



BLUE HEDGEHOG GROUP A PLAN



Jake Whamond, John Deniel Perez, Kaleb Jones, Zane Baker and Ethan Makepeace

Table Of Contents

1. [Title](#)
2. [Tables of content](#)
3. [Group cover sheet](#)
4. [Introduction](#)
5. [Business rules](#)
6. [Business rules](#)
7. [Business rules](#)
8. [Business rules](#)
9. [Reporting and log subsystems and descriptions](#)
10. [Make report fully developed use case description.](#)
11. [Report Sensor alert fully developed use case description](#)
12. [Dispatch Coordination subsystem and description](#)
13. [Choose Dispatch developed use case description.](#)
14. [Search Dispatch fully developed use case description](#)
15. [Operator subsystem and description](#)
16. [Create Roster fully developed use case description.](#)
17. [Brief use case description](#)
18. [Brief use case description](#)
19. [Brief use case description](#)
20. [Activity Diagrams – Make Report.](#)
21. [Activity Diagram – Report Sensor Alert.](#)
22. [Activity Diagram – Create Roster](#)
23. [Activity Diagram – Choose Dispatch](#)
24. [Activity Diagram – Search Dispatch](#)
25. [Activity Diagram descriptions](#)
26. [Domain class Diagram](#)
27. [Domains Class list](#)
28. [Team management meeting 1 notes](#)
29. [meeting 2 notes](#)
30. [meeting 3 notes](#)
31. [meeting 4 notes](#)
32. [meeting 5 notes](#)
33. [meeting 6 notes](#)
34. [meeting 7 notes](#)
35. [meeting 8 notes](#)
36. [Teams Analytics](#)
37. [Conclusion](#)
38. [References](#)

GROUP ASSESSMENT ITEM COVER SHEET

Student Numbers:	Emails:	FIRST NAMES	FAMILY / LAST NAMES
3 3 2 7 7 9 4	C3327794@uon.edu.au	Jake	Whamond
3 3 9 0 6 0 6	C3390606@uon.edu.au	John Deniel	Perez
3 4 0 5 6 3 4	C3405634@uon.edu.au	Kaleb	Jones
3 4 0 4 9 8 3	C3404983@uon.edu.au	Zane	Baker
3 3 7 5 4 2 1	C3375421@uon.edu.au	Ethan	Makepeace

Course Code	Course Title
S E N G 2 1 3 0	System Analysis and Design
(Example) A B C D 1 2 3 4	Intro to University

Campus of Study: Callaghan (e.g. Callaghan, Ourimbah, Port Macquarie)

Assessment Item Title: Assignment 1 Part A: Requirements Modelling Due Date/Time: 31/03/23, 17:00

Tutorial Group (If applicable): Word Count (If applicable):

Lecturer/Tutor Name: Kieran Molnar

Extension Granted: ☐ Yes ☐ No Granted Until:

Please attach a copy of your extension approval

NB: STUDENTS MAY EXPECT THAT THIS ASSIGNMENT WILL BE RETURNED WITHIN 3 WEEKS OF THE DUE DATE OF SUBMISSION

Please tick box if applicable

☐ Students within the Faculty of Business and Law, Faculty of Science, Faculty of Engineering and Built Environment and the School of Nursing and Midwifery: We verify that we have completed the online Academic Integrity Module and adhered to its principles.

☐ Students within the School of Education: We understand that a minimum standard of correct referencing and academic literacy is required to pass all written assignments in the School of Education; and we have read and understood the School of Education Course Outline Policy Supplement, which includes important information related to assessment policies and procedures.

We declare that this assessment item is our own work unless otherwise acknowledged and is in accordance with the University's [Student Academic Integrity Policy](#)

We certify that this assessment item has not been submitted previously for academic credit in this or any other course. We certify that we have not given a copy or have shown a copy of this assessment item to another student enrolled in the course, other than members of this group.

We acknowledge that the assessor of this assignment may, for the purpose of assessing this assignment:

- Reproduce this assessment item and provide a copy to another member of the Faculty; and/or
- Communicate a copy of this assessment item to a plagiarism checking service (which may then retain a copy of the item on its database for the purpose of future plagiarism checking).
- Submit the assessment item to other forms of plagiarism checking.

We certify that any electronic version of this assessment item that we have submitted or will submit is identical to this paper version.

Turnitin ID: (if applicable)

Signature: <u>Jake Whamond</u>	Date: <u>31/03/23</u>
Signature: <u>John Deniel Perez</u>	Date: <u>31/03/23</u>
Signature: <u>Kaleb Jones</u>	Date: <u>31/03/23</u>
Signature: <u>Zane Baker</u>	Date: <u>31/03/23</u>
Signature: <u>Ethan Makepeace</u>	Date: <u>31/03/23</u>

DATE
STAMP
HERE



31/03/23

Introduction

In order to combat long standing issues with slow response and communication, the city of Nuvalis has decided to combine all emergency stations into a single, emergency control center called 'SSS'. In order to achieve this, careful consideration needs to be placed into the design of a whole new system designed to run every aspect of such a facility. The objective of this report is to design and provide an overall view of what this system could look like and ultimately propose its implementation. In order to account for the many different and frequently changing fragments of such a facility, the system would need to be extensive with a heavy focus on precision and attention to detail while simultaneously avoiding any obscurity.

The proposed system will run majority of SSS's functions such as the rostering of staff, the handling of emergency reports, and the dispatching of appropriate emergency workers. As much of the work done by SSS would involve the saving of lives, it is vital that there is no room for error with the system designed to account for and remain functional during any and all potential abnormalities as the failure of any part of the system could result in serious consequences.

In order to effectively convey the proposed system, this report will utilize a number of different lists and diagrams. Business rules covering work health and safety, legislation and standards, ethics, security, and privacy have been clearly established at the beginning of the report in order to provide users of the system with clear instructions on how to operate smoothly and without issue. Use cases also play a major role in the new system so use cases have been established with explanations, descriptions and diagrams for each one. The use cases that will be analyzed are the creation of a weekly roster, report alerts, the creation of a report and the dispatching of appropriate emergency workers. A domain class diagram will also be included to provide a clear view into the functionality of each role in the system and how they all interact with each other.

By detailing all these features, the report aims to provide SSS with a clear blueprint on what should be put in place in order to create an effective emergency response system while also offering tools to be able maintain it in a way that ensures constant effectiveness and efficiency with low risk of failure.

In order to reach this goal, a number of objectives need to be achieved in a way that is extensive, detailed and clear. It is vital that the report is extensive as in order to run a clear and effective emergency response system, all possible outcomes and aspects must be considered so all staff are able to work with efficiency and their roles and responsibilities clearly established. By focusing on detail and working to achieve as much clarity as possible, there will be less room for assumption allowing for consistency and accuracy for all who use the system. Business rules assist in this as through those each member of the staff know clearly how they are expected to operate. Use case diagrams and descriptions as well as activity diagrams also are beneficial in providing extensive detail as they allow users to understand in detail how each step works and any potential alternative actions. Lastly, the domain class diagram will play an important role in assuring the objectives of this report are met as if done correctly and with focus on detail, it will work to provide great clarity on how each aspect and team member of the system works and is supposed to interact. If all of these objectives are met, the report should provide a strong blueprint of how the SSS system should be designed with each member of staff clearly knowing their place and how to interact within it

