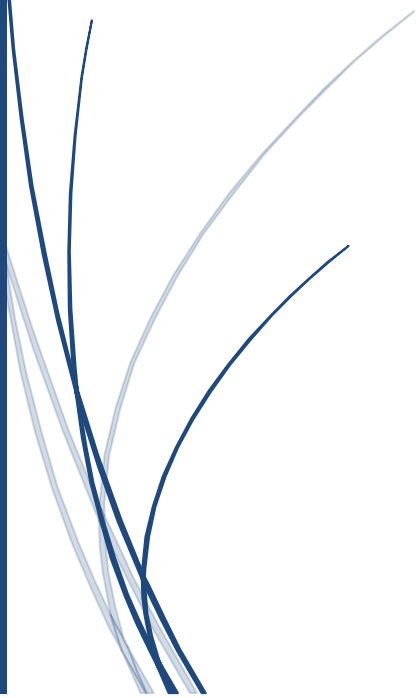
12/5/2018

Rescue911
Emergency Response Information System (ERIS)



ISTM 624-601

Team TECHTRAP

Amit Yadav, Amrutha Girish, Kaitlin Wallace, Mihir Bhende, Nishant Goel, Praveen Gadugin, & Somya Sharma

Statement of Originality

"On my honor, as an Aggie, I have neither given nor received unauthorized aid on this academic work."

Amit Yadav
Amrutha Girish
Kaitlin Wallace
Mihir Bhende
Nishant Goel
Praveen Gadugin
Somya Sharma

TABLE OF CONTENTS

Executive Summary

Introduction

Rescue911 is a county-wide emergency services company that utilizes a call center to dispatch teams to deal with emergency situations. Rescue 911 requires the creation of an Emergency Response Information System (ERIS) to account for several different business processes the company performs. The four key business processes are logging emergency calls, managing Emergency Medical Technician (EMT) base stations, managing payments of direct and indirect subscribers of the service, and capturing patient care data at the emergency scene.

Overview of Business Processes

Recording Calls

Trained operators at the call center are responsible for dispatching teams based on the skill level and teams available. The system built will allow operators to assign priority to an emergency instance, and record other relevant information as it pertains to the emergency. If more than one call has been entered for a particular emergency, the operator will have the ability to examine the details of an existing emergency and link the call if it is the same emergency. Waiting calls are calls that are not able to be assigned to a team right away. Supervisors have the ability to assign calls to teams already dispatched, or hand off the handling of the emergency to another county's emergency services.

Base Station Management

Base stations are where EMT members will clock in and out for their shifts. When a call is logged, it needs to be assigned to the team that's location is closest to the emergency. The existing GPS/GIS system will be integrated with the new ERIS in order to determine the closest team to the emergency. Teams will be dispatched from the base stations and all records of the teams' activities will be captured in this system.

Subscriber & Billing Management

There are both direct and indirect subscribers of the Rescue 911 service. Direct subscribers pay a monthly fee directly to Rescue 911. Indirect subscribers are subscribers through insurance policies, and their insurance providers pay Rescue 911 for services rendered. Finally, there are non-subscribers who pay Rescue 911 directly for services rendered. Managers will be able to retrieve payment information on direct, indirect, and non-subscribers through the subscriber management sub-system. The integration of this process with the patient care reporting process will ensure that insurance providers are given accurate information to expedite the billing process.

Recording Patient Care Data

Finally, EMTs dispatched to deal with an emergency will record all patient care data within the system. This includes vital signs as well as what hospital the patient was sent to. Integration of this process with the rest of the system is vital, as this patient care information is needed for billing purposes. Reporting will be able to be performed with this process in order to track what EMTs responded to what emergencies and related queries.

Assumptions

Some key assumption has been made in the design of the system. First and foremost, it will be assumed that Rescue 911 is located in Texas. This is important to take into consideration when it comes to laws and regulations that emergency service providers need to comply with. Because ERIS will be a countywide system, when a call cannot be fulfilled by Rescue 911 EMT teams, the call will be passed off to one of the surrounding county's emergency providers. The GPS/GIS system is an external system that will be integrated with the business processes within the ERIS. The Automatic Vehicle Location (AVL) system used for tracking EMT teams in the field is also an essential external system needed for daily operations of Rescue 911. The responsibility for integrating both the GPS/AVL systems will lie on the vendors of this software. Supervisors for operators will have privileged access to ERIS that means they can deal with waiting calls and query for EMT teams in the field when dealing with unassigned calls.

When it comes to the subscriber management processes, we assume that direct users pay a monthly fee for the service. Indirect subscribers' insurance providers will pay on a service-by-service basis. This means that some indirect subscribers may never utilize the service, and thus the insurance will not need to pay. Finally, non-subscribers will also be billed on a service-by-service basis. We also assume that subscription and payment modification requests are rare in the system and that only Rescue 911 managers have the authorization to modify subscriptions in the rare case that occurs.

When it comes to implementation and deployment of the system, we assume that a backup server that mirrors the current production server will be running in order to ensure

constant availability. The ERIS Application Server will be clustered to meet throughput and availability requirements

Finally, we have assumed that managing emergency medical technician teams is Rescue 911's core competency. For this reason, fire and police services will be considered external entities to our organization, so we do not track team members or names, employment information for the members of fire department teams or police teams as these are managed by the state/county. We do, however, treat police and fire services as we would external providers for medical emergencies in that we track a point of contact, provider name, and link calls to these external providers along with the operator who dispatched the services to the emergency.

Abbreviation List

Abbreviation	Definition
ERIS	Emergency Response Information System
CAD	Computer-Aided Dispatch System
BSMS	Base Station Management System
SMS	Subscriber Management System
EPCR	Electronic Patient Care Reporting System
EMT	Emergency Medical Technician
GIS	Geographic Information System
GPS	Global Positioning System
AVL	Automatic Vehicle Location
DFD	Data Flow Diagram
ERD	Entity Relationship Diagram
UML	Unified Modeling Language
SQL	Structured Query Language
HIPPA	Health Insurance Portability and Accountability Act
PCI-DSS	Payment Card Industry Data Security Standards

Extension of Rescue 911 Case

The Rescue 911 had cased the following extensions that were covered during the modeling of the system:

1. ERIS must allow **online and text-based reporting** of problems

   The system design shows emergency incidents reported as calls but they can also be reported online or through phone texts. These still go through the operator and all the incident details are tracked and stored by the operator. In short, the system works for emergency incidents reported through calls, texts and online.

2. ERIS should have **tracking capabilities** for all reported emergencies

   - An incident is tracked as "logged," "waiting," "actioned," "on-scene," and "closed."
   - Response teams begin and end a shift in "invalid" state. During the shift, they can be in "available," "away," or "double-dispatch" state.

   As and when the state of an incident or team changes, the system is designed to accordingly update the status of the incident or the team. Hence, using an incident ID, the status of the incident can be tracked and if a team has been assigned, then the team's status can also be tracked. They have been covered in the UML diagrams.

3. ERIS system must **work for a county** rather than a single company

   If we do not find a team available for dispatch, the call which is in waiting status is transferred to an external emergency service company in a geographically nearby county. Once they confirm that they can take in the incident, the incident call will be moved to "closed" status after storing the details of the external service provider.
   This has been reflected in the "waiting call" sequence diagram and also in other UML diagrams.

4. The call centers must be able to be **virtual**, meaning volunteers and employees can work from anywhere that has a phone line

   The system will be implemented on a 3 tier web-based architecture - browser client, web server/application server (business logic) and database server
   All volunteers and employees have a phone line or handheld devices. They will be able to access the online system to note down and update all details. The handheld devices will replace the field notebooks that were used by EMTs to record patient data.

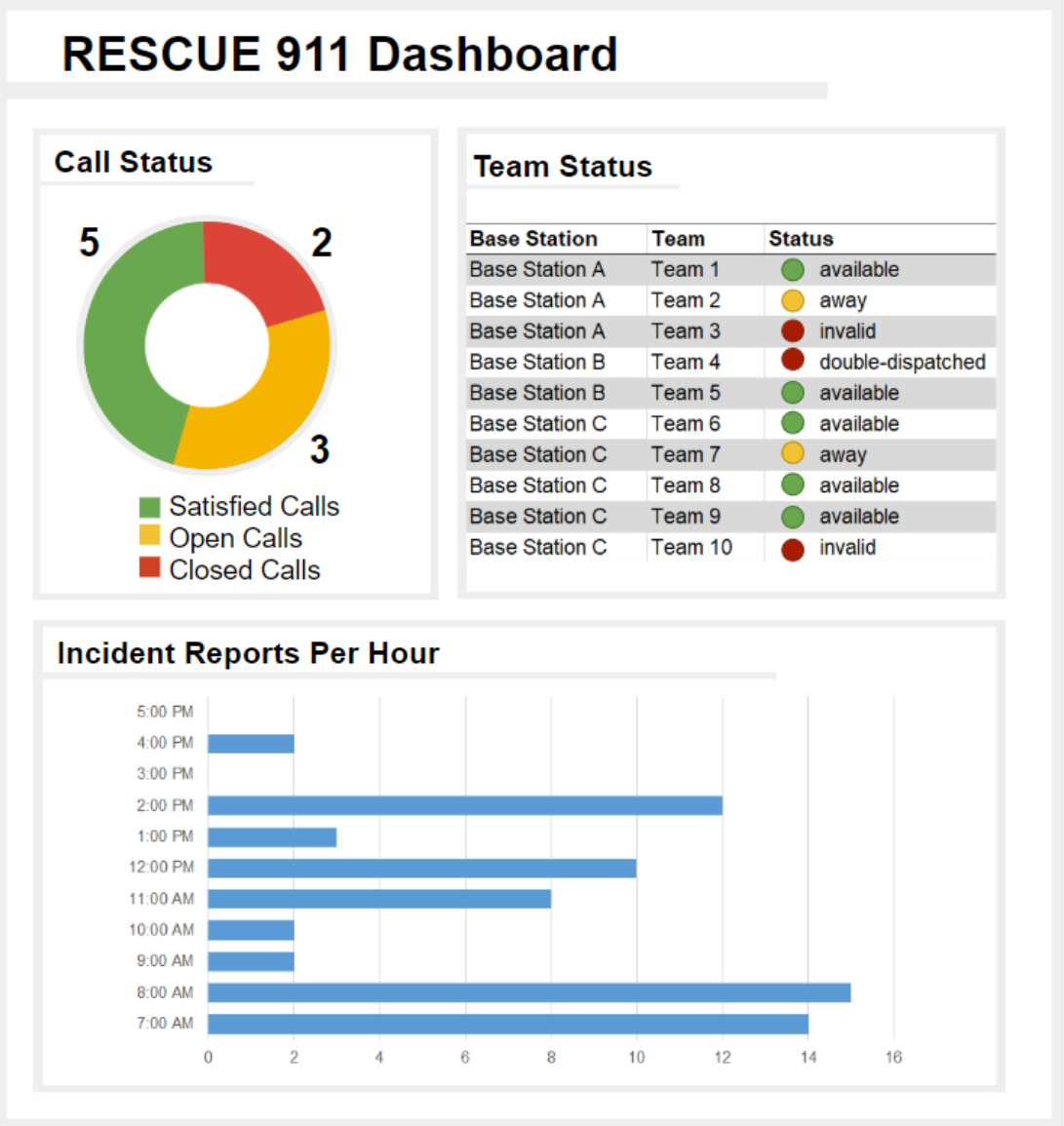
5. ERIS system must allow you to dispatch **fire, life**, or **police services** to an emergency.

   An incident is assigned an attribute, category, to determine the type of emergency. A medical emergency will be assigned EMT teams who will attend to the emergency on-scene and direct the patients to the hospital if required. Incident handling, subscriber management, EMT team dispatching, and patient care details recording will be done through the ERIS system. However, the system does not have personnel trained to handle fire or police services. In such cases, the call will be transferred to an external service provider who can assist in attending to the incidents

   The same has been incorporated in the ERDs and class diagram. The sequence diagram states this action as an assumption.

   Assumption: Fire and police services are external to our organization and so we do not track team members or names, employment information for the members of fire department teams or police teams as these are managed by the state/county.

6. A dashboard that shows how many of each call is open, satisfied, or closed for any other reason (such as the problem being out-of-area).

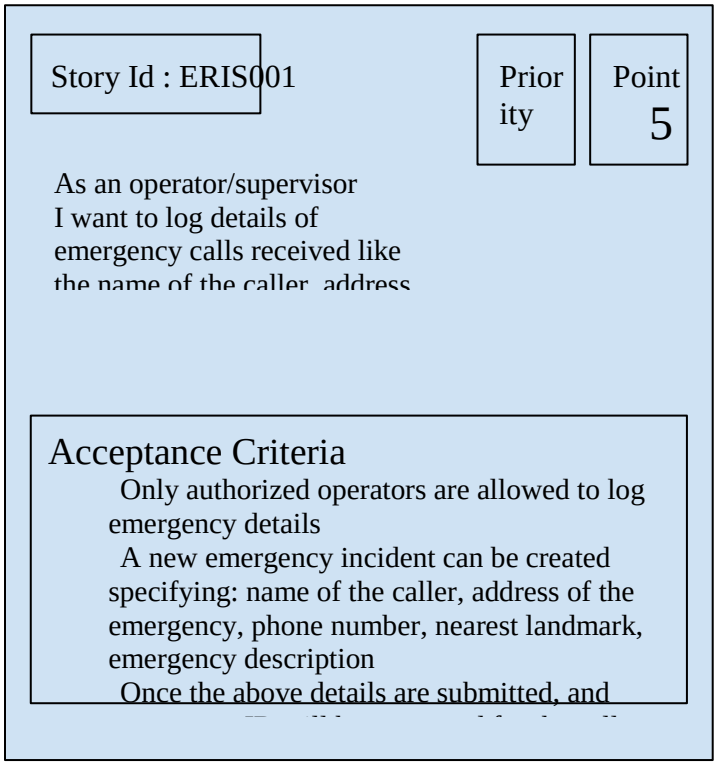

RESCUE 911 Dashboard
Call Status
5
2
3
Satisfied Calls
Open Calls
Closed Calls
Team Status
Base Station | Team | Status
Base Station A | Team 1 | available
Base Station A | Team 2 | away
Base Station A | Team 3 | invalid
Base Station B | Team 4 | double-dispatched
Base Station B | Team 5 | available
Base Station C | Team 6 | available
Base Station C | Team 7 | away
Base Station C | Team 8 | available
Base Station C | Team 9 | available
Base Station C | Team 10 | invalid
Incident Reports Per Hour
5:00 PM
4:00 PM
3:00 PM
2:00 PM
1:00 PM
12:00 PM
11:00 AM
10:00 AM
9:00 AM
8:00 AM
7:00 AM
0
2
4
6
8
10
12
14
16

Rescue911 Case User Stories

Based on the Rescue 911 Case, below user stories are identified to develop the ERIS. The user stories are prioritized based on the requirement.

Priority	Criteria
1	Highly significant feature for the system to operate
2	An important feature that adds value
3	Additional feature

Story points are assigned to the user stories considering the risk, complexity and time involved.



Story Id : ERIS002	Priority	Point 3

As an operator
I want to be able to assign a priority category to an emergency and number of

Acceptance Criteria

- Display the description of the emergency so that the operator can refer the same while assigning priority and no. of teams
- Operator should be able to input Priority Category and number of response teams corresponding to an emergency situation

Story Id : ERIS003	Priority	Point 5

As an operator
I want to be able to dispatch one or more response team once the emergency details are

Acceptance Criteria

- Operator can find the nearest base station based on the landmark and zipcode using the GIS system
- Based on the Base Station Id retrieved, availability of response teams is determined using AVL and base station records

Story Id : ERIS004	Priority	Point
		5

As an operator
I want to be able to follow up on the calls
so that I can keep a track of the

Acceptance Criteria

- Operators can query emergency records based of the emergency status or emergency id
- While querying with a particular status (like "Actioned' or 'Logged'), all the calls that belong to that status, will be displayed
- While querying with an emergency ID,

Story Id : ERIS005	Priority	Point
		8

As a supervisor
I should be able to deal with the emergency calls to which
operators are unable to dispatch

Acceptance Criteria

- Supervisor should be able to query based on the emergency status
- Display all the calls that are in "waiting" status
- Prioritize the calls according to emergency priority category

Story Id : ERIS006	Priority	Point 3

As an operator
I want to link a new call with same emergency to an existing emergency

Acceptance Criteria

Operator is query emergencies based on the zip code and landmark

Display all the emergency IDs, addresses and descriptions of calls received during the day which has the same zip code and landmark

Story Id : ERIS007	Priority	Point 3

As an operator
I want to be able to find the closest base station to the emergency location

Acceptance Criteria

Operator should be able to input landmark, address and zip code for an emergency

Display the nearest base station using GIS/GPS systems based on the address, zip code and landmark

GIS/GPS system returns the Id of the

Story Id : ERIS008	Priority	Point
		2

As an operator
I want to input the base station ID of the nearest base station so that the available team

Acceptance Criteria

- Operator should be able to input Base station ID
- The field should accept only alphanumeric characters
- The length of the text box should be limited to a maximum of 20 characters

Story Id : ERIS009	Priority	Point
		2

As an operator
I want to update the status of an emergency as “waiting” if there are no teams available at the

Acceptance Criteria

- A team that needs to be dispatched will be radioed by the operator
- The team will be marked as non-responsive if no acknowledgement is received within 15 seconds during the first attempt and within next 10 seconds during the second attempt

Story Id : ERIS010	Priority	Point
		3

As an operator
I want to assign a new team to an emergency when no response is obtained from the

Acceptance Criteria

- Operator should be able to edit and update the response team assignment for a particular emergency
- The new team that is being assigned to the emergency should have confirmed their availability and action.

