (SH1)** - The system shall report authorized and unauthorized entries to shelters to the Control Center via an IP interface.
- **FR6 (MH5)** - The system shall use protected mechanisms (e.g., passwords) to authenticate the identity of system operators and administrators.
- **FR7 (SH2)** - The system shall protect against internal and external unauthorized access.
- **FR8 (SH3)** - Intrusion detectors shall monitor and automatically report status changes to the control center.
- **FR9 (SH4)** - The system shall support SNMPv3 for securely reporting intrusion status and system status.
- **FR10 (SH5)** - The system shall use current industry-standard encryption protocols (e.g., TLS, HTTPS) for secure communication between all parts of the system.

- **FR11 (SH6)** - The system shall automatically recover from network outages and resume normal operations without manual intervention.
- **FR12 (SH7)** - The system shall keep a history of intrusion and system events/alarms for auditing and analysis purposes.

Performance Requirements

- **PR1** - The system shall maintain functionality and reliable communication over high-latency (400 ms) network connections.
- **PR2** - The system shall perform software and firmware upgrades manually initiated and able to complete over high-latency (400 ms) network connections; in case of outage, the system must remain recoverable.

Environmental Requirements

- **ER1** - Fixed facilities shall incorporate intrusion detection capabilities.

4.2 Functional Requirements

Functional requirements are necessities put forth by the client which provide a basis for system behavior. The functional requirements that are needed covers many domains of the project necessitating system action, data structure, system security, system states, and so on. We will list all the requirements, their description, related domain, a user story, rationale, verification method, and a prototype picture for the Must-Haves.

4.2.1 Must-Have (MHx) Functional Requirements

FR1 (MH1)

Description: The system shall notify operators with a cautionary alert when there is a change in the state of an intrusion detector.

Related Domain: D1

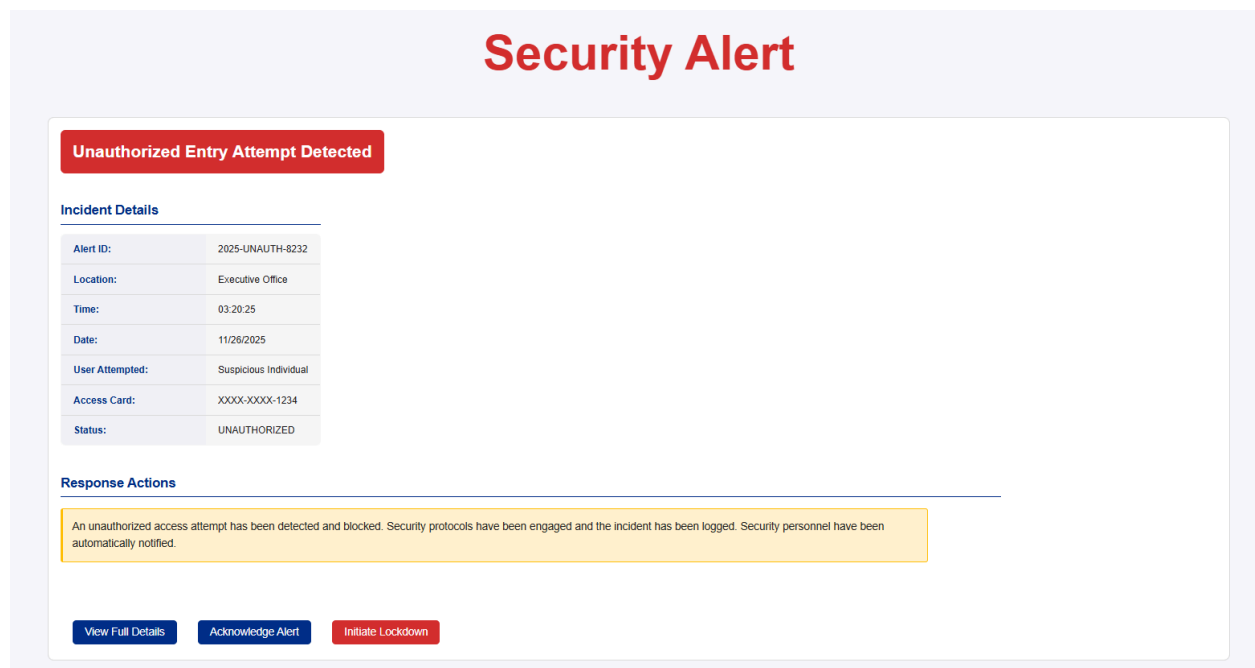
User Story:

“As an operator, I want to be alerted when an intrusion detector changes state so that I can quickly respond to potential security events.”