Business rules

General

1. First-come-first-serve basis for emergency alerts from human reporters.
2. Operators will not put human reporters on hold.
3. Automatic sensors will be able to send alerts to the relevant dispatch center during a set time after activation.
4. Human operators will be able to access/modify to include new newly contained information but not overwrite the emergency alerts sent by an automatic sensor.
5. GPS locating by mobile phone, looking up the registered address of a landline phone or sensor, or by radio triangulation will begin immediately as a reporter contacts the Triple S operator.
6. The details of each emergency report will be recorded in one log if the same incident.
7. The information of an active emergency that Triple S has already got more than one report for will be merged for ease of information.
8. Conflicting report information of same incident in a log will not be deleted.
9. A Separate log of operators' information and each emergency alert action they took since employment will be maintained.
10. Operators' logs will only have access permissions granted to managers. Overwrite and modify permissions will be denied.
11. All voice communication between an operator and reporter will be recorded from initial contact to end.
12. Recordings will be attached to the relevant emergency log.
13. An operator will send written notices to radio frequencies and local television stations as required.
14. All operators' notices written to radio frequencies and local television stations will be recorded in a separate log each day.
15. The Local Emergency Operations Controller will routinely check and have access permissions to daily logs of the operator's written notices.
16. Emergency alert logs and the attached recordings will not be deleted until 7 seven years have passed.
17. Live resources will be constantly modified by the appropriate administrator.
registered rules by a business with a sensor
18. Business supplies Address/Location which is automatically transmitted/retrieved on an alarm from its sensor(s).
19. Business supplies condition(s) that the system will automatically enact if fulfilled (Alarm active for 5+ minutes, send police).
20. Operator must remain in contact with the reporter until emergency services have arrived at the location.
21. Operator must attempt to obtain the location of the emergency through technological means or asking the reporter.
22. Operator must decide which services are necessary to be dispatched for the emergency.
23. System must maintain record of all Emergency response services and Location of dispatch centers.
24. System must maintain live updates on which resources are available to send out and which are currently in use.
25. Operator must contact all other involved emergency control centers to share information and coordinate a response.
26. Operator should pass the alert onto the Air Traffic Control center responsible for the area the aircraft is in.
27. Operators must be member of the NSW Police Force

28. Roster for operators according to anticipated demand must be made using the availability of each operator to staff the center 24/7
29. Roster for operators must take into account the conditions on a daily basis
30. Each operator must have securely stored records Basic personal information, Qualifications, Experience, Availability, Medical information and Mental health records
31. Roster only accessible to a manager
32. Manager reports at the end of each month the manager requires reports of the system for that month.
33. Operator can manually choose which dispatch center(s) to contact
34. Operator can do an automatic search for a close suitable center with available resources
35. Operator must coordinate the required services and keep them all up to date with information on the response.

Work, Health and Safety

1. Provided data will be analyzed to implement the most safe and effective measures for controlling the given situation.
2. Dispatch centers will be managed by software to ensure a safe work premises for employees, contractors, visitors and volunteers.
3. Storage of equipment used for emergency scenarios will be managed to be safe and appropriate for the equipment.
4. Vehicles and machinery will be maintained, organized and conducted in safe and appropriate ways by software.
5. Worker's qualifications will be recorded to ensure the right employee carries out the right task they are qualified for.
6. Necessary information, instructions or supervision necessary will be remotely provided to protect persons from risks to health and safety.
7. Workers compensation insurance policies will be stored in the event of work-related injury or illness.
8. Directions given by emergency services workers will be audited to ensure the directions are not with intent to coerce or induce a person.
9. When a situation develops beyond what is manageable by Triple S, instructions to evacuate will be issued to avoid unnecessary harm to personnel.
10. Emergency evacuation plan for Triple S administration offices will be managed to ensure the most up-to-date evacuation routes are used in the event of an emergency.

Legislation

1. Any person who tampers with any emergency alarm or signaling apparatus is guilty of an offence that incurs a maximum penalty of: On the first offence, 20 penalty units or imprisonment for 1 month, or both, or On the second or subsequent offence, 50 penalty units or imprisonment for 12 months, or both.
2. Any person who intentionally or recklessly gives any false alarm of an emergency is guilty of an offense that incurs a maximum penalty of: On the first offence, 20 penalty units or imprisonment for 1 month, or both, or On the second or subsequent offence, 50 penalty units or imprisonment for 12 months, or both.

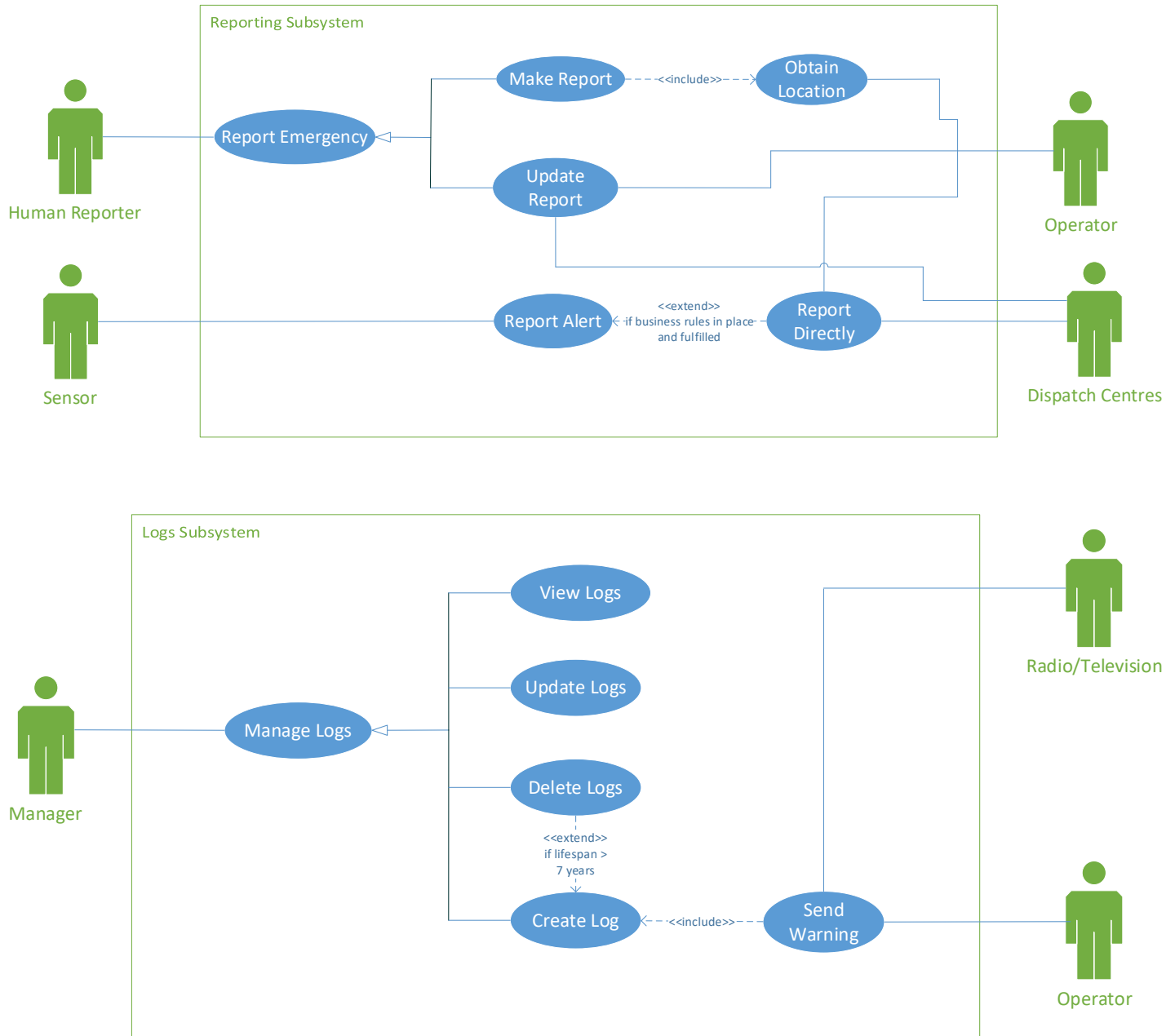
3. Any person who conducts or participates in the unauthorized collection of money or property¹ from the public in return for the provision of emergency services without the consent of the relevant emergency authorities is guilty of an offence that incurs a maximum penalty of 50 penalty units.
 4. Emergency service organisations will not be charged for services provided by the operators of emergency call services, such as the receiving and handling of calls to an emergency service number and the transfer of such calls to emergency service organisations.
 5. Emergency call service operators cannot, without written notice, change the physical location in which they operate their emergency call services.
 - 6.
 7. When emergency call service operators transfer an emergency call to the relevant emergency service organisation, they must provide the following information: The most precise location information regarding the emergency available at the time when the call is transferred; name of the caller requesting emergency services; and public number from which the emergency call was made.
 8. Emergency call service operators must give the ACMA (Australian Communications and Media Authority) a copy of the records of the number and kinds of calls received by the emergency call service on a quarterly basis – within 1 (one) month after the quarter ends.
 9. For each day that the emergency call service receives emergency calls, emergency call service operators must answer 95% of all calls made to the emergency service numbers each day within 10 (ten) seconds of the call reaching the service.
 10. If an individual makes an unsuccessful emergency call, meaning that the call did not reach the relevant termination or answering point for the situation, emergency call service operators must conduct a welfare check on the caller through contacting them by phone or SMS. If attempts to contact the caller are unsuccessful, the matter should be referred to the police force in the location where the caller is located.
 11. The emergency call operator must keep a record of all calls answered 5 seconds after a call reaches the relevant answering point for the call, 10 seconds after a call reaches the relevant answering point for the call, and (more than 10 after a call reaches the relevant answering point for the call.
-