Story Id : ERIS011	Priority	Point
		2

As an operator
I want to be able to change the status of emergency call in the system from “logged” to

Acceptance Criteria

- Operator should be able to search the emergency based on the status
- Status should be changed to “Actioned” when the team assigned for the emergency responds to the radio announcement
- A call whose status has been changed to

Story Id : ERIS012	Priority	Point 3

As an operator
I want to be able to change the status of the EMT in the system from “available” to “away”

Acceptance Criteria

Operator can change the status of EMT when response received by EMT

As soon as team is assigned for an emergency, operator is enabled to change the status of all the EMTs of that response teams

A team member’s status cannot be

Story Id : ERIS013	Priority	Point 3

As an operator
I want to be able to change the state of response team availability to ‘available’ once

Acceptance Criteria

Operator can update the status back to available when notified by response team

Operator would first search the emergency based on the unique emergency id

Operator can then find the response team working on the emergency, and change the

Story Id : ERIS014	Priority	Point 3

As a supervisor
I want to be able to change the availability status of response team status to 'double-dispatch'

Acceptance Criteria

- A supervisor is authorized to assign a team status of "double dispatch" when status is "away"
- Status can be changed to "double dispatch" only if the team was previously in "away" status.

Story Id : ERIS015	Priority	Point 5

As an EMT
I want to log into the ERIS system
so that I can report for duty

Acceptance Criteria

- EMT can login to ERIS to log the clock in/ punch in time
- EMT status changes from invalid to available
- EMT team status is changed to available when more than two members clock in the system
- We should be able to assign a shift grade to all

Story Id : ERIS016	Priority	Point
		5

As an EMT
I want to input details about patient care provided at emergency scenes

Acceptance Criteria

As EMT is assigned to an emergency, EMT should be able to edit and add information of multiple patients from that particular emergency

There should be a fields to record patient's vital signs (heart rate and Blood pressure),

Story Id : ERIS017	Priority	Point
		8

As a manager
I want to generate patient care report
so that I can analyze the

Acceptance Criteria

Manager are authorized to view details and reports of all the patients' data recorded

Manager should be able to select the details required in the report and filter accordingly

Details like patient id, patient name, reported by EMT Id, related emergency id, emergency

Story Id : ERIS018	Priority	Point
		8

As a manager of call center
I want to generate exception
report at end of each shift of all
emergency responses that are

Acceptance Criteria

Manager should be authorized to generate report for the emergency responses

Story Id : ERIS019	Priority	Point
		5

As a manager of base station
I want to generate exception
report of patient's records
so that patient's data is handed

Acceptance Criteria

Base station manager should be authorized to generate exception reports for patients

Story Id : ERIS020	Priority	Point
		8

As a manager
I want to generate Schedule reports each week
so that I can process the

Acceptance Criteria

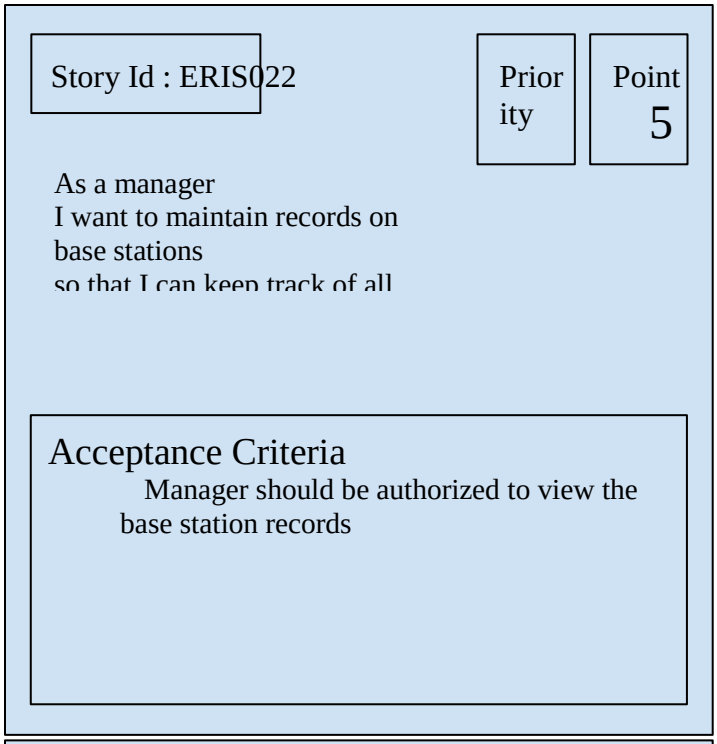
Manager should be authorized to generate schedule reports

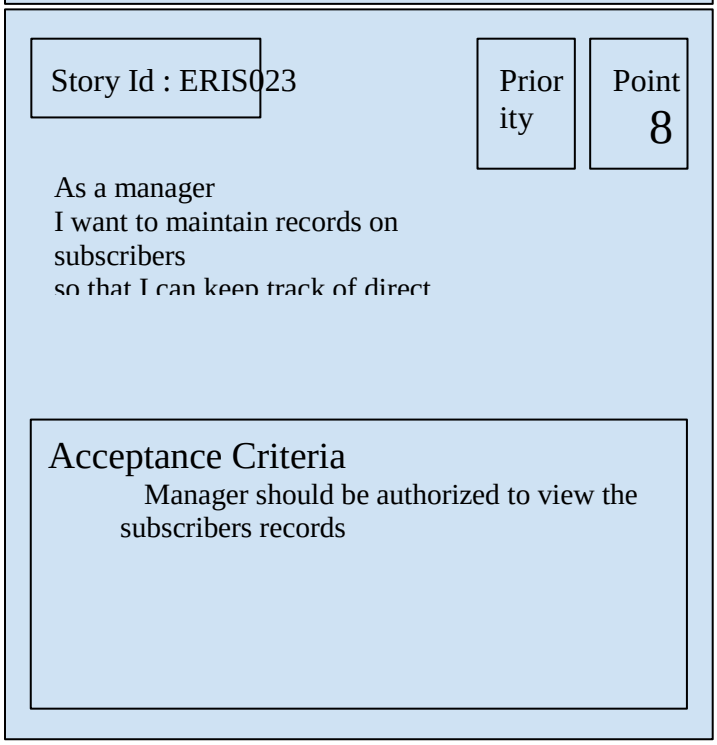
Story Id : ERIS021	Priority	Point
		8

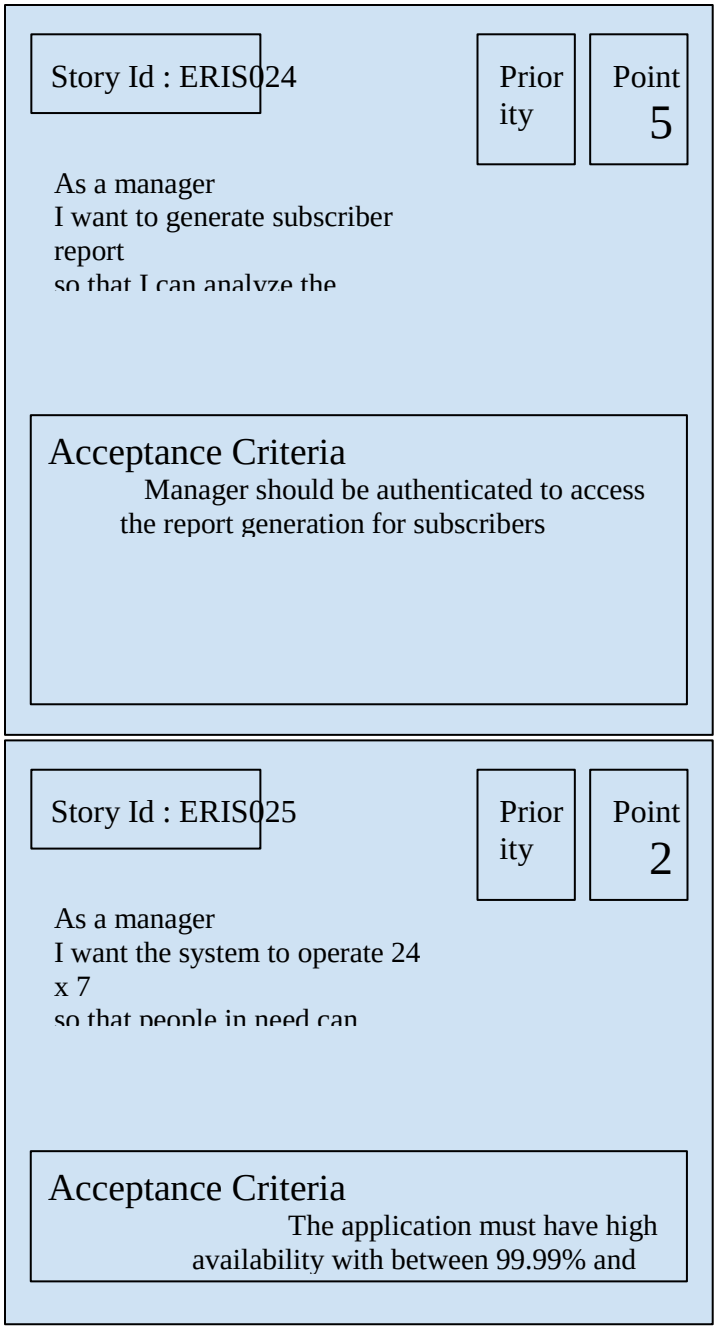
As a manager
I want to maintain response team and EMT staff records
so that I can efficiently manage

Acceptance Criteria

Manager should be authorized to view the EMT staff records


Story Id : ERIS022
Priority
Point
5
As a manager
I want to maintain records on
base stations
so that I can keep track of all
Acceptance Criteria
Manager should be authorized to view the
base station records


Story Id : ERIS023
Priority
Point
8
As a manager
I want to maintain records on
subscribers
so that I can keep track of direct
Acceptance Criteria
Manager should be authorized to view the
subscribers records


Story Id : ERIS024
Priority
Point
5
As a manager
I want to generate subscriber report
so that I can analyze the
Acceptance Criteria
Manager should be authenticated to access the report generation for subscribers
Story Id : ERIS025
Priority
Point
2
As a manager
I want the system to operate 24 x 7
so that people in need can
Acceptance Criteria
The application must have high availability with between 99.99% and

Story Id : ERIS026	Priority	Point
		8

As a COO
I want to generate a weekly summary report
so that I can keep track of

Acceptance Criteria

COO should be authenticated to access

Story Id : ERIS027	Priority	Point
		5

As a manager
I want to secure the application and data
so that the sensitive and

Acceptance Criteria

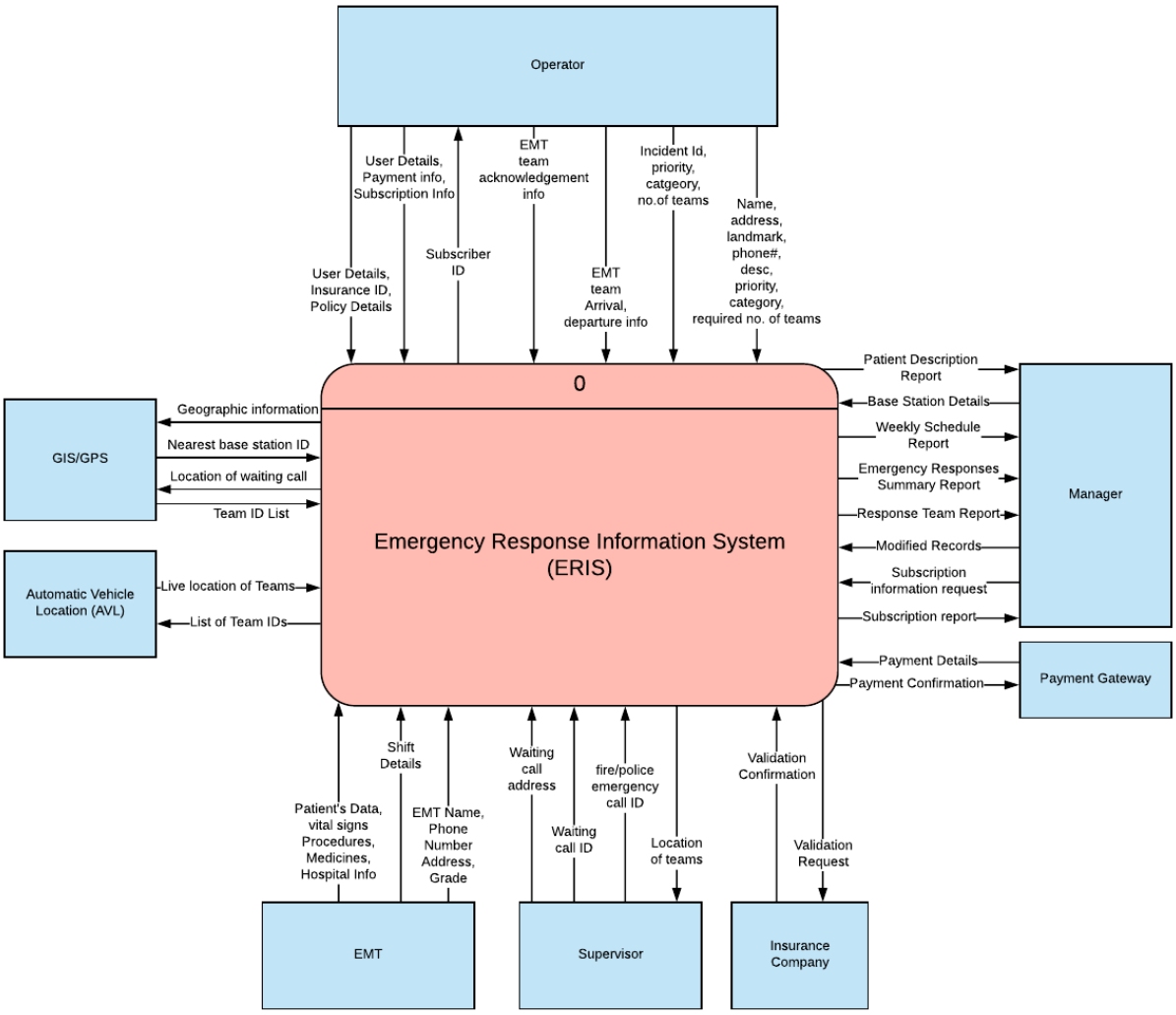
Secured data requires authenticated

Story Id : ERIS028	Priority	Point 2

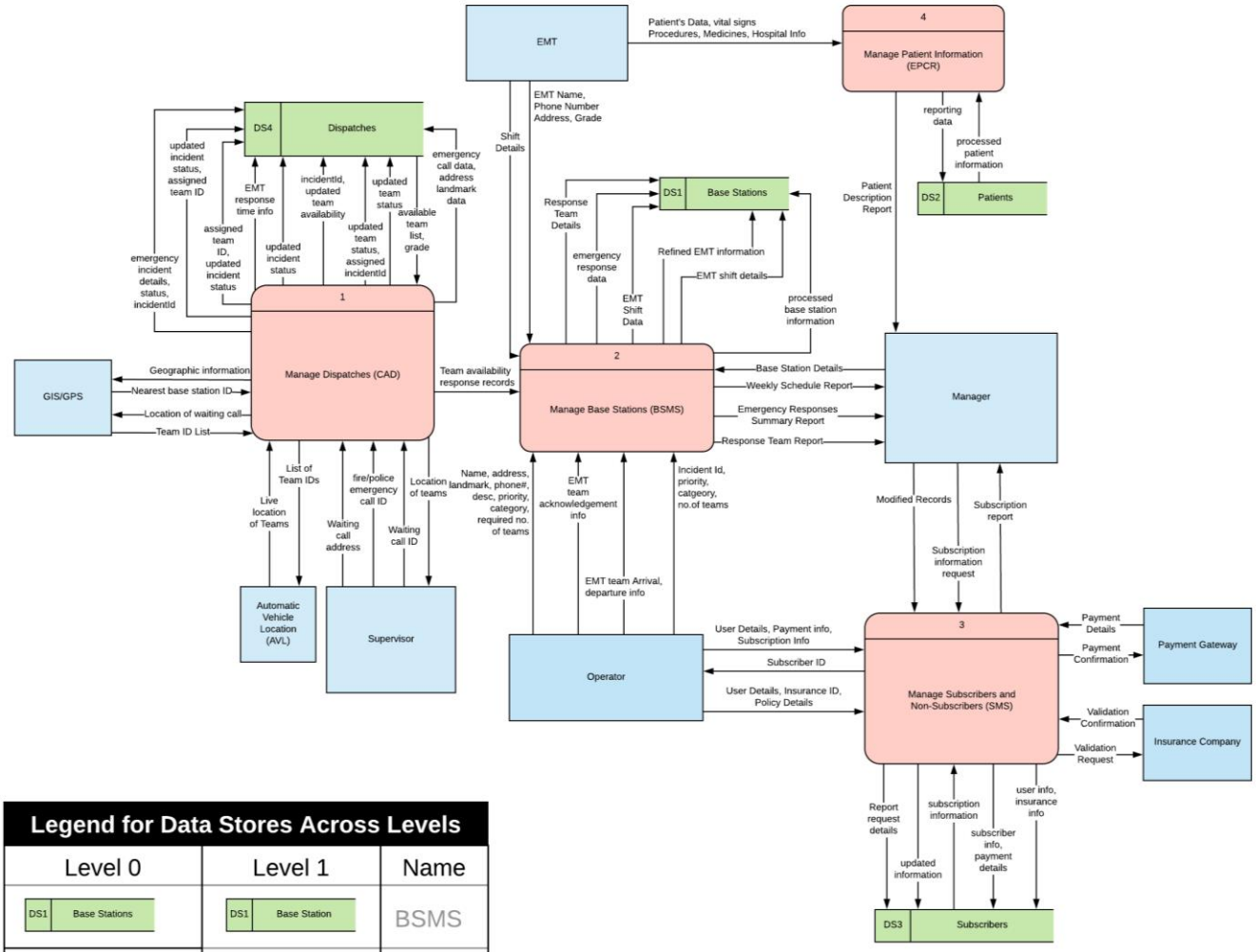
As a manager
I want high performance for the application without any