Rationale: Timely alerts ensure security events are handled quickly.

Verification Method: Trigger a simulated intrusion event and confirm that an alert is generated.

Prototype:



The prototype displays a 'Security Alert' interface. At the top, a red banner reads 'Unauthorized Entry Attempt Detected'. Below this, a section titled 'Incident Details' contains a table with the following information:

Alert ID:	2025-UNAUTH-8232
Location:	Executive Office
Time:	03:20:25
Date:	11/26/2025
User Attempted:	Suspicious Individual
Access Card:	XXXX-XXXX-1234
Status:	UNAUTHORIZED

Below the table, a section titled 'Response Actions' contains a yellow box with the text: 'An unauthorized access attempt has been detected and blocked. Security protocols have been engaged and the incident has been logged. Security personnel have been automatically notified.'

At the bottom, there are three buttons: 'View Full Details' (blue), 'Acknowledge Alert' (blue), and 'Initiate Lockdown' (red).

FR2 (MH2)

Description: The system shall notify operators with a cautionary alert when actions to enter controlled areas are detected.

Related Domains: D1, D2

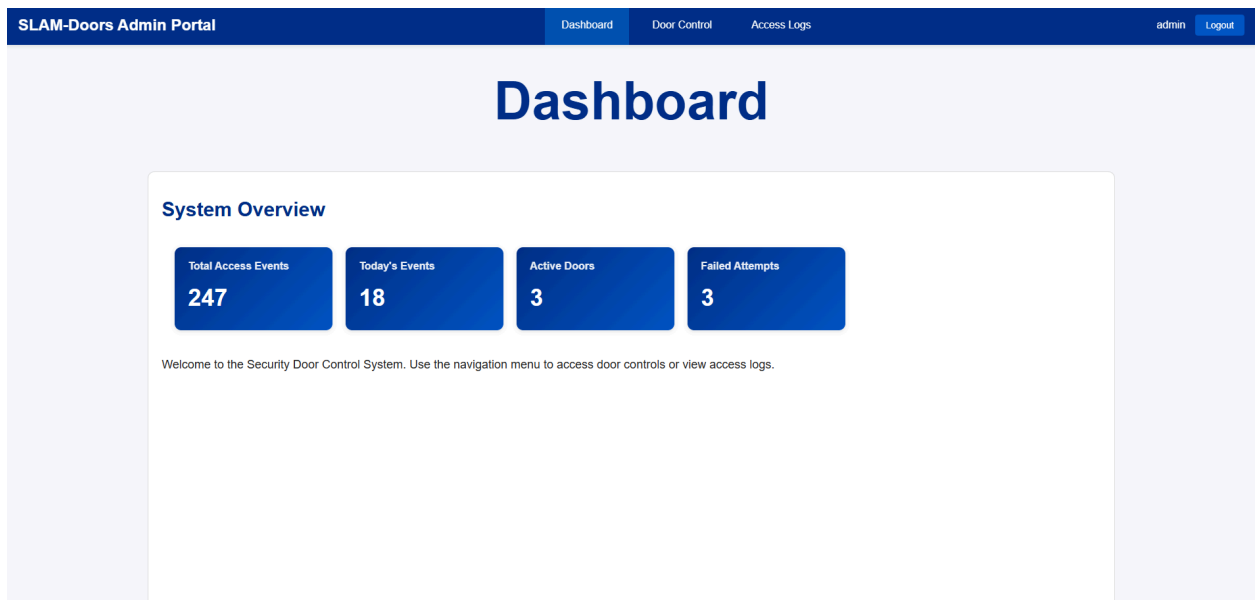
User Story:

“As an operator, I want to receive alerts when someone tries to access a controlled area so that I can monitor potentially unauthorized activity.”

Rationale: Detecting physical manipulation or entry attempts is essential to securing remote sites.

Verification Method: Simulate authorized and unauthorized access attempts and verify alert generation.