Ethical/Privacy

1. All collected user information such as phone numbers or addresses will be restricted and only accessible by workers who require such information with two factor authentication.
2. Privacy policies are to be easily accessible on the service's website and terms and conditions will need to be agreed to before using the app.
3. All employees are to complete cybersecurity training every 6 months to ensure competency in protecting data.
4. Daily Backups will be made to avoid data loss.
5. Emergency responders will be trained in patient rights and provided with a digital certificate to ensure that they always ask for consent before acting unless the patient lacks capacity to give it and their wishes are unknown or the law requires treatment to be given regardless.
6. In the event of a data breach, all who are affected will be contacted and told what data has been exposed.
7. All new employees will be required to sign a confidentiality agreement to ensure that they do not expose sensitive information.
8. All areas of the office containing sensitive information must remain locked and only accessible by appropriate staff.
9. All employees must change their passwords every 3 months.
10. In the event of an employee attempting to access sensitive information, reasoning must be provided and permission must be granted beforehand.

Use-cases

Reporting and log subsystems and descriptions



Reporting subsystem description

The Reporting subsystem handles all reporting of emergencies. It allows human reporters to both create and update reports within the system, as well as allowing sensors to directly report emergencies to appropriate dispatch centers. This as well as giving the operator the ability to obtain the location of an emergency if necessary.

Log subsystem description

The Logs subsystem handles all things relating to logs as well as handling warnings sent to appropriate media sources. The logs are generally handled by a Manager in Triple S and created in any situation that needs them. It also allows for the automatic deletion of logs after seven years.

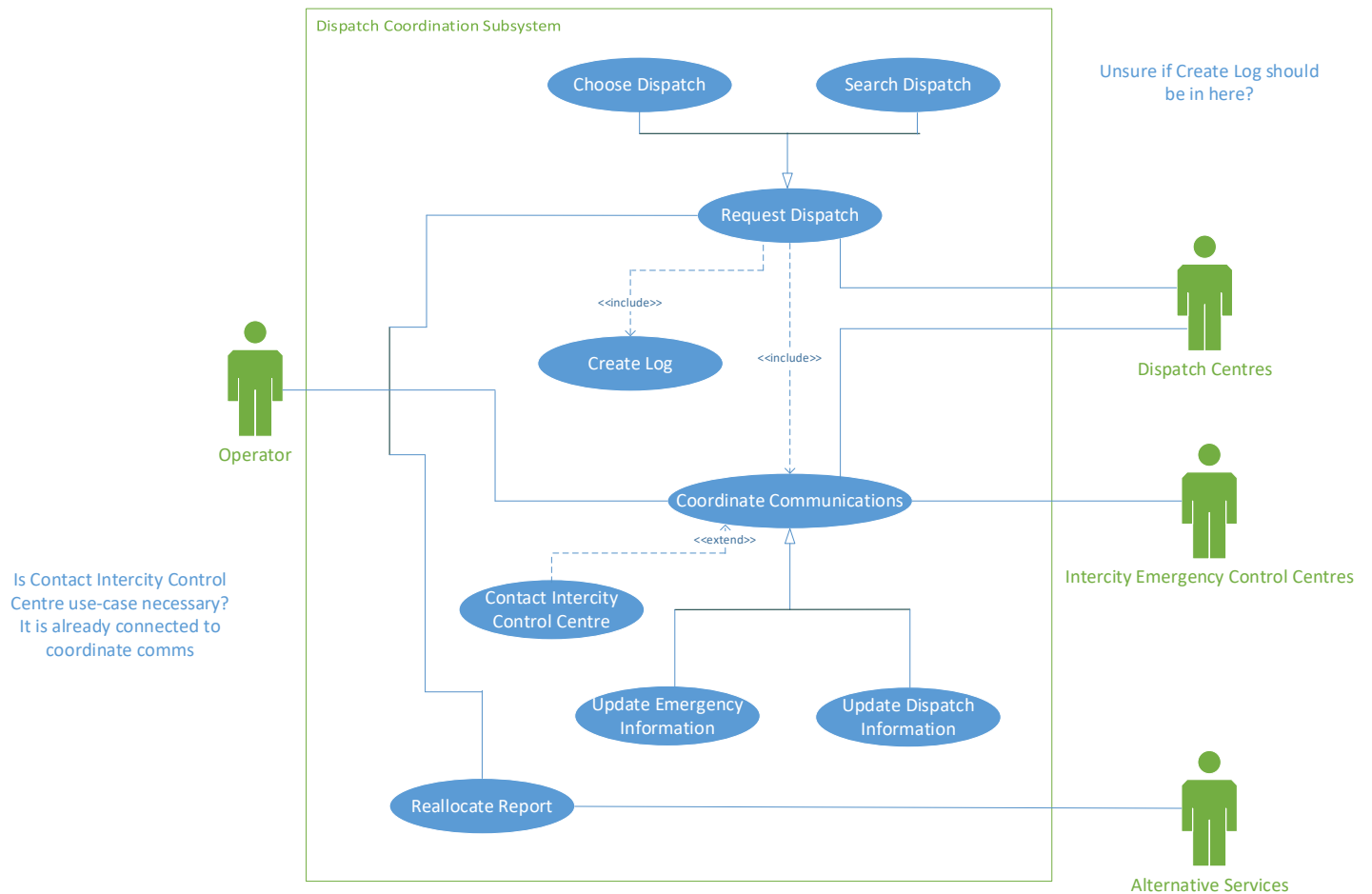
(Jake)

Use case name	Make report	
Scenario	Contact made to report an emergency	
Triggering event	Human reporter contacts Triple S operator	
Brief description	Human reporters contact Triple S to make a report about an emergency. The human operator makes contact and identifies the incident type and incident location. The operator then creates a report of these incident details, and the system saves the report and creates a relevant log.	
Actor	Human reporter, Operator	
Related use case	Obtain Location, Create Log	
Stakeholders	Government organizations, Triple S, General Public	
Preconditions	There is an operator available to take the call.	
Postconditions	Relevant report containing incident details must be created. Report must be saved in a log. Location and type of incident must be identified.	
Exception conditions	- If the reporter ends the call at any point during contact: a.) The operator will attempt to contact the reporter using the same contact details. b.) If the operator is unable to contact the reporter, creates an incident report using the known details and requests dispatch of local emergency services to the last known location of the reporter.	
Alternative flow	8.1 If a report has already been made about the same incident: a.) The system updates the existing report with the details of the current report.	
Flow of activities	Actor	System
	<ol style="list-style-type: none">1. Human Reporter calls the Triple S centre via Radio/Phone.2. Operator accepts the call.3. Operator requests reporter location.4. Reporter reports location.5. Operator requests incident details.6. Reporter reports incident details.7. Operator inputs incident details into a report in the system.	<ol style="list-style-type: none">1. System receives contact.1.1 System begins recording communication.1.2 System obtains reporter operation automatically.1.3 System notifies operator.

(John Deniel)

Use Case Name:	Report Sensor Alert	
Scenario:	An alert from a registered sensor is reported.	
Triggering Event:	A registered sensor detects an incident that requires reporting an alert/emergency.	
Brief Description:	A registered sensor reports an incident alert to a human operator. The operator takes the alert report and the registered information of the sensor to determine the location and type of incident.	
Actors:	Sensor, Operator	
Stakeholders:	Triple S, Emergency Services, Private Individuals	
Preconditions:	The sensor is registered to the Triple S system. The sensor is operational and is capable of sending reports to the Triple S system. There is a human operator available to process the report.	
Postconditions:	The sensor that reported the alert must be identified. The type of the incident reported must be identified. The location of the incident reported must be identified.	
Flow of Activities:	Actor	System
	<ol style="list-style-type: none">1. Registered sensor reports an incident alert to the system.2. Operator looks up the registered information of the sensor on the system.3. Operator identifies the location and type of the reported incident.	<ol style="list-style-type: none">1.1 System receives the report from the sensor.1.2 System sends a notification to a human operator about the report.2.1. System shows the registration information of the sensor including type of sensor and its registered address.
Alternative Flow:	<ol style="list-style-type: none">1.2. If the sensor has a registered rule to automatically dispatch emergency services is an alert is reported,<ol style="list-style-type: none">a. System confirms whether the registered rule for automatic dispatch has been met.b. If the rule is met, system sends a report to emergency dispatch services to request dispatch to the registered sensor's location.c. If the rule is not met, system treats the alert as a false alarm and ignores the report.	
Exception Conditions:		

Dispatch Coordination subsystem and description



Dispatching subsystem description

The Dispatching subsystem handles all of the coordination between an Operator and dispatches. It allows for an Operator to request dispatches either automatically or manually as well as further coordinating the dispatches sent. If necessary, allows for an Operator to also reallocate a report to necessary services.