Acceptance Criteria
There should not be any delays and

DFD Context Level



DFD Level 0

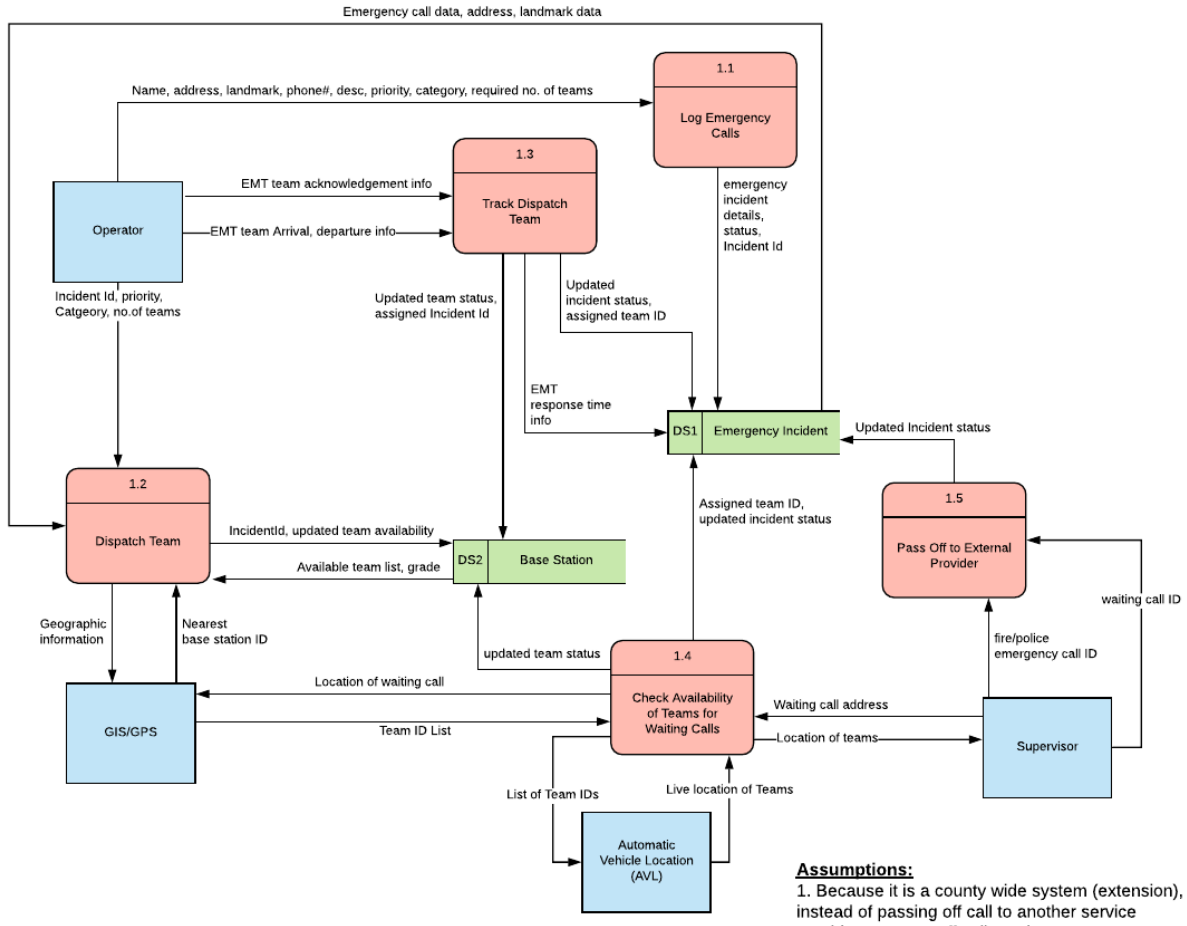


Legend for Data Stores Across Levels

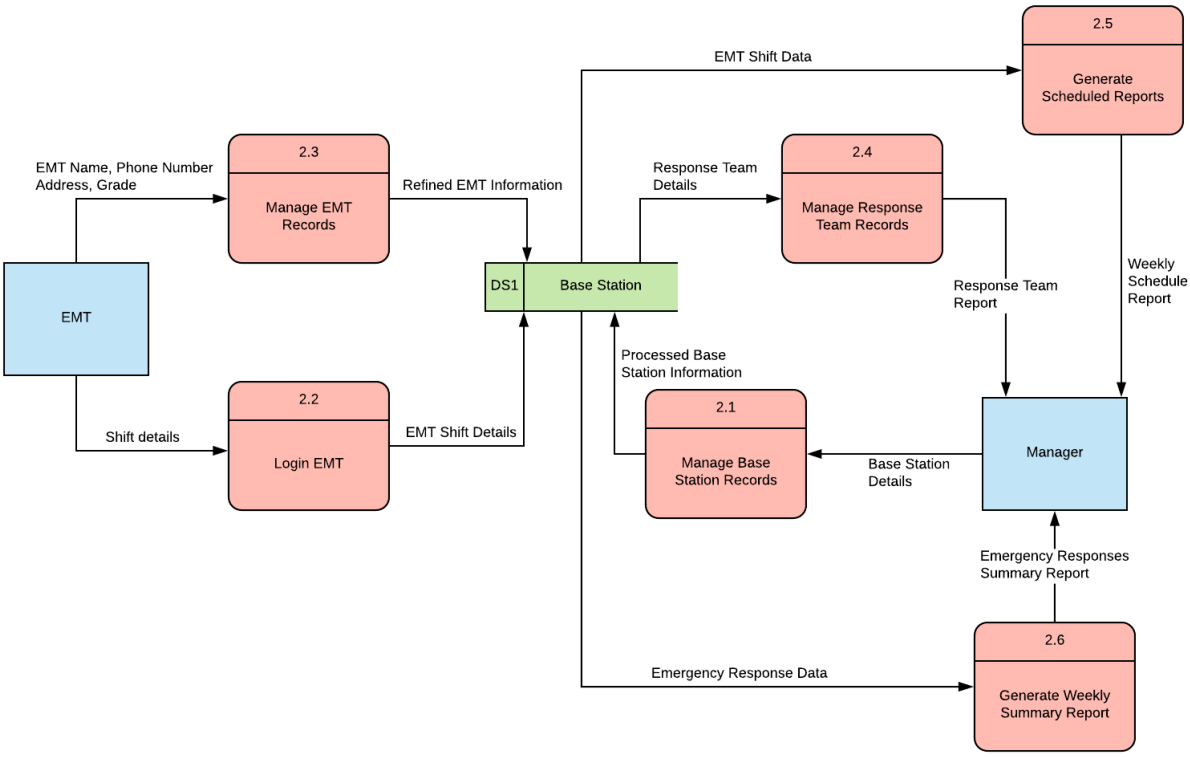
Level 0	Level 1	Name
DS1 Base Stations	DS1 Base Station	BSMS
DS2 Patients	DS1 Patient Information	EPCR
DS3 Subscribers	DS1 Subscribers	SMS
DS4 Dispatches	DS1 Emergency Incident DS2 Base Station	CAD

DFD Level 1

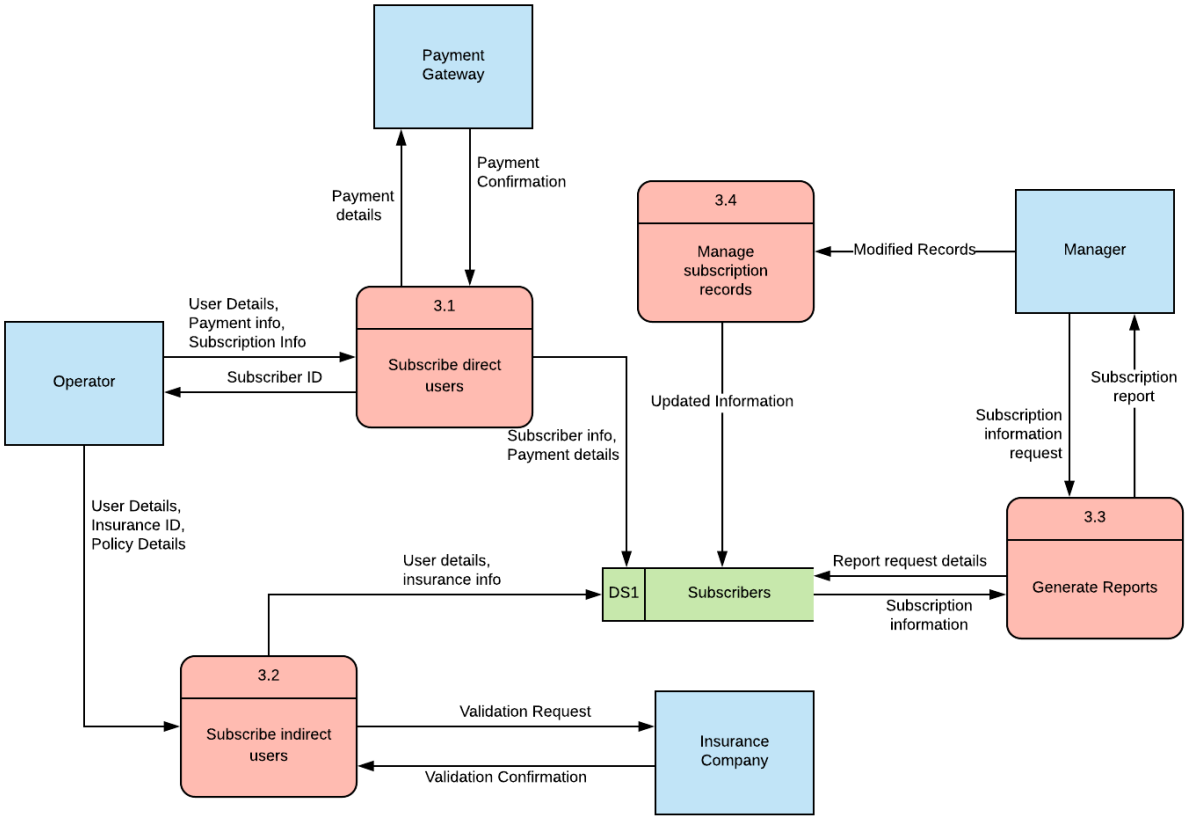
I. CAD Subsystem



II. BSMS Subsystem



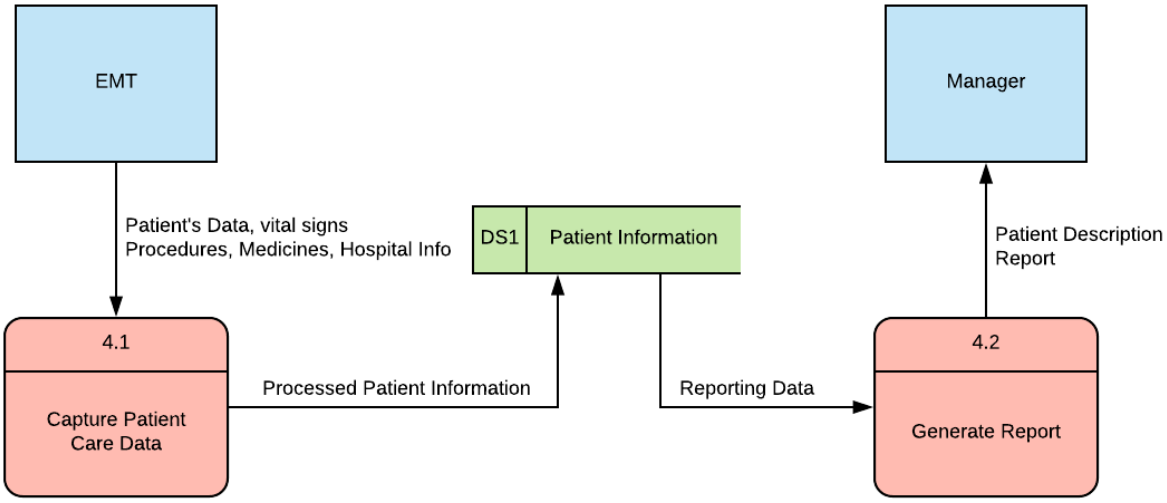
III. SMS Subsystem

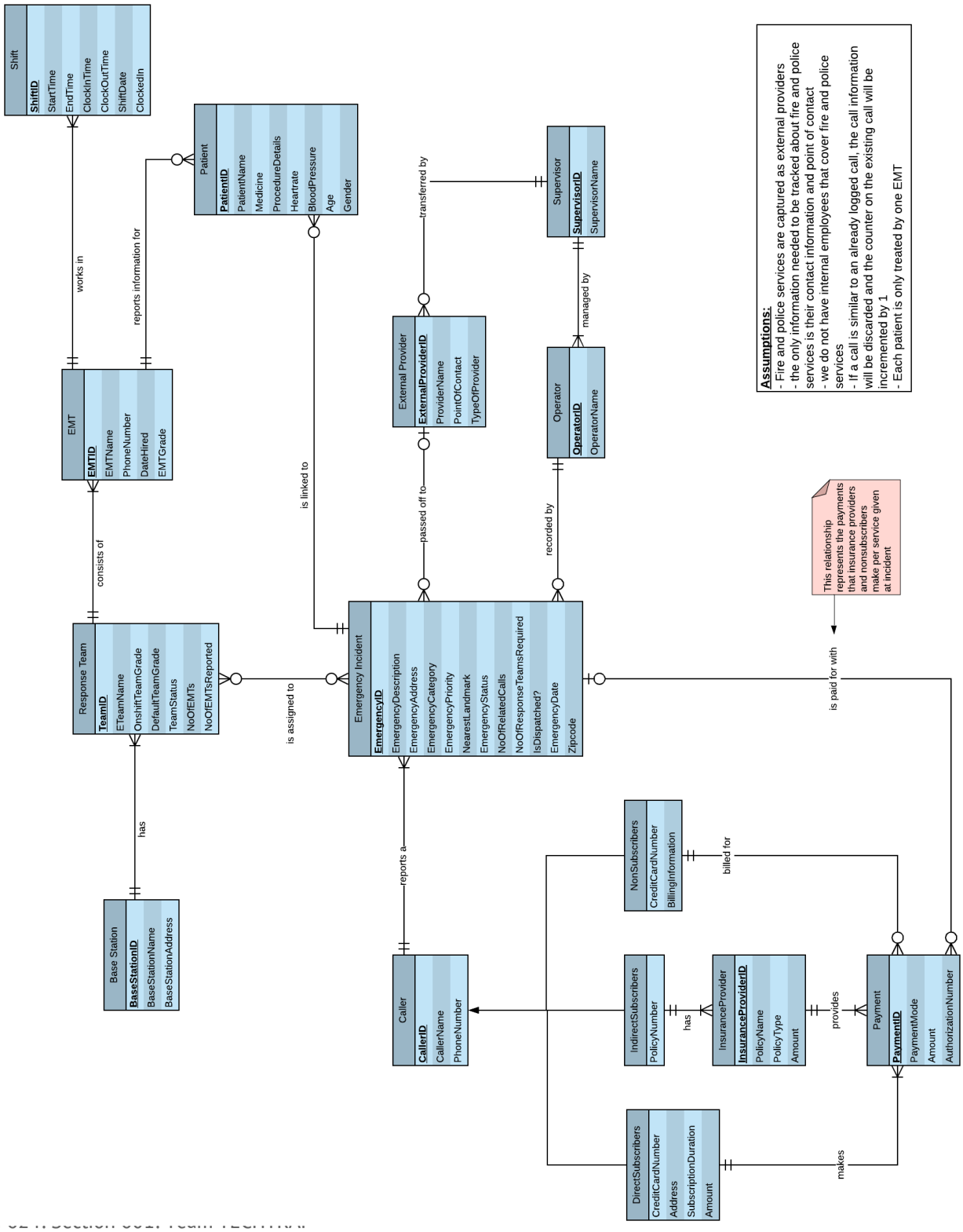


Assumptions:

1. Subcription and payment modification requests are rare.
2. Only Managers are authorized to modify subscriptions.

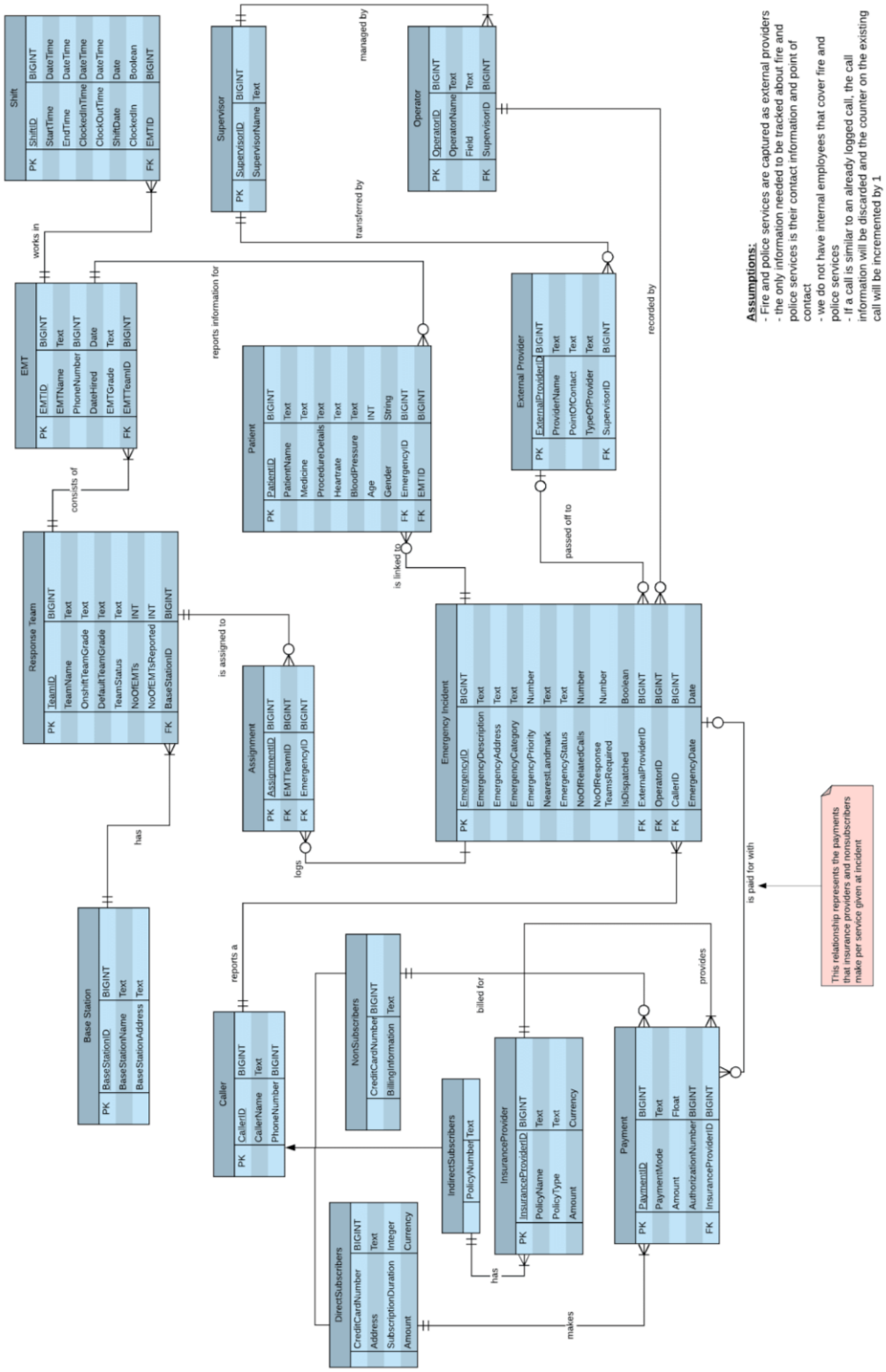
IV. EPCR Subsystem



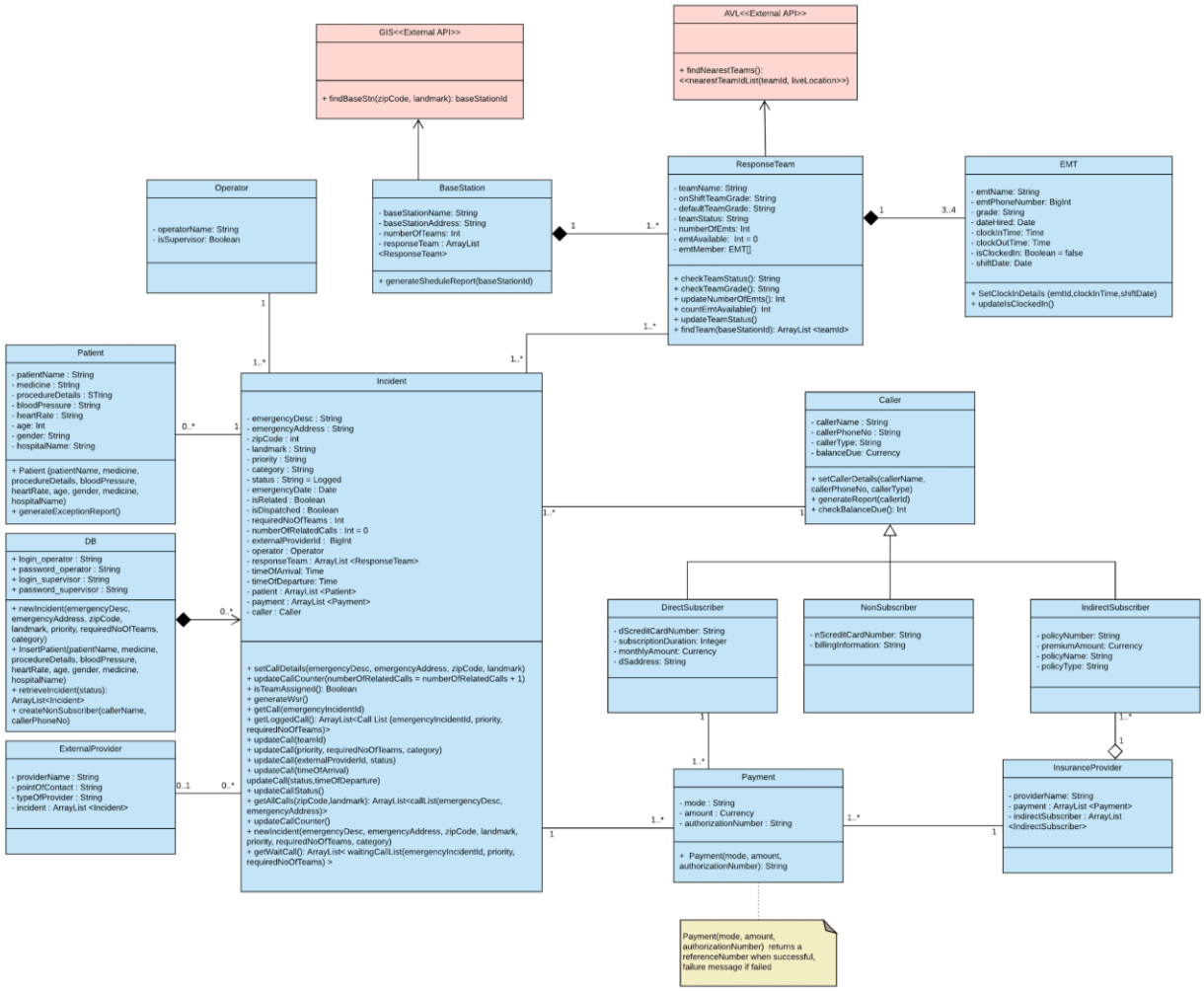

Shift
ShiftID
StartTime
EndTime
ClockInTime
ClockOutTime
ShiftDate
ClockedIn
Patient
PatientID
PatientName
Medicine
ProcedureDetails
Heartrate
BloodPressure
Age
Gender
Supervisor
SupervisorID
SupervisorName
works in
reports information for
transferred by
EMT
EMTID
EMTName
PhoneNumber
DateHired
EMTGrade
External Provider
ExternalProviderID
ProviderName
PointOfContact
TypeOfProvider
managed by
Operator
OperatorID
OperatorName
consists of
is linked to
passed off to
recorded by
Assumptions:
- Fire and police services are captured as external providers
- the only information needed to be tracked about fire and police services is their contact information and point of contact
- we do not have internal employees that cover fire and police services
- If a call is similar to an already logged call, the call information will be discarded and the counter on the existing call will be incremented by 1
- Each patient is only treated by one EMT
This relationship represents the payments that insurance providers and nonsubscribers make per service given at incident
Response Team
TeamID
ETeamName
OnshiftTeamGrade
DefaultTeamGrade
TeamStatus
NoOfEMTs
NoOfEMTsReported
Emergency Incident
EmergencyID
EmergencyDescription
EmergencyAddress
EmergencyCategory
EmergencyPriority
NearestLandmark
EmergencyStatus
NoOfRelatedCalls
NoOfResponseTeamsRequired
IsDispatched?
EmergencyDate
Zipcode
is assigned to
is paid for with
has
reports a
Base Station
BaseStationID
BaseStationName
BaseStationAddress
Caller
CallerID
CallerName
PhoneNumber
NonSubscribers
CreditCardNumber
BillingInformation
billed for
IndirectSubscribers
PolicyNumber
has
InsuranceProvider
InsuranceProviderID
PolicyName
PolicyType
Amount
provides
Payment
PaymentID
PaymentMode
Amount
AuthorizationNumber
DirectSubscribers
CreditCardNumber
Address
SubscriptionDuration
Amount
makes

Logical ERD

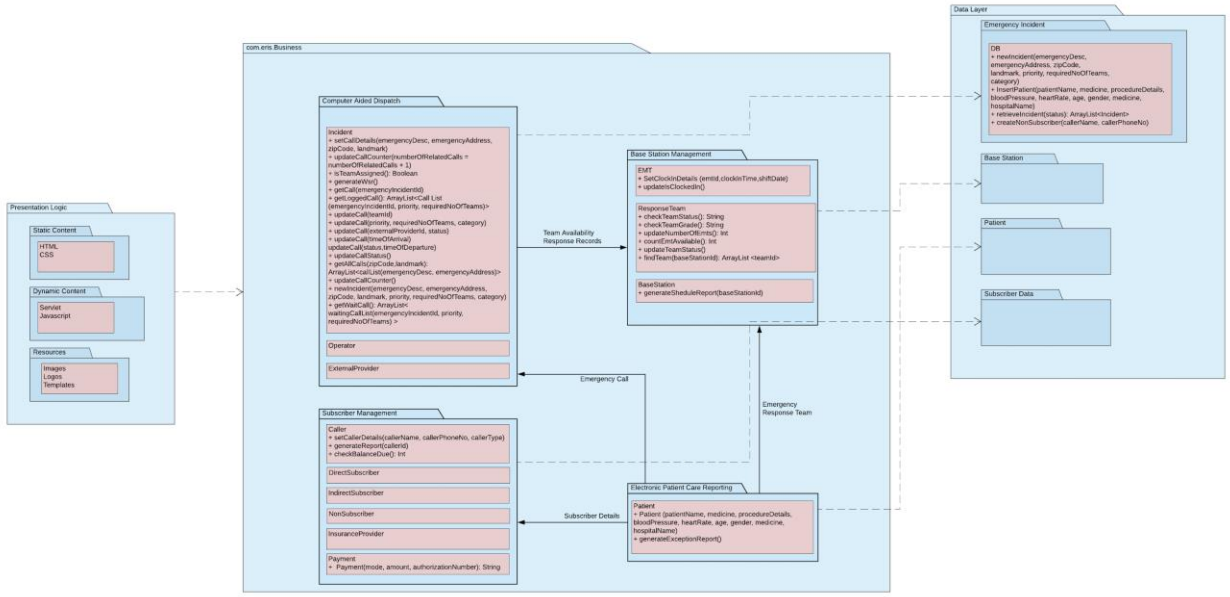
Physical ERD



Class Diagram

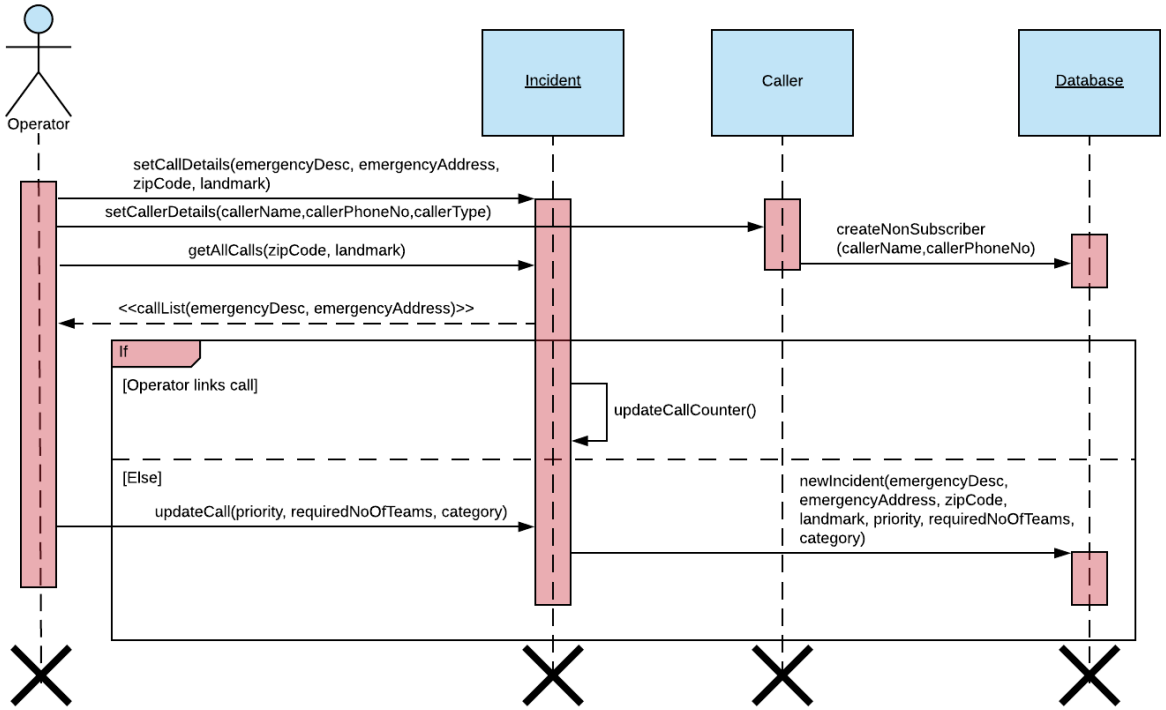


Package Diagram



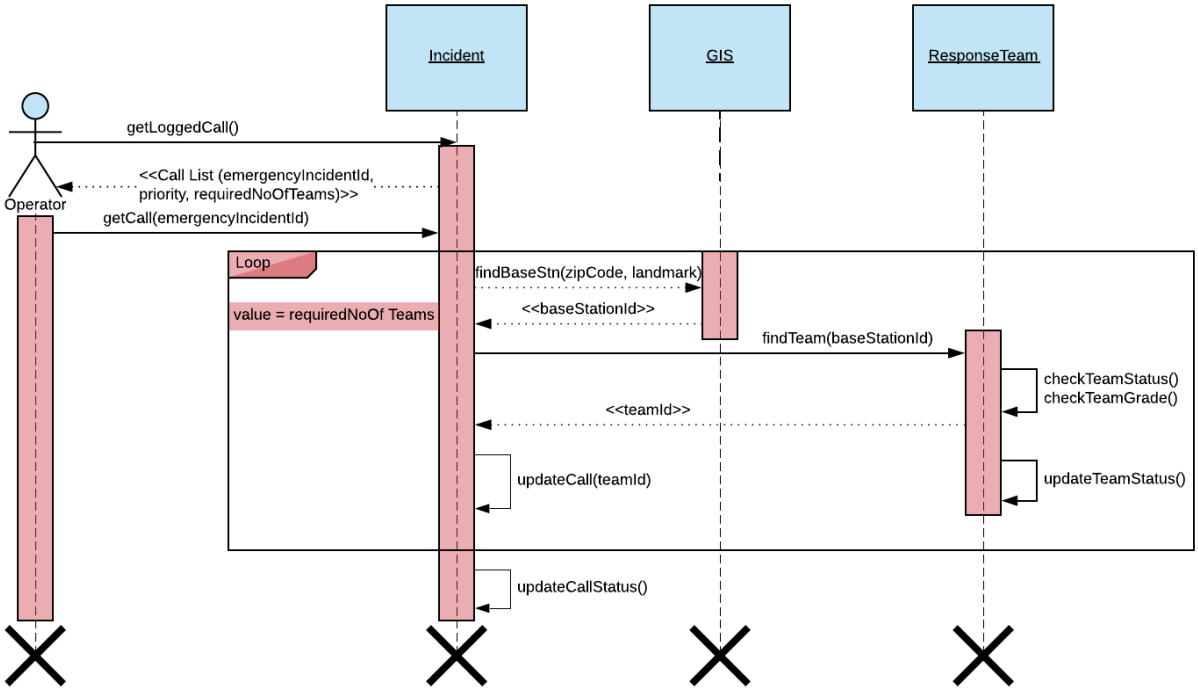
Sequence Diagrams

I. Log Emergency Calls



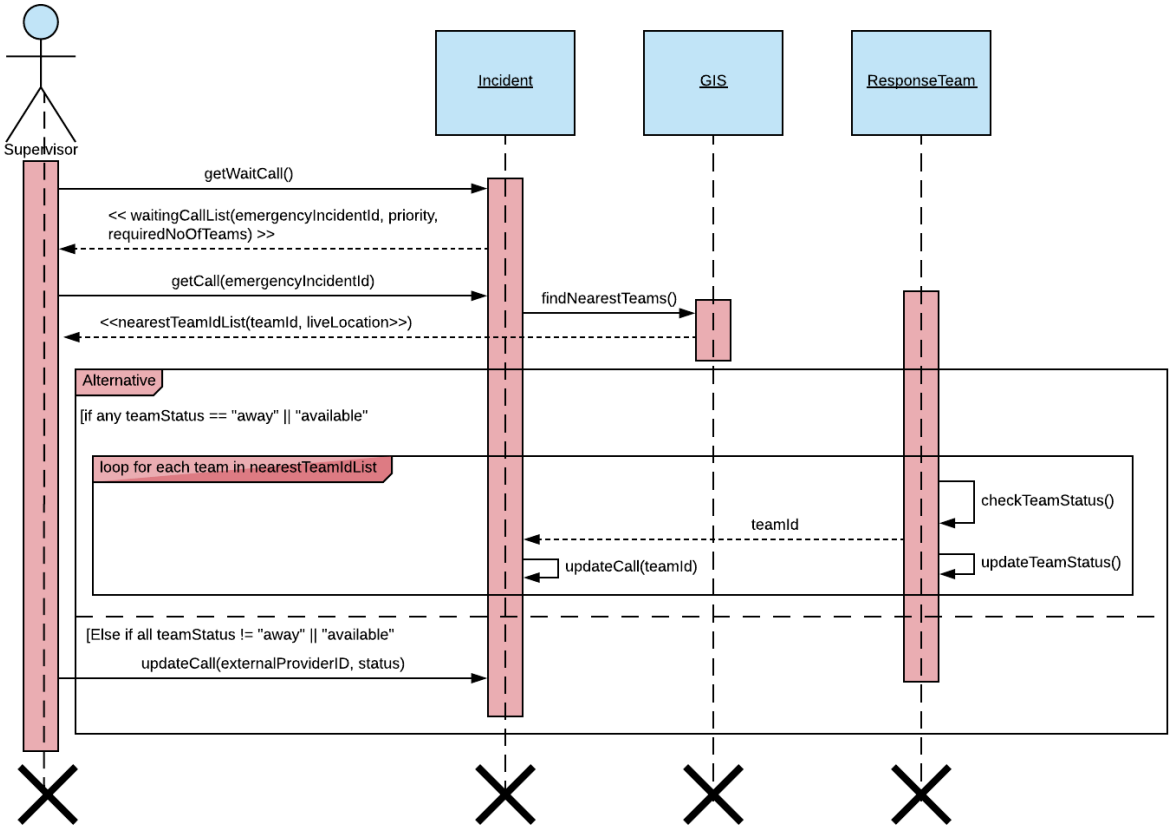
Assumptions:
- Operator decides priority, assigns category (fire, life, police), and number of teams when logging call
- If the incident requires fire or police involvement, the call will be passed off to an external agency by a supervisor (like a waiting call)
- Landmark is prepopulated in a combobox on UI