Prototype:



Access Logs

Door Access History

Filter by Door: Filter by Status:

Timestamp	Door	User	Action	Status
2025-11-06 13:45:22	Main Entrance	Unknown	Access	Unauthorized
2025-11-06 09:05:33	Main Entrance	Tom Brown	Access	Unauthorized

FR3 (MH3)

Description: The status panel shall indicate access status to a shelter.

Related Domains: D4

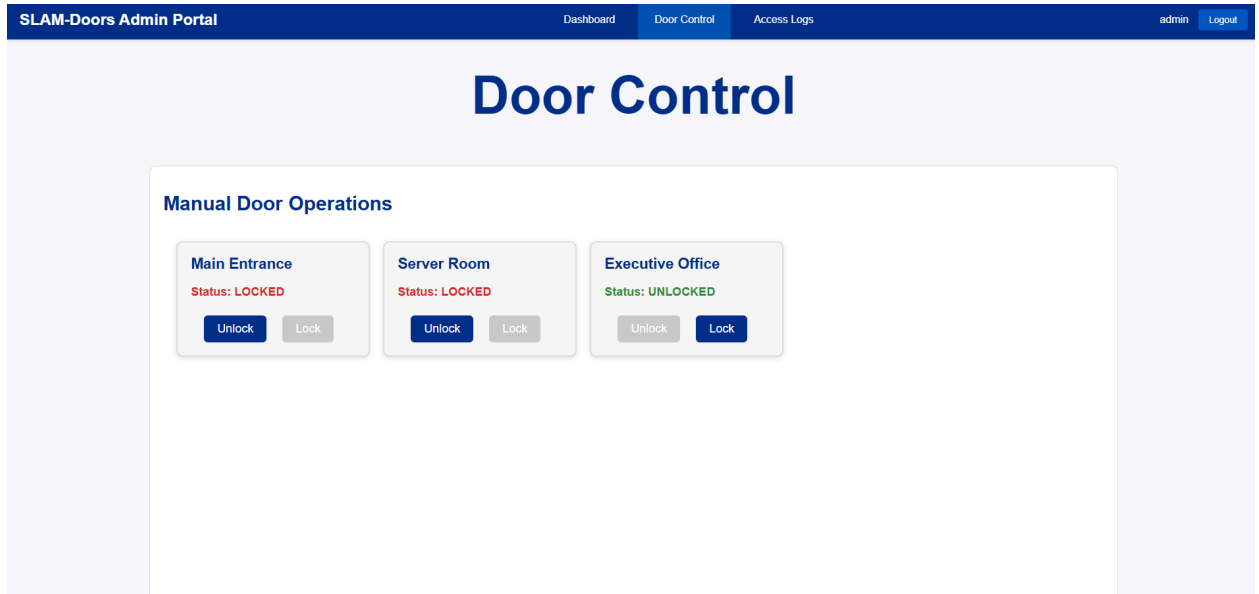
User Story:

“As an administrator, I want to view the access status of all shelters so that I can maintain situational awareness across multiple facilities.”

Rationale: A visual UI for system state simplifies decision-making.

Verification Method: When test data is sent to the database, changes in door status should be reflected in the Admin Terminal

Prototype:



FR4 (MH4)

Description: The intrusion detector shall support three states: no access, authorized access, and unauthorized access.

Related Domains: D1, D2

User Story:

“As an administrator, I want the system to classify access events into three distinct states so that I can interpret activity logs accurately and consistently.”

Rationale: The three log states allow operators and administrators to quickly distinguish normal access events from intrusion events.

Verification Method: Simulate sending all three state event logs to the backend database. The database should accept logs and the administrator application should show those events.

Prototype:

Access Logs

Door Access History

Filter by Door: Filter by Status:

Timestamp	Door	User	Action	Status
2025-11-06 14:23:45	Main Entrance	John Smith	Access	Authorized
2025-11-06 14:15:32	Server Room	Admin Override	Unlock	No Access
2025-11-06 13:58:12	Executive Office	Jane Doe	Access	No Access
2025-11-06 13:45:22	Main Entrance	Unknown	Access	Unauthorized
2025-11-06 12:30:15	Server Room	Mike Johnson	Access	Authorized
2025-11-06 11:22:08	Main Entrance	Sarah Williams	Access	Authorized
2025-11-06 10:15:44	Executive Office	Admin Override	Lock	Authorized
2025-11-06 09:05:33	Main Entrance	Tom Brown	Access	Unauthorized
2025-11-06 08:45:21	Server Room	Alice Cooper	Access	Authorized
2025-11-06 08:12:10	Main Entrance	Bob Taylor	Access	Authorized

FR6 (MH5)

Description: The system shall use protected mechanisms (e.g., passwords) to authenticate the identity of system operators and administrators.