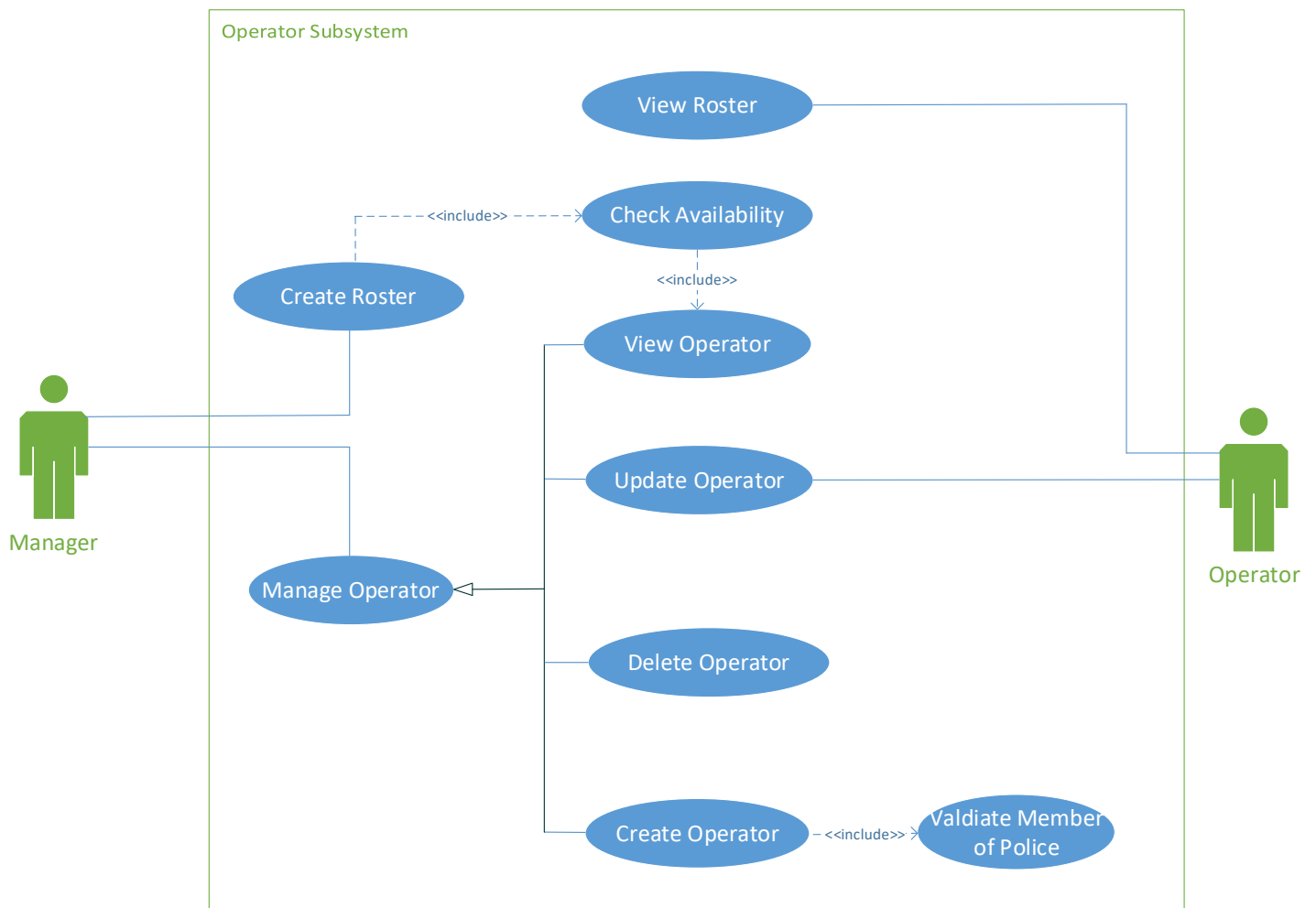
(Zane)

Use Case Name	Choose Dispatch	
Scenario	Select the appropriate emergency services to dispatch	
Triggering Event	Operator enters the dispatch request system after having identified the incident details	
Brief Description	Operators input the incident details into the system, and the system presents them with a list of all available dispatch resources sorted by distance to incident location. The operator then selects manually the dispatches and prompts the system to send a dispatch request to the relevant emergency dispatch centres.	
Actor	Operator, Dispatch Centre	
Related Use Case	Search for Dispatch Request Dispatch – parent use case	
Stakeholders	Emergency Services, Triple S	
Preconditions	Live Resources must be available Dispatch Services must be available	
Postconditions	Dispatch request successfully delivered to the dispatch centre	
Exception Conditions		
Alternative Flow	1.3. Alerts operator if no dispatches are available, resumes search for available dispatches, unless aborted by operator.	
Flow of Activities	Actor	System
	1. Operator enters the location and type of emergency into the system 2. Operator selects from available dispatches and prompts system to request dispatch	1.1. System identifies the input 1.2. System sorts and filters all available dispatches according to the input 1.3. System displays all available dispatches post-filtering 2.1. System requests dispatch from appropriate services

Full Use Case – Search for Dispatch – Kaleb Jones

Use Case Name:	Search For Dispatch	
Scenario:	Operator searching for an available dispatch	
Triggering Event:	Operator commences an automatic search for a dispatch	
Brief Description:	The operator inputs the incident details into the system. The system uses the information to automatically select the best dispatch resources to request.	
Actors:	Operator, Dispatch Centre	
Related Use Cases	Choose Dispatch, Request Dispatch	
Stakeholders:	Dispatch Centres, City Council, Triple S	
Preconditions:	A report of an emergency was received Location and type of emergency have been identified Operator available to take the report and make the dispatch request	
Postconditions:	System must have sent a dispatch request to the relevant dispatch centre	
Flow of Activities:	Operator	System
	<ol style="list-style-type: none"> Operator enters location and type of emergency Operator provides permission to the system to request dispatch 	<ol style="list-style-type: none"> System identifies information System searches for closest dispatch centres to location with available resources of emergency type System displays best dispatch resources for the identified incident System sends a dispatch request to the selected dispatch centre
Alternative Flow:	1.3. No suitable dispatch resources could be found for the incident a.) The system notifies the operator that there are no suitable dispatches to request at the moment, and continues searching for an available dispatch	
Exception Conditions:	<ol style="list-style-type: none"> Operator inputs incorrect or invalid input <ol style="list-style-type: none"> System asks for proper input, and restarts the use case Operator rejects the prompt asking for permission to request the dispatch <ol style="list-style-type: none"> System resets to resting state 	

Operator subsystem and description.



Operator subsystem description

The Operator subsystem allows for storage and handling of operator information. Through this subsystem an operator can update their information, while a manager can manage operators entirely. A manager can also schedule a roster of operators using their information.