II. Dispatch Emergency Calls



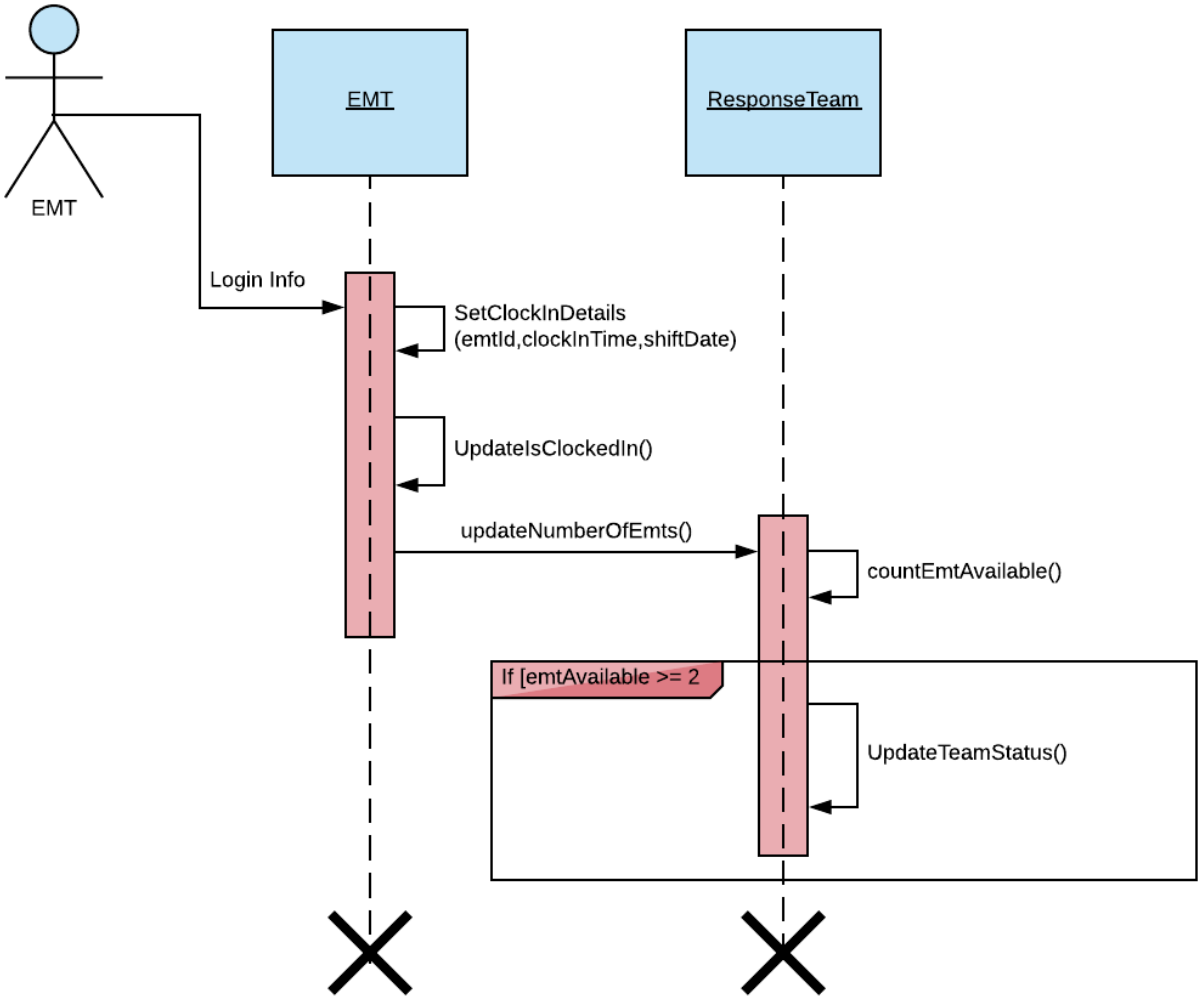
Assumptions:
- GIS class is external connection to the GIS system.

III. Assign Waiting Calls

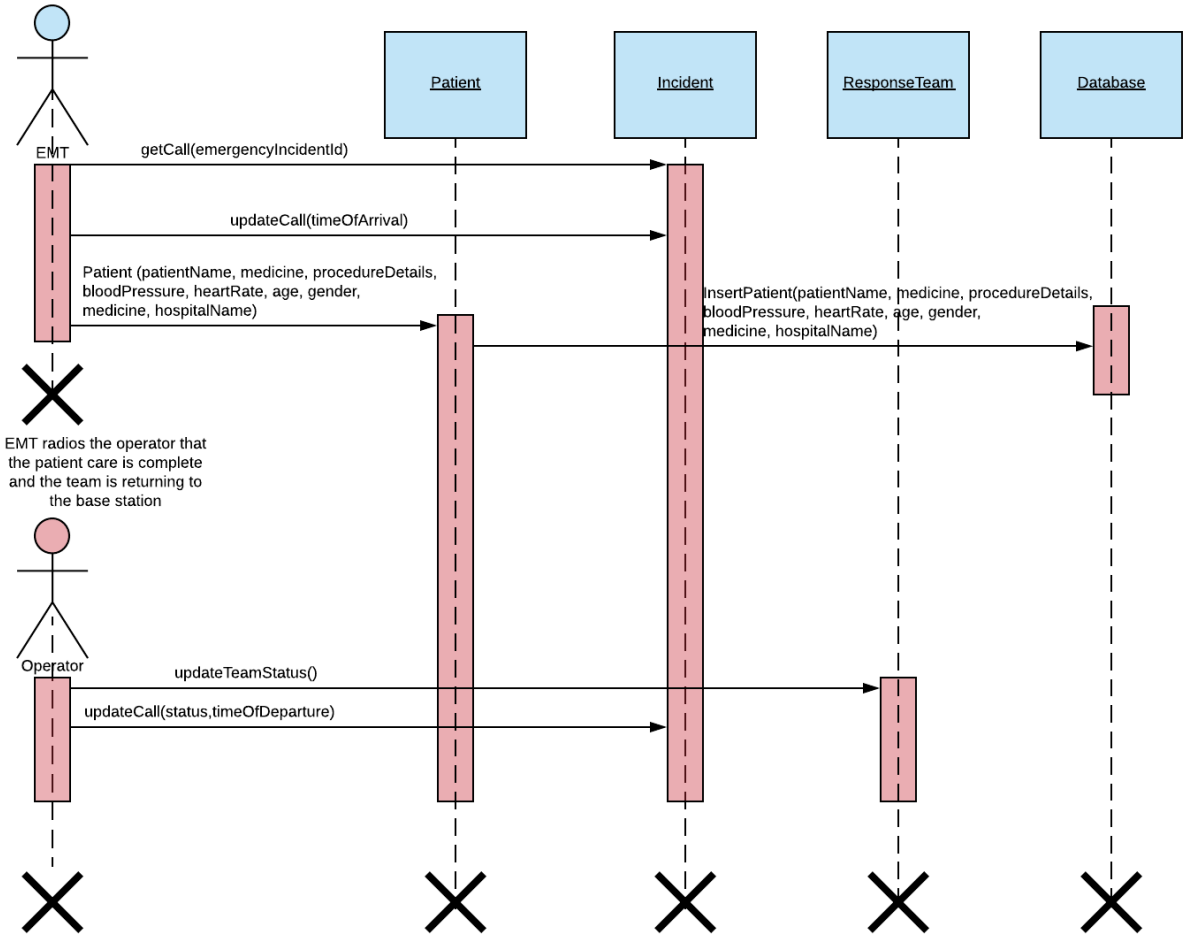


Assumptions:
- GIS integrated with AVL
- Call details like landmark can be entered through UI

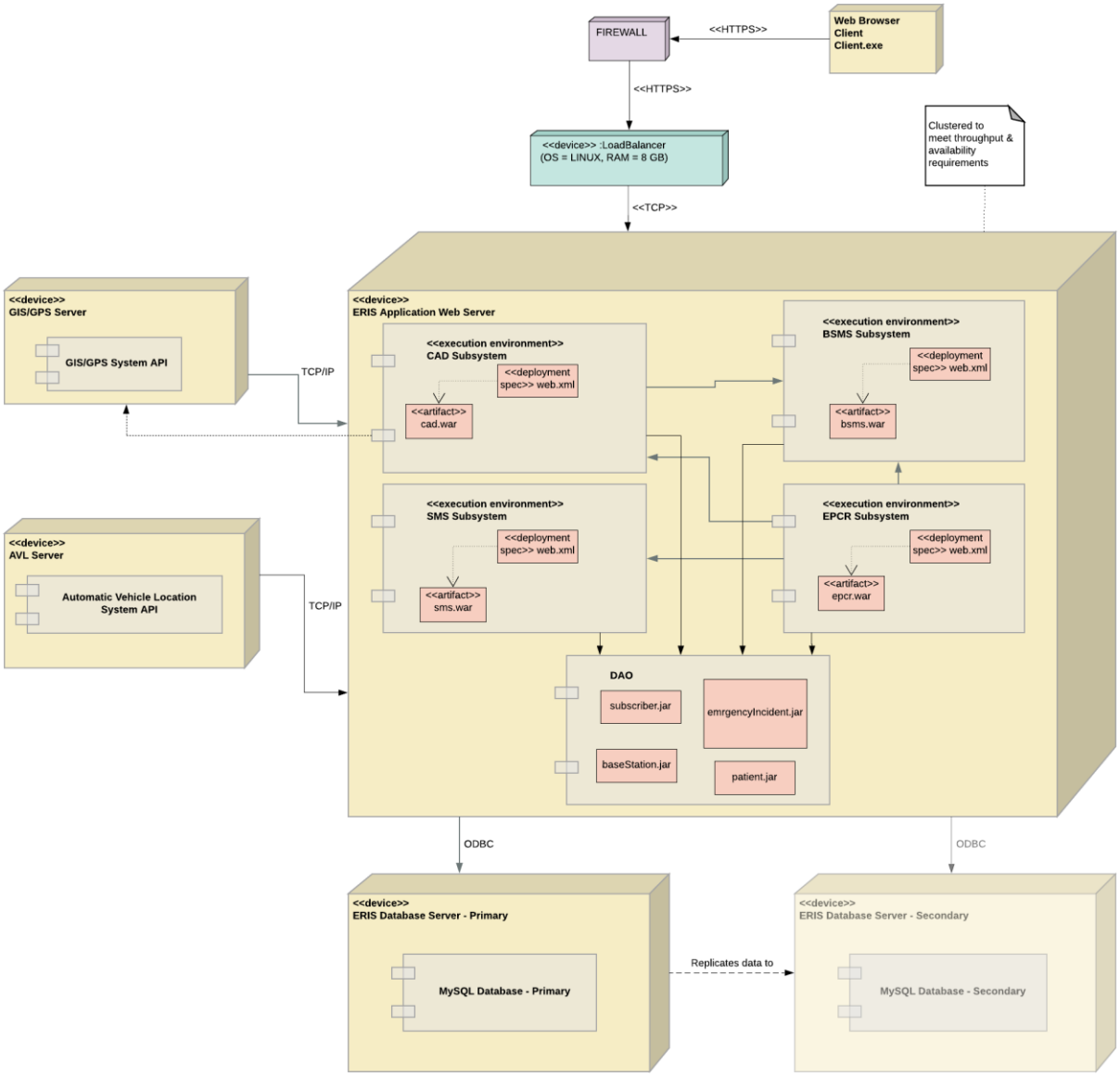
IV. EMT Clock-in



V. Record Patient Care Data



Deployment Diagram



Security Plan

Introduction

The purpose of this security plan is to detail the security requirements that the created Emergency Response Information System (ERIS) requires. This security plan will assign responsibilities and specify the management, operational, and technical controls needed to protect the system. As this system is utilized in the healthcare industry, patient data must be adequately protected under the Health Insurance Portability and Accountability Act of 1996 (HIPPA). Additionally, an entire county relies on the ERIS to dispatch emergency services to emergencies in a timely manner, so availability of this system is an element that needs to be ensured with security controls.

1. System Identification

1.1 System Operational Status

The system's operational status is currently under development. This means that the system is being built and security requirements specified into this document will be implemented into the system throughout the systems development lifecycle.

1.2 General Description/Purpose

The purpose of this system is to support the entire emergency response process and all of the associated activities. One of the functions of the system is the recording of emergency calls as they come in. Another is dispatching EMT teams to an emergency and tracking their activities

and location throughout their shift. Additionally, the system must be able to send relevant patient information to hospitals, and to manage billing of subscribers to the service and other users.

1.3 System Environment

The system will be connected to the Internet. Users of the call center system will be able to remote login to the system from any location. The servers associated with the system will be hosted with Amazon Web Services (AWS) as their SLA ensures 99.99% uptime ("Amazon Compute Service Level Agreement," 2018).

1.4 System Interconnection/Information Sharing

The system will be connected to the internet, as mentioned before, and will be connected using the TCP/IP protocol. Being connected to the Internet raises some concerns that must be mentioned. As the system allows remote logins by the operator user type, there are concerns that if credentials are compromised through social engineering or other means, that the system can be vulnerable to outside attack. This will need to be mitigated by two-factor authentication.

1.5 Laws, Regulations, and Policies Affecting the System

As the system collects healthcare information on patients at emergencies, there is a requirement to protect the information under the Health Insurance Portability and Accountability Act of 1996 ("Summary of the HIPAA Security Rule," 2013). Additionally, as an emergency services operator, Rescue911 must abide by local state regulations, such as the updated 2017 TX H 3640, which is a bill in Texas that requires emergency calls to a public safety answering point be confidential with the exception of first responders, investigators, and legal representatives of the person who made the call ("An Act Relating to the Confidentiality of an Emergency Call," 2017). The Payment Card Industry Data Security Standard (PCI-DSS) must also be considered in respect to the processing of payments from direct subscribers of the service. PCI-DSS is not

mandatory to comply with, but it is an important consideration as many states have regulations that are similar to PCI-DSS (Young, 2009).

1.6 Sensitivity of Information Handled

The system tracks a wide breadth of information and data that have different levels of sensitivity and criticality. Employee and shift information for EMTs and operators will be stored. Payment information for insurance providers and direct subscribers to the service will also be stored. Additionally, patient information and emergency information will be handled by the system. As mentioned above, as required by Texas state law, emergency call information must be confidential with the exceptions specified. Again, HIPPA compliance is vital for the system to protect medical information.

1.7 General Description of Sensitivity

The three elements of the cybersecurity triad (confidentiality, integrity, and availability) must be considered in the development and operation of the system.

1.7.1 Confidentiality

This system stores patient information that must be protected under HIPPA. The system also handles emergency call information, which is confidential under TX H 3640 ("An Act Relating to the Confidentiality of an Emergency Call," 2017). All payment information stored within the system must also be protected under PCI-DSS.

1.7.2 Integrity

In addition to being kept confidential, the integrity of payment information must be maintained and prevented from being modified by unauthorized users. Additionally, the integrity of the EMT shift information should also be protected from being changed or deleted.

1.7.3 Availability

The most critical aspect of the system that needs to be considered is availability. As this system supports the emergency response for an entire county, it needs to be available at all times. Unplanned downtime can result in critical emergencies not being responded to in a timely manner, which can result in deaths and can exacerbate patient illness and injury.

2. Management Controls

2.1 Risk Assessment and Management

The organization will perform a risk assessment on the system every year to determine the developing threat landscape, what assets need to be protected, and how risks will be mitigated.

2.2 Rules of Behavior

There will be several different types of users that will have varying levels of access and authorization to use the system. The call center operators in the call center will only have the ability and access to record calls, search existing calls, link new calls to existing calls, and dispatch EMT teams. They will not have access to the patient information that the EMTs record once on the scene of the emergency. Supervisors at the call center are a type of operator but have escalated privileges, in that they can dispatch a team to waiting calls and cancel an emergency response. EMTs have access to the emergency information of emergencies assigned to them,

ability to clock in and out using ERIS, and input details of the patient they treat at the emergency. However, they will not have access to patient information that they or a member on their team have not recorded. Finally, managers have full read access to data that the system generates, and can produce reports based on any number of criteria. Managers have write access to the master file data and have the responsibility to maintain it.

All users of the system are accountable for creating unique, strong passwords to access the system. Two-factor authentication will also be utilized for all users to build another layer of security, especially for the call center operators who are accessing the system remotely from any location. A clear segregation of duties matrix should be detailed to enforce the responsibilities and access of all different types of users in the system.

2.3 Planning for Security Throughout the Systems Development Life Cycle

The security requirements for this plan have been defined at the beginning of the project lifecycle. These security requirements are based on both the need for compliance with federal and local regulations and protecting business critical operations. However, these requirements are subject to expansion given the rapid pace that threats tend to develop. Any changes to this security plan will be assessed at the year-end risk assessment. This will give the organization a clear idea of the vulnerabilities in the system and new threats that need to be protected against.

In the implementation phase of the project lifecycle, penetration testing will be performed to determine common vulnerabilities that the system might have. Before the system is deployed into operation, a design review and relevant testing will be performed such as integration testing and regression testing ("System Security Plan," 2003, p. 17).