Related Domains: D2, D6

User Story:

“As an administrator, I want secure login mechanisms with protected credentials so that unauthorized users cannot access administrative functions.”

Rationale: Prevents attackers from accessing the system and disallows misuse.

Verification Method: Attempt to log in with correct and incorrect credentials.

Prototype:

SLAM-Doors Admin Login

Username

Password

Login

Demo Credentials:

Username: admin

Password: secure123

4.2.2 Should-Have (SHx) Functional Requirements

FR5 (SH1)

Description: The system shall report authorized and unauthorized entries to shelters to the Control Center via an IP interface.

Related Domains: D1, D2, D5, D6

User Story:

“As an administrator, I want to make sure that my system’s communication is done in a secure manner, ensuring that all alerts are authentic.”

Rationale: Ensures all communications occur with no Wi-Fi transmission.

Verification: Monitor messages transmitted over IP connections during simulated entries.

FR7 (SH2)

Description: The system shall protect against internal and external unauthorized access.

Related Domains: D2

User Story:

“As a facility operator, I want unauthorized individuals, internal or external, to be denied access so that shelters remain secure.”

Rationale: Ensures that unauthorized actors are unable to access secured shelters and authorized users can only access doors they are allowed to enter.

Verification: Test valid and invalid credentials on multiple access points.

FR8 (SH3)

Description: The system shall use current industry-standard encryption protocols (e.g., TLS, HTTPS) for secure communication between all parts of the system.

Related Domains: D6

User Story:

“As a system architect, I want all system communications encrypted so that we reduce risks of interception or man-in-the-middle attacks.”

Rationale: TLS and HTTPS allow confidentiality and integrity of communications.

Verification: Capture network traffic with AWS network services and verify encryption usage.

FR9 (SH4)

Description: The system shall support SNMPv3 for securely reporting intrusion status and system status.

Related Domains: D6

User Story:

“As a network administrator, I want intrusion detectors to use SNMPv3 so that system status reports remain secure during transmission.”

Rationale: SNMPv3 provides encryption, authentication, and access control.

Verification: Validate that all SNMP communication utilizes SNMPv3 and nothing else.

FR10 (SH5)

Description: Intrusion detectors shall monitor and automatically report status changes to the control center.

Related Domains: D1, D4

User Story:

“As a control unit, I want to automatically report intrusion state changes so that operators receive information reliably even without human oversight.”

Rationale: Automation quickens the responsiveness of the program and alerts administrators quicker.

Verification Method: Send test events and verify real-time automated reporting.

FR11 (SH6)

Description: The system shall automatically recover from network outages and resume normal operations without manual intervention.

Related Domains: D3, D7

User Story:

“As a technician, I want to reconnect to the backend automatically after a network outage so that the system can restore normal function without technician intervention.”

Rationale: Connecting back to the system without technician need increases system robustness and reduces the need for having a technician to reconnect every unit.

Verification Method: Disconnect and reconnect the control unit and verify that the unit reconnects automatically.

FR12 (SH7)

Description: The system shall keep a history of intrusion and system events/alarms for auditing and analysis purposes.

Related Domains: D5

User Story:

“As an auditor, I want the system to record all access and intrusion events so that I can review historical activity for compliance and analysis.”

Rationale: Storing logs in a database allows logs to be persistent and used for auditing and analysis and meets organizational requirements.

Prototype:

Access Logs

Door Access History

Filter by Door: Filter by Status:

Timestamp	Door	User	Action	Status
2025-11-06 14:23:45	Main Entrance	John Smith	Access	Authorized
2025-11-06 14:15:32	Server Room	Admin Override	Unlock	No Access
2025-11-06 13:58:12	Executive Office	Jane Doe	Access	No Access
2025-11-06 13:45:22	Main Entrance	Unknown	Access	Unauthorized
2025-11-06 12:30:15	Server Room	Mike Johnson	Access	Authorized
2025-11-06 11:22:08	Main Entrance	Sarah Williams	Access	Authorized
2025-11-06 10:15:44	Executive Office	Admin Override	Lock	Authorized
2025-11-06 09:05:33	Main Entrance	Tom Brown	Access	Unauthorized
2025-11-06 08:45:21	Server Room	Alice Cooper	Access	Authorized
2025-11-06 08:12:10	Main Entrance	Bob Taylor	Access	Authorized

There are currently no identified Could-Have and Won't-Have requirements at this time, so the rest of the MoSCoW requirements will not be stated here.