Use Case Name:	Create Roster	
Scenario:	Create and edit roster.	
Triggering Event:	The user attempts to log in to the roster system	
Brief Description:	A manager logs on to the roster system in order to create and edit shifts. The system presents the manager with the number of shifts needed to be filled and the manager assigns operators to these shifts.	
Actors:	Manager, Operator	
Stakeholders:	Triple S, Emergency Services, Managers, Staff	
Preconditions:	<ul style="list-style-type: none"> - Number of rostered hours needed is available. - Operator availability is known to the manager for the relevant period. 	
Postconditions:	<ul style="list-style-type: none"> - Roster is updated and uploaded to system. - Relevant users must be notified of changes made. 	
Exception Conditions:		
Alternative Flow:	<p>3.1. The operator being assigned the shift is listed as unavailable for the shift.</p> <p>a.) The system notifies the manager that the operator is unavailable to work that shift and revokes the assignment.</p> <p>b.) The system returns to the state allowing the manager to assign staff to shifts.</p> <p>3.1. The operator has been rostered for over 38 hours in the current roster.</p> <p>a.) The system notifies the manager and requests confirmation to assign the shift.</p> <p>b.) If the manager approves the overtime shift, the shift assignment is saved and returns to normal flow.</p> <p>c.) If the manager rejects the overtime shift, the shift is revoked, and the system returns to the state allowing the manager to assign staff to shifts.</p> <p>4. The manager selects to assign another shift.</p> <p>a.) The system returns to the state allowing the manager to assign staff to shifts.</p>	
Flow of Activities:	Actor	System
	<ol style="list-style-type: none"> 1. User accesses the rosters login portal. 2. User inputs login credentials. 3. User allocates staff members to specific shifts. 	<ol style="list-style-type: none"> 1.1. System asks for appropriate login credentials. 2.1. System verifies the login information. 2.2. System presents user with a table representing the work calendar. 3.1. System saves the shift assignment. 3.2. System presents manager with current

(Ethan)

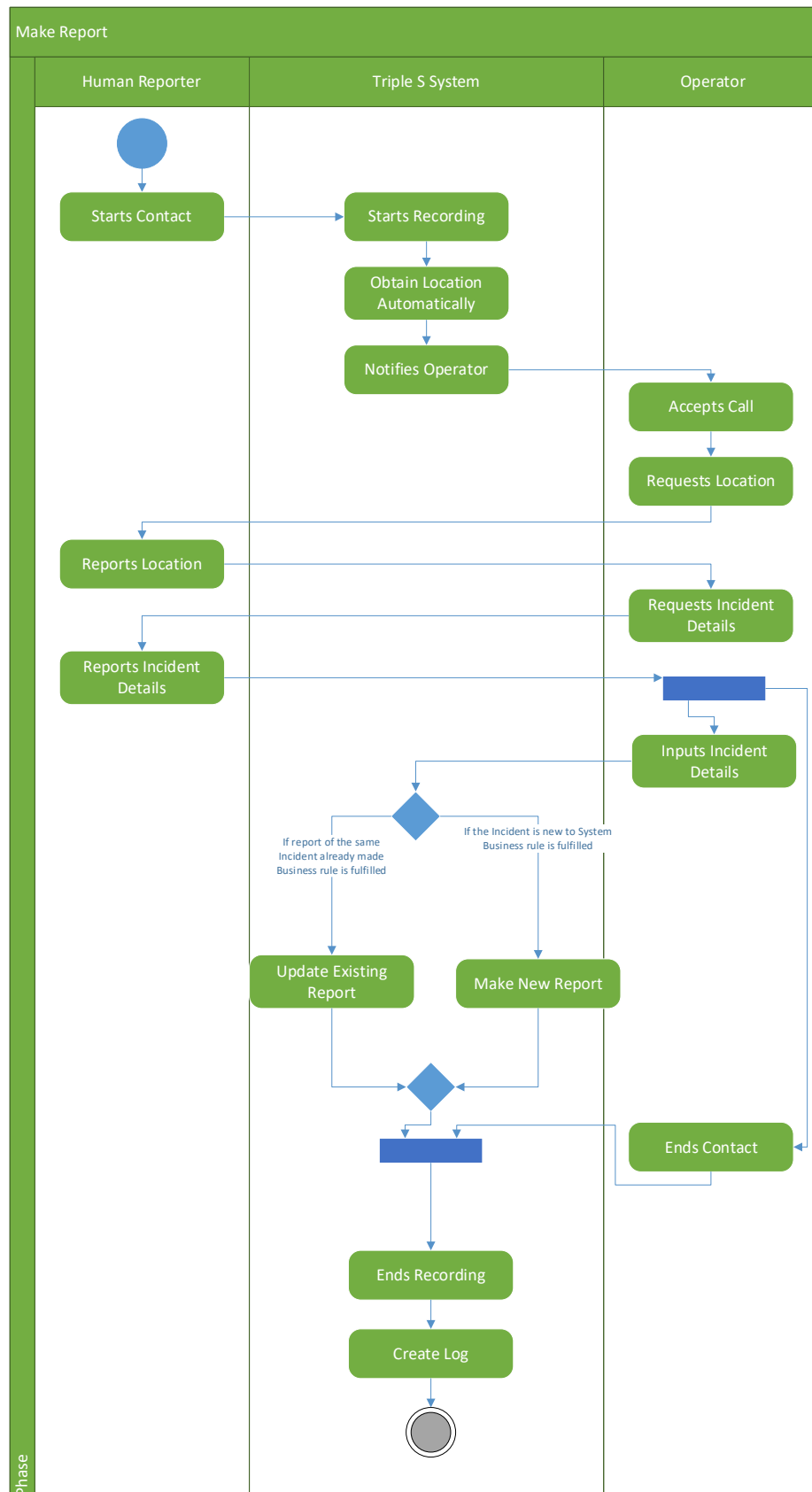
Brief Descriptions

Subsystem	Use-Case	Brief Description
Operator	Create Roster	Allows a manager to create a roster for operators that is filled 24/7. Uses Check Availability to obtain necessary information.
	Check Availability	Included in Create Roster, this use case allows for an operators availability to be checked from their currently updated information.
	Manage Operator	Manage operator is a parent use case that is for overall management of anything operator related.
	View Operator	Allows an operators information to be viewed by a manager.
	Update Operator	Update Operator is used to update an operators information through either the manager, or the operator themselves. Primarily used to update availability.
	Delete Operator	Use case for deleting the records of an operator, only accessible by a manager.
	Create Operator	Allows an operator actor to be created by a manager. This then calls the Validation use case.
	Validate Member of Police	Use case to ensure that any operator is a member of the police force before becoming an operator.
Reporting	Report Emergency	Human reporter can use the system to report an emergency, either through phone or radio.
	Make Report	Child of Report Emergency. Allows the reporter to create a report on an emergency.
	Update Report	Child of Report Emergency. Lets a human reporter update a report that has been called in, whether this has been through a