Once the system has been successfully deployed, there will be a team that is dedicated to doing regular backups of the data collected by the system, ensuring proper identity access management, and monitoring the daily operations of the system.

2.4 Plan for Eventual Disposal Phase

No system will last forever, and when the system is going to be taken offline and replaced, there must be secure data disposal practices in place. The system will store patient information throughout its use, and under HIPPA this information must be considered confidential and disposed of properly. Any devices that may have held information can be sanitized to the degree that the organization deems acceptable. A couple of options are overwriting, destroying, and degaussing the device (Violino, 2012).

3. Operational Controls

3.1 Personnel Security

The biggest risk to a system being compromised and confidential information being breached is people. Social engineering can be utilized to gain credentials to a system and compromise information. The system is connected to the Internet brings about several risks that must be accounted for. Because of this fact, all employees that will utilize the system will be screened by way of a background check before being hired. Additionally, authorization and access to the system will be strictly enforced. When employees are fired, their user accounts will be disabled while they are being informed of the firing.

3.2 Physical and Environmental Protection

Physical security will need to be accounted for in terms of the base stations. Access to the ERIS system will be available to managers at the base stations, so physical security must be enforced. Card keys will be required to gain access to the office. Tailgating will not be tolerated, as to mitigate the risk of physical security being compromised.

3.3 Contingency Planning

A contingency plan will be in place. A mirror of the production server will be running in parallel to the production server. These servers will be brought into sync at the end of each business day so that in the case that the main server goes down, the data loss is minimal. There will be drills every six months to test that the organization is ready to shift to the mirrored backup of the system. This will be essential so that in the case of an actual problem, workers know exactly what needs to be done.

3.4 Maintenance Controls

Any updates to ERIS will be required to be documented in order to ensure version control. Any changes must also be tested for impact to the usability and security of the current system before rolling out.

3.5 Data Integrity/Validation Controls

Data integrity is very important to the daily operations of this system, both in terms of the EMTs needing the most up to date information, and the patient care information being stored to be correct for billing purposes. There are a number of items that will be in place to protect data integrity.

Virus protection will be in place on all users' computers, including the operators who may be working from home. Automatic virus scans will be performed weekly during non-peak

hours. In order to ensure data entered into the online-based system, there will be data validation in place within the interface to ensure that the system is not vulnerable to SQL injection or cross-site scripting.

3.6 Documentation

Throughout the systems development life cycle, documentation will be generated for the system. All decisions when it comes to software and hardware will be documented. All tests run on the system and the results of these tests will be retained.

3.7 Security Awareness and Training

There will be mandatory awareness training for any employee who comes into contact with ERIS. The system will be dealing with healthcare data that is protected under HIPPA, so it is vital that all users know the importance of following all procedures. Topics covered will include password creation, phishing, social engineering, and general rules of behavior.

4. Technical Controls

4.1 Backup Procedures Using Amazon Web Services (AWS)

There will be a backup production server that mirrors the running system hosted on Amazon Web Services. This backup system will be tested every six months to ensure that the organization can easily shift to the backup. In the Service Level Agreement (SLA), Amazon guarantees an uptime percentage of at least 99.99% (“Amazon Compute Service Level Agreement,” 2018). This type of high availability is essential to ERIS, as it operates in a fast-paced healthcare environment that requires very minimal downtime.

4.2 Identification and Authentication

In assigning access to users of the system, users will only have access to the parts of the system essential to them performing their job duties, and no more. Users will also be required to have a password of at least 8 characters that includes uppercase, lowercase, numbers, and symbols in it. The users will be instructed to not reuse old passwords and will be prompted to upgrade passwords every year. Policies will be in place for users to reset their passwords, as a number of unsuccessful login attempts will result in the account being locked. Two-factor authentication will be utilized for users accessing the system in addition to a password. Users can be authenticated in one of three ways: something they know, something they have, or something they are (Schneider, n.d.). Examples of methods for two-factor authentication can be biometrics like a fingerprint, a smart card, or a token.

4.3 Authorization/Access Controls

There are certain standards for firewall configuration that the Payment Card Industry Data Security Standards (PCI-DSS) requires. There are several main conditions that must be followed in terms of firewalls for each Internet connection, "the network diagrams for associated cardholder data environment are to be consistent with firewall configuration standards," and "authorized personnel are to regularly review network configuration documentation" to ensure these firewalls are in place ("Firewall Requirements Policy," 2018).

The operators will be remote users of the system and access the system via a web browser from any location. With remote users comes the risk that unauthorized users may gain access to the system. There will need to be controls in place in order to mitigate the risks that accompany remote users. When users log in remotely, a standard login banner will be displayed. Login banners are useful as they define the organization's acceptable use policy, the fact that the system is constantly monitored for suspicious activity that violates the acceptable use policy, and

that users should not expect any privacy when using the system ("System/Network Login Banners," 2014). This login banner can be integral in prosecuting users who were not authorized to use the system as it warns them that they are not allowed to use the system ("System/Network Login Banners," 2014).

4.4 Audit Trails

In order to track the activity of all users of a system, there need to be several measures in place in order to track the activity. This is vital when it comes to keeping all individuals accountable for their behavior while using the system ("System Security Plan," 2003, p. 45). Additionally, keeping a log of all activity on the network allows for detecting potential intrusions and identifying potential security situations ("System Security Plan," 2003, p. 45). Controls should be in place to record several criteria of events including the type of event that happened, when, the associated user, and the affected system ("System Security Plan," 2003, p. 46). The audit log will be monitored weekly by Rescue 911's internal security team.

Test Plan

Introduction

Testing is an integral part of the software development process. There must be a dedicated quality assurance or testing team to test various functionalities and modules within the application. Exhaustive test plan and test cases must be developed beforehand for smooth implementation of the testing phase. Testing is required for performance measure for the application system. Various system aspects like performance, security, and quality are taken into consideration. With this robust testing plan, we will ensure that no hurdles are encountered during deployment and there is no ambiguity within the system.

Scope of Testing

As ERIS system follows CI/CD, test automation is essential to ensure careful completion of the application. Hence, it is imperative to have an automated test scheduled after every build cycle. The test results and reports should be generated and sent to important stakeholders so that necessary steps can be taken accordingly.

All the subsystems below should be tested individually:

- CAD - Computer Aided Dispatch Subsystem
- BSMS - Base Station Management Subsystem
- SMS - Subscriber Management Subsystem
- EPCR - Electronic Patient Care Reporting Subsystem

Interfaces between the above subsystems should be tested. Interfaces between the subsystems and the external systems that i.e. GIS and AVL should be validated.

Some of the main objectives to design the test plan are:

1. To test the ERIS functionality
2. To validate interfacing between the subsystems
3. To test the response time for dispatching an emergency call
4. To test the response time to access the BSMS to clock-in and clock-out
5. To test the system when the system is loaded with 50+ logged in operators at a time
6. To test the system when trying to simultaneously access to EMT dispatch

Purpose of Testing

The purpose of designing a test plan for ERIS is as follows:

1. To implement an error-free Rescue 911 system.
2. To implement a Rescue 911 system that is scalable and reliable enough to address the emergency requirements.
3. To create a system that can address requirements for multiple types of emergencies using a common platform.
4. To design a system that can be used easily by the users.

Unit Testing

TEST OBJECTIVE

Unit testing should be performed for all the modules of the code developed for ERIS. The purpose of unit testing is to ensure that the individual chunk of code performs as expected.

TEST APPROACH

1. Unit test cases and test scripts to be designed by the developer.
2. PHPUnit and JUnit Framework will be used to validate the Unit test cases.
3. Once the test cases are executed, the code coverage report should be generated.
4. 100% code coverage is the acceptance criteria so that every line of code is tested at least once.

TEST CASES FOR TESTING THE USER STORIES

1. Test Case 1

Test Id: TC001		Test Priority:
User Story: As an operator/supervisor, I want to log details of emergency calls received like the name of the caller, address, phone number, nearest landmark, and description so that appropriate action could be taken for the emergency.		
Acceptance Criteria	Test Steps	Expected Result
Only authorized operators are allowed to log emergency details	1. The operator will input the user ID in the Username field 2. The operator will input a password in the password field 3. Check the authorization status of the operator if operator can access the system or not	1. On successful authorization, operator will gain access to ERIS system. 2. If authorization failed, it will prompt the individual to input the credentials again.
A new emergency	1. The operator will take the details from	1. The incident will be

incident can be created specifying: name of the caller, address of the emergency, phone number, nearest landmark, emergency description	the caller 2. These details will be filled in the respective fields by the operator 3. An emergency incident will be logged once the operator submits the caller request	reflected in the system on logging the emergency.
Once the above details are submitted, and emergency ID will be generated for the call	1. The operator submits the caller's details 2. Once the request is submitted, check for the emergency ID	1. Emergency ID will be created once the incident is logged.
Emergency ID should be an auto-generated GUID	1. Upon generation of emergency ID, check the emergency ID 2. The emergency ID must be auto-generated and random and there should be no human intervention during emergency ID generation	1. The emergency ID will be system generated and hexadecimal
The status of the emergency call will be marked as "Logged"	1. On submission, enter the emergency ID in the status field of the call 2. Check the status of the call	1. The most recently added emergency incident in the system will have the status as "Logged"
The operator should be able to see the call details while querying with the emergency ID	1. Once the details are submitted, check the emergency ID generated 2. The emergency ID generated will be entered in the emergency ID field to check the details. 3. Press the submit button and check the details of the caller and emergency	1. The system will output the details of caller such as name of the caller, address of the emergency, phone number and incident details like nearest landmark, emergency description 2. On inputing wrong emergency ID, system will prompt the operator to input correct ID.

2. Test Case 2

Test Id: TC002		Test Priority:
User Story: As an operator, I want to be able to assign a priority category to an emergency and number of response teams required to an emergency so that the properly qualified teams are dispatched.		
Acceptance Criteria	Test Steps	Expected Result
Display the description of the emergency so that the operator can refer the same while assigning priority and no. of teams	1. While the input boxes are provided to assign priority and no. of teams to an emergency call, verify that the description of the emergency remains on the screen for the operator to refer 2. ###Once the inputs are entered and saved, all the details of the call including priority, description and no. of teams should be displayed on the screen	1. Before assigning priority and number of teams required for the emergency, the system will output the description of emergency when queried with emergency ID 2.On querying wrong emergency ID, system will prompt the operator to input correct ID.
The operator should be able to input Priority Category and number of response teams corresponding to an emergency	1. Record the details provided during the emergency call 2. Once the call details are recorded, check that there is a provision to input priority and no. of teams required. 3. Once saved, the priority and no. of teams should also be displayed along with the call details if queried with the emergency ID	1. The system will output a webpage to input the priority category and number of teams required for the emergency 2. On querying with correct emergency ID, the priority, number of teams required and emergency details will be displayed.
Priority Category should be determined by the operator based on the emergency situation as Minor / Stable/ Serious/ Critical	1. Record the details provided during the emergency call 2. While assigning priority to a call, the valid values can only be minor/stable/ serious/ critical 3. The input method should not accept any other value	1. While inputting the priority for emergency, a drop down list will be provided with values-minor, stable, serious and critical
Number and type of response teams required would depend on the priority	1. Record the details provided during the emergency call 2. The system shall not automatically assign any values to priority and no.	1. The operator will input the values of priority and number of teams required as per his/her understanding and expertise.

category of the situation	of teams; the values should be manually entered by the operator	

3. Test Case 3

Test Id: TC003		Test Priority:
User Story: As an operator, I want to be able to dispatch one or more response team once the emergency details are recorded so that response team can take control of the situation.		
Acceptance Criteria	Test Steps	Expected Result
The operator can find the nearest base station based on the landmark and zip code using the GIS system	1. Check landmark and zip code among the details the caller shared 2. Enter these details in the GIS/GPS system and search for the closest base station	1. The GIS system will output the nearest base station ID based on the details submitted by the operator like zipcode and landmark 2. The GIS system will prompt the operator to input valid values in case of invalid entries
Based on the Base Station Id retrieved, availability of response teams is determined using AVL and base station records	1. Get the nearest base station ID from the GIS/GPS system. 2. Enter this base Station ID in BSMS 3. Check for the available EMT teams	1. The system will output the status of EMT teams of the nearest base station on querying with the base station ID. 2. The system will prompt the operator to enter correct base station ID on inputting wrong value
The operator should be able to select a number of response teams from the list displayed and assign the emergency ID to the response team	1. From the available list of response teams, the operator can input the number of response teams required for a particular emergency. 2. The operator will also input the emergency ID to the response team	1. An emergency team will be allocated to a particular emergency as per the operator
The operator can select multiple teams	1. From the displayed list of response teams, the operator can check for the	1. The system will give an operator with option of

of different grades based on the emergency requirement	grades of the response team 2. The selected team by the operator will be assigned to an emergency of different grades.	assigning multiple teams of different grades based on the emergency

4. Test Case 4

Test Id: TC004		Test Priority:
User Story: As an operator, I want to be able to follow up on the calls so that I can keep a track of the dispatched team and emergency.		
Acceptance Criteria	Test Steps	Expected Result
Operators can query emergency records based on the emergency status or emergency id	1. Enter the emergency ID or emergency status in the respective fields 2. Querying emergency ID will output the associated emergency and all the details regarding it. 3. Searching for the specific emergency status, will output all the emergencies which are associated with that emergency status	1. The system will output the all emergencies with respect to the emergency ID or emergency status queried by the operator. 2. On inputting wrong or invalid emergency ID or status, the system will prompt the operator to enter correct values
Operators can update the emergency status based on the input from the response teams	1. The inputs of the emergency are obtained from the response teams. 2. Based on the inputs, Operator will query that particular emergency ID in the system. 3. Check the initial emergency status assigned to the emergency 4. Update the emergency status accordingly	1. The system will provide an option of updating the emergency status of an particular emergency to a operator 2. On inputting wrong or invalid emergency ID, the system will prompt the operator to enter correct value

5. Test Case 5

Test Id: TC005		Test Priority:
User Story: As a supervisor, I should be able to deal with the emergency calls to which operators are unable to dispatch a response due to response team unavailability so that no emergencies go unattended.		
Acceptance Criteria	Test Steps	Expected Result
Display all the calls that are in "waiting" status	1. Record the details of various emergency calls and assign them a different status 2. Check that Supervisor can query based on the emergency status 3. Verify that the supervisor gets a list of all calls in "waiting" status	1. The system will output the list of emergencies with "waiting" status when supervisor queries for "waiting" in emergency status field. 2. On inputting wrong or invalid emergency status, the system will prompt the supervisor to enter correct value
Prioritize the calls according to emergency priority category	1. Record the details of various emergency calls and assign them different priority numbers and mark the status as "waiting" 2. When supervisor queries for a list of calls in "waiting" status, the calls should be arranged in decreasing order of priority	1. The system will output the waiting emergencies in descending order
The supervisor should be able to find teams in the field close to the emergency scene using GIS/GPS and track their real-time location via AVL	1. Pick a particular call from "waiting" status list 2. Using the zip code and landmark, find the nearest response teams using GIS/ GPS 3. Once the teams are identified, track their live location using AVL	1. The GIS system will output the nearest response team for a list of waiting emergencies 2. The AVL system will provide the real time location of nearest team.
If a supervisor is unable to dispatch a team, he/she should transfer the call to	1. Pick a particular call from "waiting" status list 2. Assuming there are no other teams found through GIS/GPS and AVL,	1. The call or the emergency is transferred to another service provider company in absence of any available

another emergency service provider company	check that the supervisor can transfer the call to another emergency service provider in the county 3. Verify that the call can be closed by the supervisor 4. The only Supervisor should be authorized to transfer the call to the external provider	response teams.
Once the other service provider confirms that they are attending the call supervisor should be able to change the status of the call to "closed"	1. Pick a particular call from "waiting" status list 2. Verify that the call can be closed by the supervisor after transferring it to another service provider and noting down the relevant details of the other provider	1. The call or emergency status will be changed to "closed"

6. Test Case 6

Test Id: TC006		Test Priority:
User Story: As an operator, I want to link a new call with the same emergency to an existing emergency so that there is a record of all calls received for an emergency and avoid duplication		
Acceptance Criteria	Test Steps	Expected Result
The operator is querying emergencies based on the zip code and landmark	1. Record a call with all the details and get an ID generated for it after saving the details 2. Record details of another call. Before saving/assigning priority, check if the call is related to previously registered emergency	1. The operator will check the existing emergencies when a new call is recorded for similarities to avoid duplication of same emergency
Display all the emergency IDs, addresses and descriptions of calls received during the day which has the same zip code and landmark	1. Record calls with same and different zip codes and landmark combinations. 2. Verify that the call can be linked to the existing emergency by increasing the count of the number of related calls to the previous/existing call	1. The new call will be linked to an existing emergencies and counter for that emergency will be incremented.

7. Test Case 7

Test Id: TC007		Test Priority:
User Story: As an operator, I want to be able to find the closest base station to the emergency location so that response teams can get the exact location of the emergency.		
Acceptance Criteria	Test Steps	Expected Result
Operator should be able to input landmark, address and zip code for an emergency	1.Check if the emergency description is related to any previously assigned calls. 2. Generate Emergency ID after making sure that the emergency is new and exclusive	1.System will output the details of the call along with emergency ID, description, priority and the number of teams required
Display the nearest base station using GIS/GPS systems	1.Make use of the zip code and landmark, find the nearest response teams using GIS/ GPS	1.All Base stationed are displayed in increasing order of their distance in miles from the emergency location
GIS/GPS system returns the Id of the nearest Base station	1.Check the distance between emergency location and the displayed Base station. 2. Check for the number of available teams and their qualifications	1.All the details of the nearest base station ID is displayed
Display teams that are available and on shift in the base station along with their qualifications and grade as per the priority assigned to the emergency by the operator	1. Check for team availability in the displayed base station 2. Match with the priority of the emergency 3. Check if the teams available are on shift and qualified 4. Make sure the teams available is/are more than teams required. If teams are not sufficient display the number of teams yet to be found	1.Display the nearest base station with all the details and also the next nearest base station for any extra teams required

8. Test Case 8

Test Id: TC008		Test Priority:
User Story: As an operator I want to update the status of an emergency as “waiting” if there are no teams available at the time so that there is a record of an emergency waiting to be addressed.		
Acceptance Criteria	Test Steps	Expected Result
A team that needs to be dispatched will be radioed by the operator	1.Team ID and contact information is obtained from the nearest Base station ID and details obtained there of 2.Most appropriately qualified team/teams among the available teams on shift is/are radioed	1.Teams average response times is displayed while the operator make an attempt to radio.
The team will be marked as non-responsive if no acknowledgement is received within 15 seconds during the first attempt and within next 10 seconds during the second attempt	1.Teams with least average response times are radioed first to ensure success in the first attempt. 2. Teams that do not respond within 10 seconds in the second attempt are tagged non-responsive 3.Next available teams on shift are radioed if any team is non-responsive.	1.System updates the average response time of both responsive and non-responsive teams 2. System updates the call status as ‘waiting’ when the teams are non-responsive and the next available teams are being radioed.