4.3 Performance Requirements

Performance requirements are requirements that measure specific aspects of the system. These requirements are by their nature testable and can be quantified to determine if the system meets requirements or not. In this section, we will review the performance requirements that the Coast Guard has defined and determine which domain they belong to and which aspect needs to be measured to ensure success.

PR1

Description: The system shall maintain functionality and reliable communication over high-latency (400 ms) network connections.

Related Domains: D3

User Story:

“As an administrator, I want the system to function reliably even when network connections are extremely slow so that Coast Guard sites with weak connectivity remain operational.”

Tests: No Packet loss that disrupts functionality at 400 ms.

Verification: Apply artificial network delay and monitor performance

PR2

Description: The system shall perform software and firmware upgrades manually initiated and able to complete over high-latency (400 ms) network connections; in case of outage, the system must remain recoverable.

Related Domains: D3, D7

User Story:

“As a technician, I want firmware and software updates to complete safely—even under high latency or outages—so that Pi units do not enter an unrecoverable state and that a technician would have to be sent to fix the issue.”

Test: Unplug device from network during an update.

Verification: Control unit returns to a working rollback state, pre-update.

4.4 Environmental Requirements

Environmental requirements are constraints which are dictated by external factors such as federal government regulations. Also, because these requirements are typically beyond the control of the client, environmental requirements must be complied with. In this section we will review the single environmental requirement put forth by the Coast Guard, its domain, and why it qualifies

as an environmental requirement.

ER1

Description: Fixed facilities shall incorporate intrusion detection capabilities.

Related Domains: D6

User Story:

“As a Coast Guard site manager, I want intrusion detection devices installed directly at facility entry points so that unauthorized access attempts are immediately detected and reported.”

Rationale: The system will be useless if it is unable to be installed into federal buildings as there are governmental agencies that dictate what an organization can install into its facilities.

Verification: Inspect federally allowed hardware and software and ensure the system meets spec.

5.0 Potential Risks

Given the secure nature of the requirements, there are still risks which may need to be considered that are not explicitly stated in the requirements. Being heavily focused on hardware and software, the requirements lack oversight of the architecture which will make the solution even stronger for the Coast Guard. In recent events, we have seen industries suffer greatly when a service that is being used to host applications crash. Another issue that can arise is if a subcomponent fails, how will that failure affect the service/system as a whole? Both of these situations can be addressed in design as fault tolerance and redundancy.

The risks we see in hosting our solution on a single server is critical. A server outage can cause system wide outages which may not be recoverable from. Either database or application server outages will prevent the system from working. To avoid this possible disaster, it is recommended to host applications and/or databases on multiple, distributed servers to ensure that service is not disrupted. This is called “redundancy”. This design solution has some drawbacks in that it may make implementation more complex and incur additional costs for the solution.

Secondarily, the current design requirements do not include a point to address if a subcomponent (such as the control unit) fails. This critical as a failure of a control unit can cause an entire site to become unresponsive and not allow users to enter a site that they must be able to enter. To avoid this possible failure, it is recommended to at least have two control units at a site in case one goes down, the second control unit is able to pick up operations and continue service. This is called “fault tolerance”. While practical to have two control units to back each other up in case of failure, it may not be feasible for all locations. This is because size constraints that may be present at the site to install two controllers. Also this would incur more cost, since it would, in essence, double the number of control units a site may need.

6.0 Project Plan

The Portcullis system solution is a multi-domain project which will require a plan to keep the team focused on fulfilling client requirements in a timely manner. Because some requirements are fulfilled by one or a combination of system components, we will group each requirement with its corresponding component(s) that satisfy it below. The time list will also give us an estimate of which components will be completed and when while also keeping the team on track.

Milestone 1

Pipeline: November 1st - December 10th

Tasks: Database - Data Tables, Database - Transactions

Related Requirements: FR1, FR2, FR4, FR5, FR6, FR12

Goals: Produce a functioning PostgreSQL backend that can store, retrieve, manage audit logs and access events, and store user accounts.

Milestone 2

Pipeline: November 1st - January 30th

Tasks: Administrator Website - API, Administrator Website - Admin Terminal

Related Requirements: FR1, FR2, FR3, FR5, FR6, FR9, FR8

Goals: A fully functional administrator web application with real-time updates and alerts and secure communication to backend APIs.

Milestone 3

Pipeline: February 1st - February 30th

Tasks: Docker - Create configurable environment for Raspberry Pi units

Related Requirements: FR4, FR6, FR7, FR12

Goals: An environment which can be deployed to any number of control units.

Milestone 4

Pipeline: February 1st - February 28th

Tasks: AWS - Configure IoT Infrastructure

Related Requirements: FR5, FR7, FR6, FR8, FR9

Goals: Create cloud infrastructure which is capable of receiving high-latency, secure communication and managing event pipelines.

Milestone 5

Pipeline: February 1st - May 30th

Tasks: Program Raspberry Pi Control Unit, Integrate Hardware with Control Unit

Related Requirements: FR1, FR2, FR4, FR7, FR8, FR9, FR10, FR11

Goals: A complete embedded security system capable of handling authentication, intrusion detection, and communication with backend services.

Milestone 6

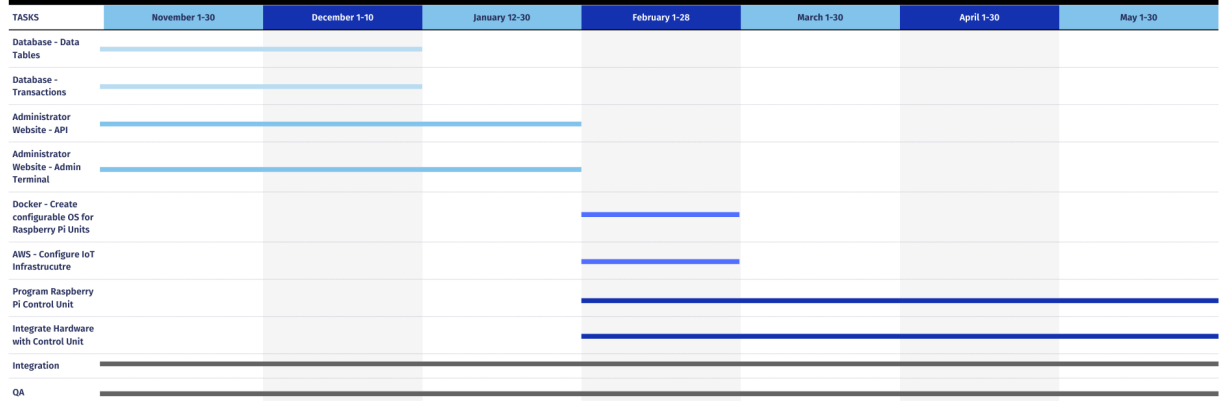
Pipeline: November 1st - May 30th

Tasks: Integration + QA

Related Requirements: All Requirements

Goals: A verified, deployable system successfully meeting all 15 Coast Guard requirements.

Portcullis System Development Gantt Chart



7.0 Conclusion

The Coast Guard is in need of revamping their door access control solution for their installations, which include sites that are in austere environments with dynamic situations. This arises from the need for national security and outdated technology, which is causing the Coast Guard to be behind national efforts in securing federal buildings. Our client, GDMS, has tasked Portcullis to develop a new solution that is updated, dependable, and secure.

Our proposed solution is a system that integrates current latch technology with a Raspberry Pi control unit to securely send and receive data in high-latency, unstable network environments. This solution also incorporates databases to capture and retain logs, credentials, and data to be used in authentication operations, analytics, and audits. Tying everything together, the administrator application will be responsible for interacting with all components of the system, ensuring that events, statuses, and commands are all securely communicated via security protocols to allow notifications for administrators in a timely manner.

In this document, we describe a clear understanding of the system's requirements, establishing objectives that will fulfill each one, identifying risks that fall outside the scope of the requirements, and providing a structured roadmap for development and testing. As we progress with our development, our team is confident that our solution will meet all of the Coast Guard's requirements and enhance GDMS's security infrastructure services. By addressing all these requirements, the project will provide a solid foundation for an extensible framework for future iterations.