9. Test Case 9

Test Id: TC009		Test Priority:
User Story: As an operator I want to be able to change the status of emergency call in the system from “logged” to “actioned” so that there is record that the team responded has been dispatched.		
Acceptance Criteria	Test Steps	Expected Result
A call whose status has been changed to “Actioned” will not appear in any other list pulled out by status field (like “Logged” / “On scene”)	1.Query for the Team ID of the team dispatched. 2. Make sure appears on the “Actioned” list 3. Query for other lists and assure the teams exclusion from those lists.	1.System displays the call status “actioned”

10. Test Case 10

Test Id: TC010		Test Priority:
User Story: As an operator I want to be able to change the status of the EMT in the system from “available” to “away” once the dispatch team confirms via radio so that there is a record that a team member is unavailable and proceeding towards an emergency location		
Acceptance Criteria	Test Steps	Expected Result
A team member’s status cannot be changed to “away” when the team member has not checked-in.	1.Check for the status of team member.i.e whether the team member has clocked into the system or not. 2.Query for “actioned” and check if all team members’ status of all teams in the list is “away”	1.All the team members in the “away” list are also in “checked-in” list when queried.

INTEGRATION TESTING

TEST OBJECTIVE

Post unit testing, all the modules are integrated and tested as a single interface. The objective of integration testing is to check the interaction between different units or modules of the system and identify any potential gaps between them.

TEST APPROACH

1. As agile is followed to develop ERIS, in a continuous integration environment integration testing should be triggered after every build cycle.
2. Mockito Framework will be utilized for integration testing.
3. Test suites should cover all the possible subsystem interfaces as well as the integration of modules with the subsystem.

TEST CRITERIA

Test Case ID	**Test Objective**	**Steps**	**Expected Result**
ITC001	Test the interface between CAD and SMS	Search with a subscriber id	Return the subscriber details
ITC002	Test the interface between CAD and GIS	Search the nearest base station based on the landmark and zip code	Base Station Location with Id is provided
ITC003	Test the interface between CAD and AVL	Search the live location of EMT team based on the Id	EMT team information with location is provided
ITC004	Test the interface between BSMS and CAD	CAD subsystem allots the EMT with an emergency incident	BSMS records for EMT status are updated for corresponding EMT

ITC005	Test the interface between EPCR and CAD	CAD subsystem can interact with the EPCR based on a emergency id to find the corresponding patient details	Patient data is retrieved from the EPCR and displayed

SYSTEM TESTING

TEST OBJECTIVE

After integration testing, the end-to-end system-wide testing is performed to check the functioning of the entire system. System testing is performed in an environment which is similar to the production environment. This is the first instance in the testing process where the whole system is tested as one unit.

TEST APPROACH

1. Test cases to be designed for data-driven testing to validate the functionality of the entire system.
2. Selenium used to test all the system modules and components.
3. Automated test suites and scripts to be developed.
4. Test cases for expected scenarios and exception scenario must be included.
5. Results and reports should be recorded.

FUNCTIONAL TESTING

TEST OBJECTIVE

Functional testing is performed to test the functional aspects and requirements of the system. The functionalities are tested with respect to the technical and business requirements specified in the case. It tests the behavior of the system with respect to inputs given to the system.

TEST APPROACH

1. Functional test cases should be designed based on the business requirements.
2. Defect reports should be generated

STRESS TESTING

TEST OBJECTIVE

Stress testing is performed to check the endurance of the system. Stress testing gives us a clear and precise understanding of the system under heavy traffic. Through this testing, the performance of the system can be checked under heavy load conditions and potential failure or breakdown points can be analyzed.

TEST APPROACH

1. JMeter test scripts are to be designed to test the APIs in case of heavy traffic.
2. Test results and reports to be captured and logged.
3. In case of issue, identify the component causing the problem.

PERFORMANCE TESTING

TEST OBJECTIVE

Performance is one of the critical requirements for Rescue 911 system as it is an emergency management system and the delay in the processing could result in undesirables outcomes. Performance testing is used to test the efficiency, loading time, and availability of the system. These metrics are highly important so that performance can be effectively evaluated.

TEST APPROACH

1. JMeter scripts to be designed for all the modules for which performance test is required.
2. Based on the results, the response time of each module can be identified individually.

USABILITY TESTING

TEST OBJECTIVE

Usability testing mostly deals with design and user interface of the system. This is a continuous process where users and various stakeholders repeatedly interact with the system, evaluating for simplicity and easy navigation throughout the system. The main difference is that usability testing is performed by the users and stakeholders and not by the testing or project team.

TEST APPROACH

1. A tool like “TryMyUI” can be used to test the usability of the application.
2. It should involve multiple platform cross-browser testing.
3. The feedback should be utilized for making necessary improvements.

ACCEPTANCE TESTING

TEST OBJECTIVE

UAT or User Acceptance Test is final testing which is performed by the users before handing over the application to the customer. UAT is performed in dedicated production environment wherein customer ensures that all the business and technical requirements are satisfied.

TEST APPROACH

1. Customer satisfaction test cases are to be identified and test scripts are to be designed.
2. Once all the tests are satisfied, the system is to be tested on a dedicated production server environment
3. All the related test suites, test reports, and identified bugs are to be tracked in the system

REGRESSION TESTING

TEST OBJECTIVE

Regression Testing is essential to ensure that any modifications do not impact the existing application.

TEST APPROACH

1. The automated regression test suite should be created.
2. After every build cycle, a regression script should be executed.
3. Regression script should include the prioritization of the regression test cases.

Scrum Essay

Scrum is one of the most popular agile methodologies used in the information technology industry. Scrum provides flexibility and transparency in project development and provides an opportunity for continuous improvement throughout the project. We applied the scrum approach to develop the deliverables for the Rescue 911 Emergency Response Information System (ERIS). The scrum approach proved very useful in achieving the desired outcome of the project. Scrum provided abundant resources to iterate and improve the final product and helped us achieve the desired goal efficiently.

Dividing the project into sprints helped to achieve project objective with the desired product increment at the end of each sprint. Sprints helped us to break down the entire project into smaller subcomponents with a predefined set of deliverables and expectations which were agreed upon by the entire team. The team decided to meet at the beginning of each sprint to plan out the list of tasks or stories to be picked up and completed at the end of the sprint. All stakeholders were required to attend the sprint planning meeting and every team member's input was considered before finalizing the list of stories for the sprint.

One of the most crucial parts of the scrum methodology is the daily standup meetings. Daily standup meetings help all team members to keep abreast of the latest developments in the project and makes it easier for the product owner to keep track of the updates and advancements in the project. We conducted our daily standups four times a week which proved useful in keeping development on schedule and make sure they are no obstacles or blockers. Daily standups also helped us to know each team member's workload and bandwidth available for the next story at the top of the product backlog.

The sprint demo is a way to showcase the progress made in the project and get an

approval from the product owner to move forward with the changes. We received some valuable feedback from Dr. Gomillion during our demo presentations which helped us to improve the quality of deliverables and brought our attention to requirements that were missed in the first draft of our stories. We also conducted sprint retrospectives at the end of each sprint. The retrospective was a good way to analyze the sprint completed and look for ways to improve the team's efficiency in the future sprints.

Scrum also provides various artifacts which help in the smooth progression of the project. The scrum board was on display at every daily standup and helped us to easily track the completion status for each story or subtask currently being worked on. The scrum board also provided a concise view of the sprint and every team member had a clear picture of current progress in the sprint. A burndown chart was widely used in the project to track the amount of work completed each day, and a velocity chart helped us to estimate the amount of work we as a team can complete in a sprint. Typically a velocity chart will stabilize as the team works together more, but since we only completed four sprints, the velocity chart increased every sprint. If the team had two to three more sprints, the velocity chart would have stabilized.

Overall, the scrum approach used by the team throughout the design of Rescue 911's ERIS was very successful. Scrum enabled us to divide the entire project into modules which were then worked upon iteratively, using priority to establish the order of stories worked on. Scrum provided us with a framework which we could use and modify to fit our requirements, thus providing flexibility to the overall project development. The iterative approach allowed us to focus on the current tasks at hand and not worry about the future modules of the project. Another big advantage of scrum is that it provided us with the option to develop and test the

changes in parallel which resulted in the team having more confidence that the project was moving in the desired direction.

Agile Essay

Agile is an adjective used to define someone who moves quickly and is light-footed ("Agile," 2018). In today's fast-paced world, it is imperative for everyone to be agile to stay ahead of the competition. On similar lines, even the IT industry is looking to catch up with the never-ending customer requirements, change requests, and system proposals. To satisfy these requirements in addition to completing projects within-budget, on-time, and to meet the quality standards, we need a robust framework to accomplish such demands. Agile methodologies provide the needed platform. There are various agile methodologies that use an incremental and iterative nature which provides teams with a competitive advantage over the traditionally used waterfall process. The various agile methodologies that are being compared are Scrum, Kanban, Rational Unified Process (RUP) and Extreme Programming (XP).

Scrum is one of the most widely accepted and used agile methodology. We have used the Scrum framework for the completion of the Rescue 911 case. The scrum approach reduces the amount of required documentation and focuses more on communication through regular team involvement. The team involvement is carried out through scrum rituals like sprint planning, daily standup meetings, sprint demo, and sprint retrospective. These rituals keep the team abreast of the project progress and help in identifying the pending list of tasks in the product backlog. Through these rituals, we learned our areas of expertise and drawbacks, analyzed and worked on them and tried to incorporate these points in future sprints. The artifacts, which are also an integral part of the scrum process, are the tools which helped us manage the workload in the scrum. The artifacts like product backlog, sprint backlog, and scrum board list all the functionalities, requirements, and user stories that had to be implemented throughout the sprints. The velocity chart and burndown chart helped us to identify how the team was progressing both

in the sprint as a whole and each day of the sprint. The burndown chart also showed the pending work that needed to be completed before the sprint ended. Scrum helped us to provide the deliverables on-time as it is highly timeboxed which helped us to maintain a fixed timeline for the project. As a team, we considered scrum as a feedback-driven approach wherein we got feedback at every stage like in the daily standup meetings, sprint demo, and the sprint retrospectives. Communication plays an integral role in the success of a project which makes scrum as the most preferred methodology.

Kanban is another agile methodology but is less popular compared to scrum. Kanban is less structured and more flexible with respect to scrum. The integral part of Kanban framework is Kanban board and Kanban cards (Krush, 2017). The Kanban board is similar to the scrum board. Kanban cards are placed on Kanban board which consist of work items traversing through different phases before completion ("Everything You Need to Know About Kanban Cards," n.d.). The main use of the Kanban board is to track the progress a task throughout the project. The focus of Kanban is to reduce multitasking. (Krush, 2017). This is achieved by limiting the number of Kanban cards in the "In Progress" phase (Krush, 2017). The main difference between Scrum and Kanban is that there are no sprints in Kanban, and Kanban is a continuous process with no timeboxing and no explicit team involvement activities unlike in scrum with rituals like daily standup and sprint retrospectives (Krush, 2017). In Kanban, the team must utilize self-made feedback loops in contrast to scrum where standups, demo, and retrospective help to enforce constant communication (Krush, 2017). Also, in Kanban, there is no dedicated role like scrum master to coordinate all the activities in the sprint which makes Kanban less structured (Rehkopf, n.d.). The structure of the methodology is compromised for the flexibility in Kanban. In Kanban, to take up a specific task which is recently added into the list of tasks, we do not have

to wait till the end of the sprint like in Scrum (Krush, 2017). Also, due to the absence of velocity chart and burndown chart in Kanban, it is difficult to track the pace of work and to estimate the completion time for the project. Instead of velocity and burndown charts, Kanban utilizes tools like cumulative flow diagrams and work in progress limits to monitor the progress of work items (Rehkopf, n.d.). The team would not have done well if we had opted for Kanban as a development methodology for this case. Most likely this would have led to untimed and delayed deliverables due to lack of timeboxing. Also, the activities and tasks would have been significantly more disorganized since there is no scrum master to guide the team members.

Extreme Programming, abbreviated as XP, is another popular agile methodology which focuses on improving the result of the software development process and testing procedures (Green, 2016). Extreme programming includes various practices which help to deliver products within predefined timelines and of the highest quality ("Extreme Programming," n.d.). This approach emphasizes system design, coding, and testing (Rasmusson, n.d.). Initial planning is conducted which focuses on the desired features and functionalities in the prioritized order (Rasmusson, n.d.). Extreme programming is an iterative approach in which the system is implemented in small releases with pre-decided features (Rasmusson, n.d.). It focuses on optimization with no duplication of code, simple design of the system and clear and precise test cases (Rasmusson, n.d.). This is a collective team effort where all the members are responsible for changes in code or system. The major practices followed in extreme programming are pair programming, continuous system integration, and 24/7 customer presence with the project team (Rasmusson, n.d.). As a team, there were very few instances where pair programming was used with the Scrum methodology. Two resources were allocated to identify and draft user stories and design sequence diagrams. This helped us to understand the advantages of pair programming

which improved the efficiency and throughput. This resulted in a few mistakes and minimalistic rework requirement. But there are also downfalls of extreme programming. Implementing extreme programming would have led to the continuous presence of the customer with the project team. This adds pressure and stress to the team that would not exist in Scrum because the user and customer involvement in extreme programming is more than in scrum.

The Rational Unified Process is often abbreviated to RUP. It is considered a software development process, process product, and a process framework (Kruchten, 2004). RUP is divided into four phases across predefined software development activities ("RUP - Rational Unified Process," n.d.). These four phases are Inception, Elaboration, Construction, and Transition ("RUP - Rational Unified Process," n.d.). RUP is an iterative process functioning in these four phases ("RUP - Rational Unified Process," n.d.). On the outlook, RUP looks to share some characteristics of the traditional waterfall methodology (Green, 2016). The teams following traditional waterfall methodology who want to follow more of an iterative approach can utilize RUP as it combines the advantages of the waterfall into an iterative process (Green, 2016). RUP demands high levels of skills and expertise to implement this methodology successfully (Krebs & Shuja, 2008). In RUP, customer interaction is the sole source itself to make improvements which make customers and stakeholders feedbacks very critical for the success of the project (Green, 2016). On the other hand, in scrum feedback is obtained from multiple sources at different instances like standup meetings, demo, and retrospective. This makes scrum a much-preferred methodology over RUP. If the team had utilized a RUP methodology, too much time would have been spent in trying to learn how to use RUP instead of completing the project. Scrum enabled the team to successfully produce functionality for every sprint, resulting in a fully-fleshed out design at the project end.

The differences between these methodologies are not glaring but each of the methodologies has its own significance. The choice of these which of these agile methodologies to use depends on the system being developed and how suitable is the methodology for a particular case. The selection of the methodology also depends on the team dynamics, bonding among the team members, and the comfort level of every team member towards a specific methodology. Scrum was the easiest methodology for a new team like ours to pick up and use without a severe learning curve. As a team, we found scrum to result in a very productive process and allowed us to draft the deliverables both within the pre-decided timeline and as per the quality mentioned in the project requirements and project case.

Agile vs Waterfall for Rescue911

In a genuine waterfall development project, phases like conception, requirement gathering, analysis, design, implementation, testing, and deployment constitute a distinct methodology of software development and the stages are usually rigidly sequential. We followed the agile methodology for our project which is an iterative and incremental model that accommodates for changing needs in the course of development. If we had taken the waterfall route for designing Rescue 911's Emergency Response Information System (ERIS), we would have compromised on the adaptability and quality of the system, overlooked mid-process feedback, and delayed the release of our deliverables.

Waterfall's major drawback is its hindered capacity to cater to changing needs. During the middle and end stages of our project development, we addressed many flaws in our user stories, class diagrams, and diagrams. Some of the flaws in the design were fundamental to the functionality of the system. As we were approaching the project iteratively this was accounted for on our plans and did not warrant a dramatic leap backward whereas the waterfall approach would have cost us a lot of time and investment on the project. Agile's short development cycles gave us the much-needed flexibility and adaptability.

At the end of each sprint, we made sure that one of the group members acted as the end user and gave feedback on the deliverables we had accomplished, which was valuable to improving these deliverables in the next sprint. This allowed us to orient ourselves in the right direction and shift our perspective in the course of the project. As testing takes place at the end of the project life cycle in the waterfall approach, if we had discovered that the users are not satisfied, it would have been really difficult to address that. Immediate user feedback ensured

that the project is guided by the high-level business requirements which diminished the risk that we developed a system that no one would want to use.

The most demanding part of waterfall is gathering of the requirements in an insightful way. As we have learned through class lectures this semester, in projects where requirements are completely straightforward, extremely clear, and well documented like in legacy systems, waterfall can be a very good approach to take. However, the case we were given did not have very clear requirements up front. The requirements continued to evolve over the course of the project, especially with the case extension given to complement the case. Going on with poorly gathered requirements would have led to insurmountable challenges at the end of the project and would have had dire consequences on the success of the project. Stakeholders, at times, are not perceptive enough to envision the whole system from a requirement document and detailing all the requirements up front may be an overwhelming endeavor. Sometimes, unless a working prototype is in hand, stakeholders won't know what they really want. The agile approach really helped us in conceptualizing the system iteratively until we actualized an evolved design of the system that met all business requirements.

Unlike waterfall, the need for speed is one of the basic tenets of the agile methodology. In waterfall, actual development doesn't begin until at least four phases of the project life cycle have been completed. A viable working product won't be in the picture until the late stages of the project. If we had taken waterfall route, it would have been ironic in that it would have clashed with the system being designed which by its very name is "emergency," which always require a rapid response. Having working prototype in a short developing cycle not only helped us visualize the end product but also catered to the needs of the client with a minimum viable system to address the dispatching of life-saving teams to emergencies.

Both agile and waterfall are feasible project development approaches which have stood the test of time. Waterfall can be apt in simple and predictable projects with well laid out requirements. It can provide stability with its sequential path which is clear, documented and disciplined by design. But for the design of a system, like Rescue 911's ERIS, which warrants flexibility, need for speed, and continuous feedback to evolve, the agile approach is the most suitable approach.

Scrum Master Experience Essay

I. Sprint 1 Scrum Master - Amit Yadav

Helming the first sprint of my team project as a scrum master was one of the best learning experiences throughout this semester. It allowed me to tap into my managerial potential that I had developed throughout my undergraduate engineering program and afforded me an opportunity to apply my classroom learning from my Advanced System Analysis and Design (ASAD) class as well as my Project Management class.

Just a couple of days into our first sprint, I soon realized the key differences between managing teams in any other agile framework versus scrum. Scrum enforces a participative and affiliative management style and trims down a manager's tendency to be autocratic and directive. I felt a reduced onus of managing the team; the team managed itself, my role was more of a facilitator and a counselor. Instead of focusing just on deliverables, my responsibility was to make sure that scrum practices were followed properly throughout the sprint.

As a Scrum Master, I felt that one of the biggest advantages of scrum was that it helped me organize a motley of tasks into sprints. We had two disparate sets of deliverables – the case deliverables and project deliverables, and keeping them organized altogether was a challenge in itself. Segregating each category of tasks in product backlog and sprint backlog proved very helpful.

I feel that what makes scrum effective is that team members pick each task according to their skills and I didn't have to assign tasks to people myself. Not only did it reduce my workload, but I was able to understand team members interests and abilities better. Instead of doing all the planning myself, the whole team gathered to ponder on prioritizing the stories and

assigning an effort point to each and including them in our first sprint.

Throughout our sprint, we made use of both a virtual and physical scrum board. So, it was really easy for me to track who was working on which task and the progress each individual had made. I had the pleasure of conducting our very first stand up. The initial few times, conducting a daily stand up felt a little uncomfortable because it was kept a little formal. This formality inhibited team members from speaking up. But these meetings were later toned down to short, casual gatherings which encouraged rather taciturn team members to open up and share the progress of their deliverable and communicate blockers if they faced any.

Based upon their availability, people volunteered to help each other accomplish their tasks, which significantly increase our team's productivity. Since it was our very first sprint, I could not estimate my team's capability to accomplish the story points and as a result, we ended up including fewer stories than we could have covered. But after the sprint was finished, I was able to analyze my team's capability through a burndown chart and a velocity chart which helped us to better estimate the number of stories to be picked for our next sprint.

All-in-all, I learned a great deal by spearheading our team project as a scrum master. It expanded my techno-managerial and leadership potential while also enhancing my understanding of the scrum framework. I now feel more confident working in a scrum based team and am excited at the prospect of working in a similar setting in my professional life.

II. Sprint 2 Scrum Master - Amrutha Girish

During my one week stint as the scrum master, I was in charge of conducting the following rituals: sprint planning at the beginning of the sprint, daily standups on all the days our team met, sprint demo and sprint retrospective at the end of the sprint.

As a scrum master, you need to come prepared for the sprint planning meeting than the rest of the team members. I read the project backlog thoroughly to pick out the stories that I felt should be included in sprint two. I also reviewed the sprint backlog from sprint one to understand if any of them needed to be modified and included as an extra story in sprint two. This required inputs from individual team members who worked on the tasks. It was a little hard to enhance the work from sprint one as it had to be assigned on top of the stories selected for sprint two. It was also tough to get the same person who worked on the story in sprint 1 to continue working on it. While picking out stories, I made a mental note that I should try and divide the tasks logically so that each team member can continue on what they previously worked.

The standups had become a routine affair. Conducting them was relatively easy though solving blockers wasn't always easy. During standups, I needed to know what the task a person was working on, how long it would take, and if they were facing any issues/blockers. Thankfully, everyone was on track despite facing a few issues here and there. Most issues we faced were with respect to consistency between diagrams. Since each individual was separately working on each UML diagram, it was a substantial task on its own to get the diagrams in sync. It was decided that for now, we would follow a naming convention for classes and methods and later, we would all sit together and get the diagrams consistent. The other problem that we usually had was where a person needed a helping hand to finish the work assigned to them. Most of them were legitimate and luckily, we had a few members who could manage it. So that was

always taken care of. During my tenure as a scrum master, we have never had to consult Dr. Gomillion for a question to be resolved.

In our project team, scrum masters were given the task of creating the sprint demo PowerPoint. I also created the sprint burndown chart and velocity chart. For the PowerPoint, I had to just arrange all the deliverables in a logical order so that audience of the demo can understand, inform the team, and include necessary pictures, links, and assumptions. During our last print, our velocity was pretty low and we knew that we could take up more story points. So during my tenure, I wanted to get an approximate velocity of our team and we did that successfully thanks to all the efforts of the team during sprint planning.

For the scrum retrospective, scrum masters did not have to do a lot of preparation as such but they had to be all ears to the conversation from team members. I had decided that all the feedback from the discussion was going into one of the three buckets: what went well, what did not go well, and how we could improve for the next sprint. Most of the improvements were byproducts of what did not go well. To this, I also communicated the feedback from Dr. Gomillion because those were also things that we missed or did not do correctly.

III. Sprint 3 Scrum Master - Kaitlin Wallace

I was the scrum master for the third sprint of our project. By this time in the project, everyone knew each other a lot better, and we were a lot more comfortable with each other. The task of being the scrum master was not as daunting as it might have been if I had the role in the first sprint. Like a true scrum team, our team was self-organized and very self-motivated to complete the given tasks. Sprint planning went by super quickly, as we knew what strengths people had and since this was the third sprint, we were better at estimating our effort points. In one of our first standups, we realized that we had no idea how to do the security plan, which was a blocker. None of the team members had ever worked in security before, and I had only had course experience in the subject matter. To alleviate this blocker, I delegated the job of emailing Dr. Gomillion for clarification to my team member Amrutha. The response to this email helped guide us in completing a comprehensive security plan with the important factor of availability considered in the design of the Emergency Response Information System (ERIS).

One of the things I struggled with was that I found myself often getting talked over by my more vocal team members in daily standups. Since I have never practiced scrum in an actual development environment, I am not sure if this is the norm, or if I should have been more assertive in conducting our daily standups. The problem really occurred when someone had a blocker. Everyone on the team was very excited about helping everyone else and figuring out how to solve their blockers. However, this led to a lot of noise and disorganization at the daily standups which often led to off-topic discussions.

I felt that as soon as I got the hang of being the scrum master, the sprint was over. It was actually a very fulfilling job to be the scrum master. One common complaint about the standups I led was that they went too fast. I went from person to person very quickly when asking about

what people were working on, and what blockers they had. I feel that I could have spent a long time discussing blockers. Only one blocker, the security plan, that was mentioned by a team member at a daily standup. However, I believe that there might have been additional blockers as when the team went to review the UML Sequence Diagrams, many of them were not done correctly or were not actually complete. This was a concern that was noted down for the next sprint, and I made an effort to be more vocal about voicing blockers. Overall, being the scrum master was a fun experience, and I wish that I had longer to be the scrum master because I feel like the team would have benefited from at least one constant in the project.

IV. Sprint 4 Scrum Master - Somya Sharma

I was the Scrum Master for the fourth sprint, which was also the last sprint for the project. Being the scrum master in this sprint was both exciting and a stressful experience. We had to pick up all the pending tasks and the enhancement stories which was a lot of work. It was a very crucial sprint of the project as we had to accomplish all the unfinished tasks as well as establish consistency across all the documents and diagrams for the design of the Emergency Response Information System (ERIS).

As a scrum master, I made sure to pick all the essential stories from the product backlog. During the sprint planning, I ensured that everyone had almost equal story points to work on. I assured that the sprint planning was time-boxed and that all the stories pulled from the product backlog to the sprint backlog were assigned with a developer and estimated story point. My job was to maintain the scrum board so that it clearly showed sprint progress. I also ensured that all sprint standups were within the timeframe and to the topic. Asking who is working on what, how much progress is made, and if they faced any impediments was the major focus of the sprint standups. As a scrum master, I had significant responsibilities to ensure that all the team members were collaborating together. I also tried to establish healthy team dynamics and enable everyone to achieve productive results. I had to coordinate with all of the team members to understand the progress of the stories as well as the blockers. Another major responsibility I had as the scrum master was to ascertain every member of the team had some task to work on throughout the sprint. Thus, ensuring all the members has been assigned equal effort was an imperative obligation as a scrum master.

Every member of the team was present for the standups on time. Everyone clearly stated the progress made on the stories and informed me if they had any blockers. This really helped to

have a transparent work environment. If any member was unable to pick any story or was still working on the previous one, I ensured the person who was available at that point could pick up the task so that we could keep progressing. Furthermore, the quality assurance deliverable I created helped the team members to track and review the progress on the deliverables. It also enabled me to ascertain that each deliverable produced fits the quality check.

We could have worked on prioritizing the tasks better. As in the last sprint we were dealing with consistency stories to ensure synchronization across all the diagrams which made the tasks a bit dependent on each other. I believe prioritizing the subtasks of the stories would have made work easier and faster for us. Facilitating the team and resolving the conflicts was my major responsibility and needed to expedite the progress of the project.

Though we did not have any official demo for the fourth sprint, we decided to present the deliverables to the Product Owner. We created all the final diagrams, documents, and asked the Product Owner to review them to identify if they satisfied the acceptance criteria. Based on the review feedback, we tried to fix a few glitches that were identified. The final project deliverable of the Consolidated Project Report was a major accomplishment made in this sprint which showcased our effort throughout the entire project.

Youtube Channel

Title	Link
Sprint 1 - Standup 1	https://www.youtube.com/watch?v=mGRgH_Zv2p8
Sprint 1 - Standup 2	https://www.youtube.com/watch?v=UChjHXHTQPY
Sprint 1 - Standup 3	https://www.youtube.com/watch?v=GqIUecXPo1U
Sprint 1 - Standup 4	https://www.youtube.com/watch?v=0OlMILilP8w
Sprint 1 - Demo	https://www.youtube.com/watch?v=NKSVpdclG0w
Sprint 1 - Retrospective	https://www.youtube.com/watch?v=LNu81lBl-cs
Sprint 2 - Planning	https://www.youtube.com/watch?v=yAB4-f-H8ZE
Sprint 2 - Standup 1	https://www.youtube.com/watch?v=S9ureOy2UW0
Sprint 2 - Standup 2	https://www.youtube.com/watch?v=3XbM-RkMj-A
Sprint 2 - Standup 3	https://www.youtube.com/watch?v=sSx9NHIbLMg
Sprint 2 - Standup 4	https://www.youtube.com/watch?v=sdVXOmWhic0
Sprint 2 - Retrospective	https://www.youtube.com/watch?v=_lgTPGZuQq0
Sprint 3 - Planning	https://www.youtube.com/watch?v=zEovYDPawc0
Sprint 3 - Standup 1	https://www.youtube.com/watch?v=puNApbUiDs0
Sprint 3 - Standup 2	https://www.youtube.com/watch?v=UT_EcyvfJcE
Sprint 3 - Standup 3	https://www.youtube.com/watch?v=hx3VIfKe_o8
Sprint 3 - Standup 4	https://www.youtube.com/watch?v=hjV1PASDYEc
Sprint 3 - Retrospective	https://www.youtube.com/watch?v=DujAW1Q60sE
Sprint 4 - Planning	https://www.youtube.com/watch?v=yTId13ZjpBk
Sprint 4 - Standup 1	https://www.youtube.com/watch?v=u2scjaFk-AI
Sprint 4 - Standup 2	https://www.youtube.com/watch?v=pG1D69f_tvo
Sprint 4 - Standup 3	https://www.youtube.com/watch?v=I9Gh6F8Hikg
Sprint 4 - Standup 4	https://www.youtube.com/watch?v=PEGFNBcwcoQ

Quality Control for Deliverables

Throughout the project all the deliverables were reviewed and approved by all the members in the team. Final status for any deliverable was typically among - "To Do", "In Progress", "Modifications Required", "To be reviewed" and "Reviewed and Approved". The status of the deliverable for each individual was changed from "Pending" to "Approved" after being reviewed. Final status of the deliverables was changed to "Reviewed and Approved" once signed off by all the members in the team.

Deliverables	Amit Yadav	Nishant Goel	Praveen Gadugin	Mihir Bhende	Amrutha Girish	Kaitlin Wallace	Somya Sharma	Final Status
Cover Page	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Executive Summary	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Case User stories	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Acceptance Criteria for stories	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
DFD Context Level	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
DFD Level 0	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
DFD Level 1 - CAD	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
DFD Level 1 - BSMS	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
DFD Level 1 - SMS	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
DFD Level 1 - EPCR	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Logical ERD	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved

Physical ERD	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Class Diagram	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Deliverables	**Amit Yadav**	**Nishant Goel**	**Praveen Gadugin**	**Mihir Bhende**	**Amrutha Girish**	**Kaitlin Wallace**	**Somya Sharma**	**Final Status**
Package Diagram	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Seq - Log calls	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Seq - Dispatch Calls	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Seq - Assign Waiting Calls	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Seq - EMT Clock in	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Seq - Patient Care	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Deployment Diagram	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Security Plan	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Test Plan	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Scrum Essay	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Agile Essay	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Agile vs Waterfall	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Scrum Master Experience Essay - Kaitlin	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Scrum Master Experience Essay - Amit	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Scrum Master Experience Essay - Amrutha	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Scrum Master Experience Essay - Somya	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved

In class Demos	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Extension Incorporation Details	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved
Consolidated Project Report	Approved	Approved	Approved	Approved	Approved	Approved	Approved	Reviewed and Approved

References

Agile. (2018). In the *Oxford Living Dictionary*. Retrieved from https://en.oxforddictionaries.com/definition/agile

An Act Relating to the Confidentiality of an Emergency Call, 2017 TX H 3640, Legislature of the State of Texas. (2017, September 1). Retrieved from https://custom.statenet.com/public/resources.cgi?id=ID:bill:TX2017000H3640&ciq=ncsl27&client_md=03c39788dcfec0f8e57f699b6d52048a&mode=current_text

Amazon Compute Service Level Agreement. (2018, February 12). Retrieved from https://aws.amazon.com/compute/sla/

Atlas Software Testing Plan Template. (2017, January 8). Retrieved from https://www.atlascode.com/wp-content/uploads/2017/08/Atlas_Software_Testing_Plan_Template.pdf

Everything You Need to Know About Kanban Cards. (n.d.). Retrieved from https://www.smartsheet.com/everything-you-need-know-about-kanban-cards

Extreme Programming. (n.d.). Retrieved from https://www.agilealliance.org/glossary/xp/

Firewall Requirements Policy. (2018, April). Retrieved from https://pcicompliance.stanford.edu/sites/g/files/sbiybj7706/f/1._firewall_requirements_policy_0.pdf

Green, S. (2016, February 11). Choose Your Project Management Methodology: Pros and Cons of Agile, Waterfall, PRiSM and more. Retrieved from https://www.workflowmax.com/blog/choose-your-project-management-methodology-pros-and-cons-of-agile-waterfall-prism-and-more

Krebs, J. & Shuja, A. K. (2008, June 26). Welcome to the IBM Rational Unified Process and Certification. Retrieved from http://www.informit.com/articles/article.aspx?p=1155863&seqNum=4

Kruchten, P. (2004, March 19). An Introduction to the Rational Unified Process. Retrieved from http://www.informit.com/articles/article.aspx?p=169549

Krush, A. (2017, November 14). Agile Framework Comparison: Scrum vs Kanban vs Lean vs XP. Retrieved from https://www.objectstyle.com/agile/agile-scrum-kanban-lean-xp-comparison

Rasmusson, J. (n.d.) Extreme Programming. Retrieved from http://www.agilenutshell.com/xp

Rehkopf, M. (n.d.). Kanban vs. Scrum. Retrieved from https://www.atlassian.com/agile/kanban/kanban-vs-scrum

RUP - Rational Unified Process. (n.d.). Retrieved from https://www.webopedia.com/TERM/R/RUP.html

Schneider, F. (n.d). Something You Know, Have, or Are. Retrieved from https://www.cs.cornell.edu/courses/cs513/2005fa/NNLauthPeople.html

Summary of the HIPAA Security Rule. (2013, July 26). Retrieved from https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html

System/Network Login Banners. (2014, January). Retrieved from https://security.tennessee.edu/login-banners/

* System Security Plan: Development Assistance Guide. (2003, April 1). Retrieved from https://www.sans.org/projects/ssp.zip

Violino, B. (2012, February 6). The in-depth guide to data destruction. Retrieved from https://www.csoonline.com/article/2130822/it-audit/the-in-depth-guide-to-data-destruction.html

Young, F. (2009, February 27). Is PCI Compliance a Law? Should it be? Retrieved from https://www.pcicomplianceguide.org/is-pci-compliance-a-law-should-it-be/

* This source was utilized as template to structure the security plan. When information is used from the template prompt, it is cited in text with the appropriate page number.