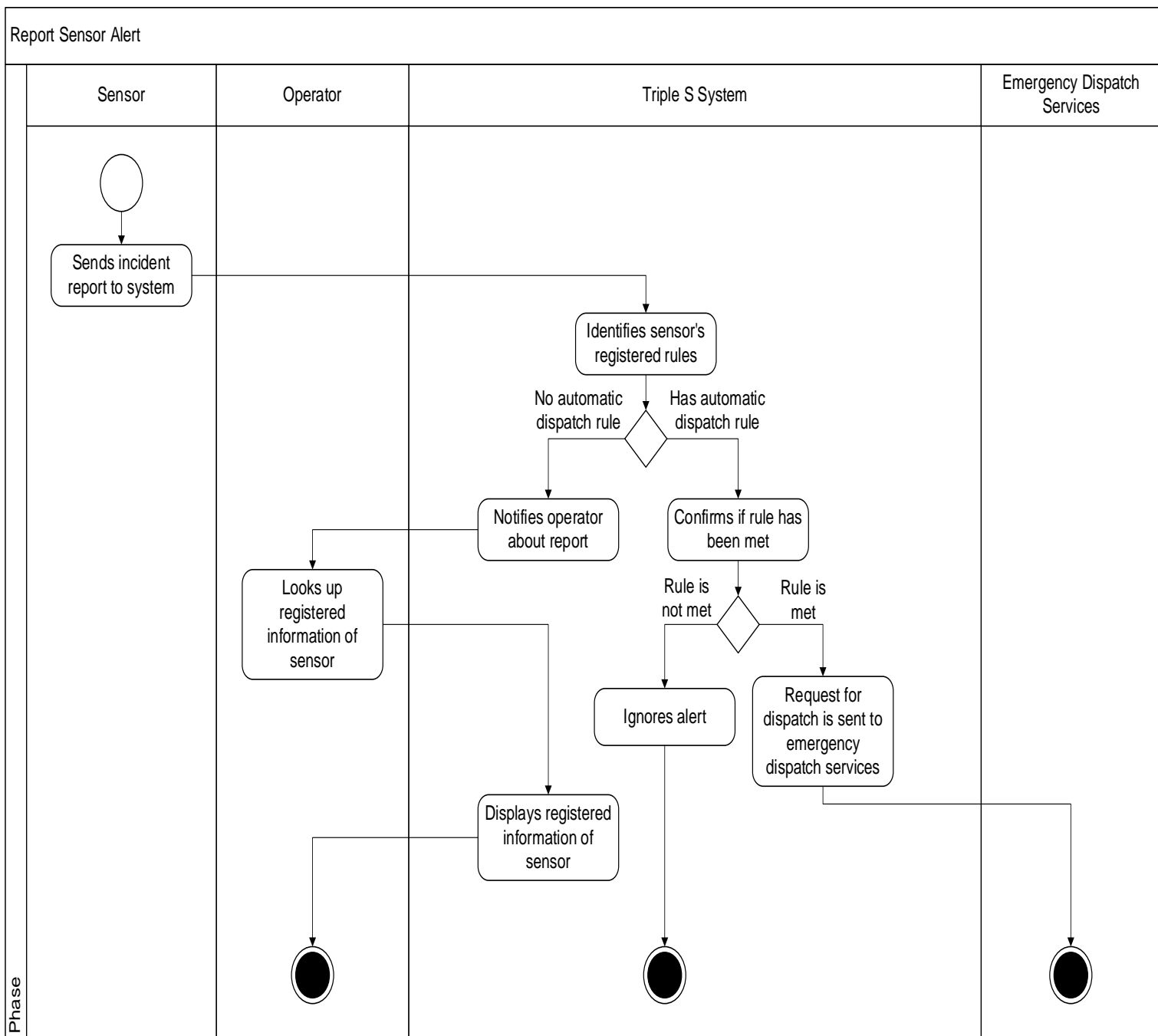
		continuous report or through multiple.
	Obtain Location	Allows an operator to obtain the location of an emergency that is in progress. Can be done through GPS and many other technological methods.
	Report Alert	Allows a sensor to report an alert due to it automatically sensing a problem. Will be sent to an operator, or potentially through Report Directly use case.
	Report Directly	Report Directly extends Report Alert through the function of business rules. These rules are set up by a business with a sensor to automatically contact a dispatch after a condition is filled.
Logs	Manage Logs	A parent use case for the general management of logs. Can only be accessed by a manager.
	View Logs	Allows logs to be viewed by a manager. Used for reviewing and validating reports.
	Update Logs	Update Logs allows for created logs to be updated. This can be done if a report has progressed past the point of an initial report.
	Delete Logs	Allows logs to be deleted by a manager. Delete Logs also extends Create Logs by having a condition that if the logs are over seven years old, then they will be automatically deleted.
	Create Logs	Create Logs is a use case for creating logs. Typically done automatically through processes such as Send Warning or Make Report
	Send Warning	Operator can send a warning of an emergency or potential emergencies
Dispatching	Choose Dispatch	Child of Request Dispatch. Allows for an operator to manually choose which dispatch

		to send to an emergency that is believe to best suit it.
	Search Dispatch	Child of Request Dispatch. Allows for an operator to use the system to automatically search for the best available dispatch that suits the emergency.
	Request Dispatch	Parent use case for requesting a dispatch. Allows an operator to request dispatch centres or call Reallocate Report use case for a reported emergency.
	Coordinate Communications	Coordinate Communications allows for an operator to appropriately give out and update the information of a given report to all necessary actors.
	Update Emergency Information	Child of Coordinate Communications. Used to directly update the information regarding an emergency on the system.
	Update Dispatch Information	Child of Coordinate Communications. Used to update and information on the dispatches that have been sent to handle the emergency.
	Reallocate Report	Allows for an operator to reallocate a report to potential alternative services if the scope of a reported emergency is beyond that of Triple S.

Activity Diagrams



(Jake)



(John Deniel)

UML Activity Diagram: Create Roster

Swimlanes: Manager, Roster System, Staff Member

Flow:

- Manager:** Starts at a start node, leading to "Input information at roster's login portal".
- Roster System:**
 - Receives input from Manager, leading to "Confirms login information".
 - Then "Displays work calendar".
 - Receives input from Manager, leading to a join node (black rectangle).
 - From the join node, the flow goes to "Allocates staff member to specific shift".
 - From "Allocates staff member to specific shift", the flow goes to a decision diamond.
 - Decision 1:** "Shift does not conflict with operator availability". If true, it goes to "Operator rostered for more than 38 hours". If false, it goes to "Shift conflicts with operator availability".
 - Decision 2:** "Operator rostered for more than 38 hours". If true, it goes to "Notifies manager and requests confirmation". If false, it goes to "Operator rostered for less than 38 hours".
 - Decision 3:** "Shift conflicts with operator availability". If true, it goes to "Notifies manager and revokes shift assignment". If false, it goes to a join node (black rectangle).
 - Decision 4:** "Notifies manager and requests confirmation". If true, it goes to "Approves overtime shift". If false, it goes to "Rejects overtime shift".
 - Decision 5:** "Approves overtime shift". If true, it goes to a join node (black rectangle). If false, it goes to "Rejects overtime shift".
 - Decision 6:** "Rejects overtime shift". If true, it goes to a join node (black rectangle). If false, it goes to "Manager clicks assign shift".
 - Decision 7:** "Manager clicks assign shift". If true, it goes to "Prompts system to save roster". If false, it goes to "Manager clicks save roster".
 - Decision 8:** "Manager clicks save roster". If true, it goes to "Prompts system to save roster". If false, it goes to "Uploads roster".
 - Decision 9:** "Prompts system to save roster". If true, it goes to "Uploads roster". If false, it goes to "Manager clicks assign shift".
 - Decision 10:** "Uploads roster". If true, it goes to "Notifies operators of roster change". If false, it goes to "Manager clicks assign shift".
- Staff Member:** Receives input from "Notifies operators of roster change", leading to a final node (black circle).

```
graph TD
    subgraph Manager
        Start(( )) --> Input[Input information at roster's login portal]
        Input --> Join1(( ))
        Join1 --> Alloc[Allocates staff member to specific shift]
        Alloc --> Dec1{ }
        Dec1 -- "Shift does not conflict with operator availability" --> Dec2{ }
        Dec1 -- "Shift conflicts with operator availability" --> Dec3{ }
        Dec2 -- "Operator rostered for more than 38 hours" --> Dec4{ }
        Dec2 -- "Operator rostered for less than 38 hours" --> Dec5{ }
        Dec4 -- "Notifies manager and requests confirmation" --> Dec6{ }
        Dec4 -- "Approves overtime shift" --> Dec7{ }
        Dec6 -- "Rejects overtime shift" --> Dec8{ }
        Dec6 -- "Approves overtime shift" --> Dec9{ }
        Dec8 -- "Rejects overtime shift" --> Dec10{ }
        Dec8 -- "Manager clicks assign shift" --> Dec11{ }
        Dec11 -- "Manager clicks save roster" --> Dec12{ }
        Dec12 -- "Prompts system to save roster" --> Dec13{ }
        Dec13 -- "Uploads roster" --> Dec14{ }
        Dec13 -- "Manager clicks assign shift" --> Dec11
        Dec14 -- "Notifies operators of roster change" --> End(( ))
    end

    subgraph Roster_System [Roster System]
        Confirms[Confirms login information]
        Displays[Displays work calendar]
        Join1
        Alloc
        Dec1
        Dec2
        Dec3
        Dec4
        Dec5
        Dec6
        Dec7
        Dec8
        Dec9
        Dec10
        Dec11
        Dec12
        Dec13
        Dec14
        Join2(( ))
        Join3(( ))
        Join4(( ))
        Join5(( ))
        Join6(( ))
        Join7(( ))
        Join8(( ))
        Join9(( ))
        Join10(( ))
        Join11(( ))
        Join12(( ))
        Join13(( ))
        Join14(( ))
        Join15(( ))
        Join16(( ))
        Join17(( ))
        Join18(( ))
        Join19(( ))
        Join20(( ))
        Join21(( ))
        Join22(( ))
        Join23(( ))
        Join24(( ))
        Join25(( ))
        Join26(( ))
        Join27(( ))
        Join28(( ))
        Join29(( ))
        Join30(( ))
        Join31(( ))
        Join32(( ))
        Join33(( ))
        Join34(( ))
        Join35(( ))
        Join36(( ))
        Join37(( ))
        Join38(( ))
        Join39(( ))
        Join40(( ))
        Join41(( ))
        Join42(( ))
        Join43(( ))
        Join44(( ))
        Join45(( ))
        Join46(( ))
        Join47(( ))
        Join48(( ))
        Join49(( ))
        Join50(( ))
        Join51(( ))
        Join52(( ))
        Join53(( ))
        Join54(( ))
        Join55(( ))
        Join56(( ))
        Join57(( ))
        Join58(( ))
        Join59(( ))
        Join60(( ))
        Join61(( ))
        Join62(( ))
        Join63(( ))
        Join64(( ))
        Join65(( ))
        Join66(( ))
        Join67(( ))
        Join68(( ))
        Join69(( ))
        Join70(( ))
        Join71(( ))
        Join72(( ))
        Join73(( ))
        Join74(( ))
        Join75(( ))
        Join76(( ))
        Join77(( ))
        Join78(( ))
        Join79(( ))
        Join80(( ))
        Join81(( ))
        Join82(( ))
        Join83(( ))
        Join84(( ))
        Join85(( ))
        Join86(( ))
        Join87(( ))
        Join88(( ))
        Join89(( ))
        Join90(( ))
        Join91(( ))
        Join92(( ))
        Join93(( ))
        Join94(( ))
        Join95(( ))
        Join96(( ))
        Join97(( ))
        Join98(( ))
        Join99(( ))
        Join100(( ))
        Join101(( ))
        Join102(( ))
        Join103(( ))
        Join104(( ))
        Join105(( ))
        Join106(( ))
        Join107(( ))
        Join108(( ))
        Join109(( ))
        Join110(( ))
        Join111(( ))
        Join112(( ))
        Join113(( ))
        Join114(( ))
        Join115(( ))
        Join116(( ))
        Join117(( ))
        Join118(( ))
        Join119(( ))
        Join120(( ))
        Join121(( ))
        Join122(( ))
        Join123(( ))
        Join124(( ))
        Join125(( ))
        Join126(( ))
        Join127(( ))
        Join128(( ))
        Join129(( ))
        Join130(( ))
        Join131(( ))
        Join132(( ))
        Join133(( ))
        Join134(( ))
        Join135(( ))
        Join136(( ))
        Join137(( ))
        Join138(( ))
        Join139(( ))
        Join140(( ))
        Join141(( ))
        Join142(( ))
        Join143(( ))
        Join144(( ))
        Join145(( ))
        Join146(( ))
        Join147(( ))
        Join148(( ))
        Join149(( ))
        Join150(( ))
        Join151(( ))
        Join152(( ))
        Join153(( ))
        Join154(( ))
        Join155(( ))
        Join156(( ))
        Join157(( ))
        Join158(( ))
        Join159(( ))
        Join160(( ))
        Join161(( ))
        Join162(( ))
        Join163(( ))
        Join164(( ))
        Join165(( ))
        Join166(( ))
        Join167(( ))
        Join168(( ))
        Join169(( ))
        Join170(( ))
        Join171(( ))
        Join172(( ))
        Join173(( ))
        Join174(( ))
        Join175(( ))
        Join176(( ))
        Join177(( ))
        Join178(( ))
        Join179(( ))
        Join180(( ))
        Join181(( ))
        Join182(( ))
        Join183(( ))
        Join184(( ))
        Join185(( ))
        Join186(( ))
        Join187(( ))
        Join188(( ))
        Join189(( ))
        Join190(( ))
        Join191(( ))
        Join192(( ))
        Join193(( ))
        Join194(( ))
        Join195(( ))
        Join196(( ))
        Join197(( ))
        Join198(( ))
        Join199(( ))
        Join200(( ))
        Join201(( ))
        Join202(( ))
        Join203(( ))
        Join204(( ))
        Join205(( ))
        Join206(( ))
        Join207(( ))
        Join208(( ))
        Join209(( ))
        Join210(( ))
        Join211(( ))
        Join212(( ))
        Join213(( ))
        Join214(( ))
        Join215(( ))
        Join216(( ))
        Join217(( ))
        Join218(( ))
        Join219(( ))
        Join220(( ))
        Join221(( ))
        Join222(( ))
        Join223(( ))
        Join224(( ))
        Join225(( ))
        Join226(( ))
        Join227(( ))
        Join228(( ))
        Join229(( ))
        Join230(( ))
        Join231(( ))
        Join232(( ))
        Join233(( ))
        Join234(( ))
        Join235(( ))
        Join236(( ))
        Join237(( ))
        Join238(( ))
        Join239(( ))
        Join240(( ))
        Join241(( ))
        Join242(( ))
        Join243(( ))
        Join244(( ))
        Join245(( ))
        Join246(( ))
        Join247(( ))
        Join248(( ))
        Join249(( ))
        Join250(( ))
        Join251(( ))
        Join252(( ))
        Join253(( ))
        Join254(( ))
        Join255(( ))
        Join256(( ))
        Join257(( ))
        Join258(( ))
        Join259(( ))
        Join260(( ))
        Join261(( ))
        Join262(( ))
        Join263(( ))
        Join264(( ))
        Join265(( ))
        Join266(( ))
        Join267(( ))
        Join268(( ))
        Join269(( ))
        Join270(( ))
        Join271(( ))
        Join272(( ))
        Join273(( ))
        Join274(( ))
        Join275(( ))
        Join276(( ))
        Join277(( ))
        Join278(( ))
        Join279(( ))
        Join280(( ))
        Join281(( ))
        Join282(( ))
        Join283(( ))
        Join284(( ))
        Join285(( ))
        Join286(( ))
        Join287(( ))
        Join288(( ))
        Join289(( ))
        Join290(( ))
        Join291(( ))
        Join292(( ))
        Join293(( ))
        Join294(( ))
        Join295(( ))
        Join296(( ))
        Join297(( ))
        Join298(( ))
        Join299(( ))
        Join300(( ))
        Join301(( ))
        Join302(( ))
        Join303(( ))
        Join304(( ))
        Join305(( ))
        Join306(( ))
        Join307(( ))
        Join308(( ))
        Join309(( ))
        Join310(( ))
        Join311(( ))
        Join312(( ))
        Join313(( ))
        Join314(( ))
        Join315(( ))
        Join316(( ))
        Join317(( ))
        Join318(( ))
        Join319(( ))
        Join320(( ))
        Join321(( ))
        Join322(( ))
        Join323(( ))
        Join324(( ))
        Join325(( ))
        Join326(( ))
        Join327(( ))
        Join328(( ))
        Join329(( ))
        Join330(( ))
        Join331(( ))
        Join332(( ))
        Join333(( ))
        Join334(( ))
        Join335(( ))
        Join336(( ))
        Join337(( ))
        Join338(( ))
        Join339(( ))
        Join340(( ))
        Join341(( ))
        Join342(( ))
        Join343(( ))
        Join344(( ))
        Join345(( ))
        Join346(( ))
        Join347(( ))
        Join348(( ))
        Join349(( ))
        Join350(( ))
        Join351(( ))
        Join352(( ))
        Join353(( ))
        Join354(( ))
        Join355(( ))
        Join356(( ))
        Join357(( ))
        Join358(( ))
        Join359(( ))
        Join360(( ))
        Join361(( ))
        Join362(( ))
        Join363(( ))
        Join364(( ))
        Join365(( ))
        Join366(( ))
        Join367(( ))
        Join368(( ))
        Join369(( ))
        Join370(( ))
        Join371(( ))
        Join372(( ))
        Join373(( ))
        Join374(( ))
        Join375(( ))
        Join376(( ))
        Join377(( ))
        Join3
```

<u>(Ethan)</u>

UML Activity Diagram: Create Roster

Swimlanes: Manager, Roster System, Staff Member

Flow:

- Manager:** Starts at a start node, leading to "Input information at roster's login portal".
- Roster System:**
 - Receives input from Manager, leading to "Confirms login information".
 - Then "Displays work calendar".
 - Receives input from Manager, leading to a join node (black rectangle).
 - From the join node, the flow goes to "Allocates staff member to specific shift".
 - From "Allocates staff member to specific shift", the flow goes to a decision diamond.
 - Decision 1:** "Shift does not conflict with operator availability". If true, it goes to "Operator rostered for more than 38 hours". If false, it goes to "Shift conflicts with operator availability".
 - Decision 2:** "Operator rostered for more than 38 hours". If true, it goes to "Notifies manager and requests confirmation". If false, it goes to "Operator rostered for less than 38 hours".
 - Decision 3:** "Shift conflicts with operator availability". If true, it goes to "Notifies manager and revokes shift assignment". If false, it goes to a join node (black rectangle).
 - Decision 4:** "Notifies manager and requests confirmation". If true, it goes to "Approves overtime shift". If false, it goes to "Rejects overtime shift".
 - Decision 5:** "Approves overtime shift". If true, it goes to a join node (black rectangle). If false, it goes to "Rejects overtime shift".
 - Decision 6:** "Rejects overtime shift". If true, it goes to a join node (black rectangle). If false, it goes to "Manager clicks assign shift".
 - Decision 7:** "Manager clicks assign shift". If true, it goes to "Prompts system to save roster". If false, it goes to "Manager clicks save roster".
 - Decision 8:** "Manager clicks save roster". If true, it goes to "Prompts system to save roster". If false, it goes to "Uploads roster".
 - Decision 9:** "Prompts system to save roster". If true, it goes to "Uploads roster". If false, it goes to "Manager clicks assign shift".
 - Decision 10:** "Uploads roster". If true, it goes to "Notifies operators of roster change". If false, it goes to "Manager clicks assign shift".
- Staff Member:** Receives input from "Notifies operators of roster change", leading to a final node (black circle).

```
graph TD
    subgraph Manager
        Start(( )) --> Input[Input information at roster's login portal]
        Input --> Join1(( ))
        Join1 --> Alloc[Allocates staff member to specific shift]
        Alloc --> Dec1{ }
        Dec1 -- "Shift does not conflict with operator availability" --> Dec2{ }
        Dec1 -- "Shift conflicts with operator availability" --> Dec3{ }
        Dec2 -- "Operator rostered for more than 38 hours" --> Dec4{ }
        Dec2 -- "Operator rostered for less than 38 hours" --> Dec5{ }
        Dec4 -- "Notifies manager and requests confirmation" --> Dec6{ }
        Dec4 -- "Approves overtime shift" --> Dec7{ }
        Dec6 -- "Rejects overtime shift" --> Dec8{ }
        Dec6 -- "Approves overtime shift" --> Dec9{ }
        Dec8 -- "Rejects overtime shift" --> Dec10{ }
        Dec8 -- "Manager clicks assign shift" --> Dec11{ }
        Dec11 -- "Manager clicks save roster" --> Dec12{ }
        Dec12 -- "Prompts system to save roster" --> Dec13{ }
        Dec13 -- "Uploads roster" --> Dec14{ }
        Dec13 -- "Manager clicks assign shift" --> Dec11
        Dec14 -- "Notifies operators of roster change" --> End(( ))
    end
    subgraph Roster_System [Roster System]
        Confirms[Confirms login information]
        Displays[Displays work calendar]
        Join1
        Alloc
        Dec1
        Dec2
        Dec3
        Dec4
        Dec5
        Dec6
        Dec7
        Dec8
        Dec9
        Dec10
        Dec11
        Dec12
        Dec13
        Dec14
        Join2(( ))
        Saves[Saves shift assignment]
        Displays2[Displays current roster]
        Uploads[Uploads roster]
        Notifies[Notifies operators of roster change]
    end
    subgraph Staff_Member [Staff Member]
        End
    end
    Input --> Confirms
    Confirms --> Displays
    Join1 --> Alloc
    Alloc --> Dec1
    Dec1 -- "Shift does not conflict with operator availability" --> Dec2
    Dec1 -- "Shift conflicts with operator availability" --> Dec3
    Dec2 -- "Operator rostered for more than 38 hours" --> Dec4
    Dec2 -- "Operator rostered for less than 38 hours" --> Dec5
    Dec4 -- "Notifies manager and requests confirmation" --> Dec6
    Dec4 -- "Approves overtime shift" --> Dec7
    Dec6 -- "Rejects overtime shift" --> Dec8
    Dec6 -- "Approves overtime shift" --> Dec9
    Dec8 -- "Rejects overtime shift" --> Dec10
    Dec8 -- "Manager clicks assign shift" --> Dec11
    Dec11 -- "Manager clicks save roster" --> Dec12
    Dec12 -- "Prompts system to save roster" --> Dec13
    Dec13 -- "Uploads roster" --> Dec14
    Dec13 -- "Manager clicks assign shift" --> Dec11
    Dec14 -- "Notifies operators of roster change" --> End
```

Choose Dispatch

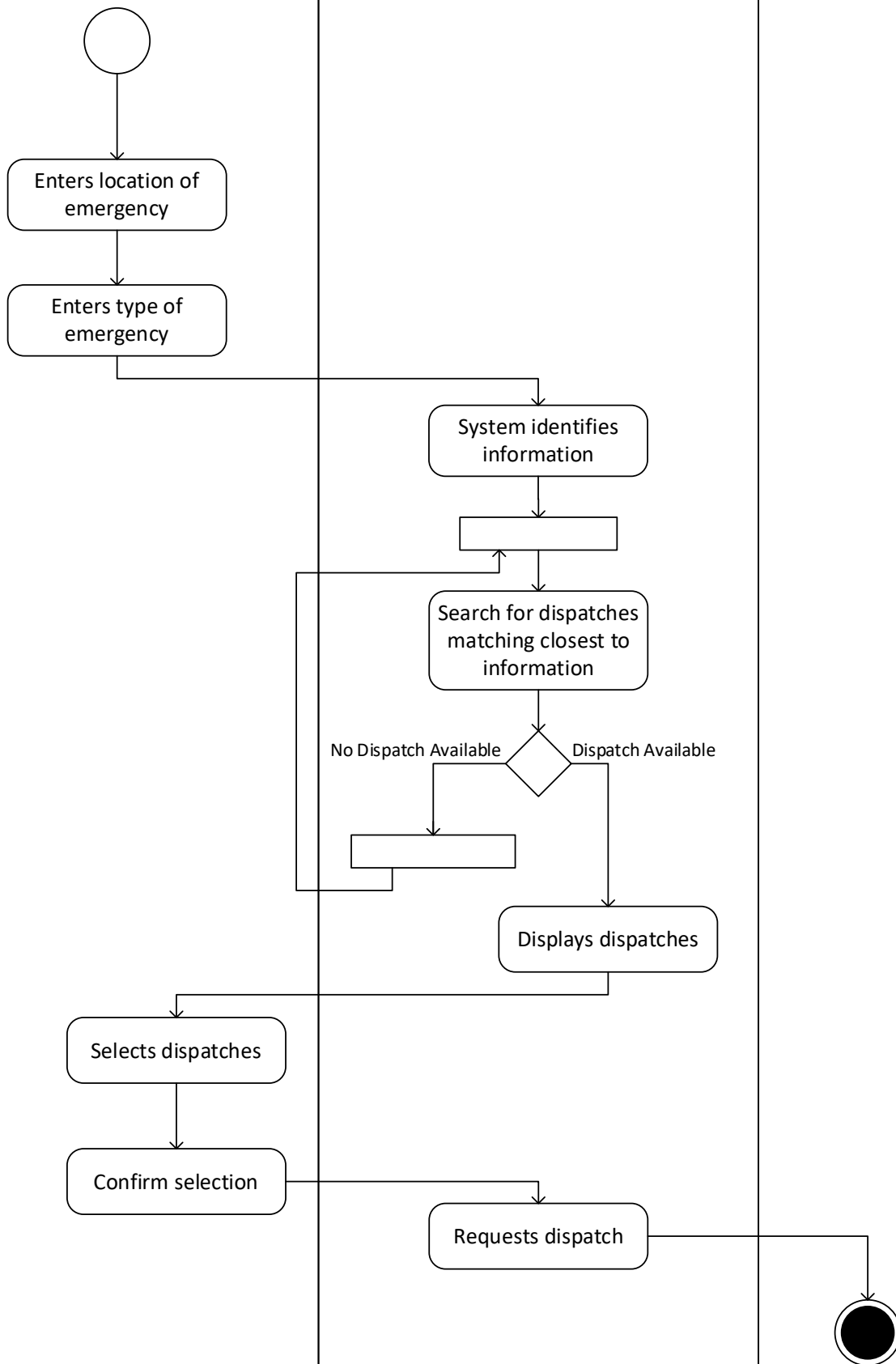
(Kaleb)

Phase

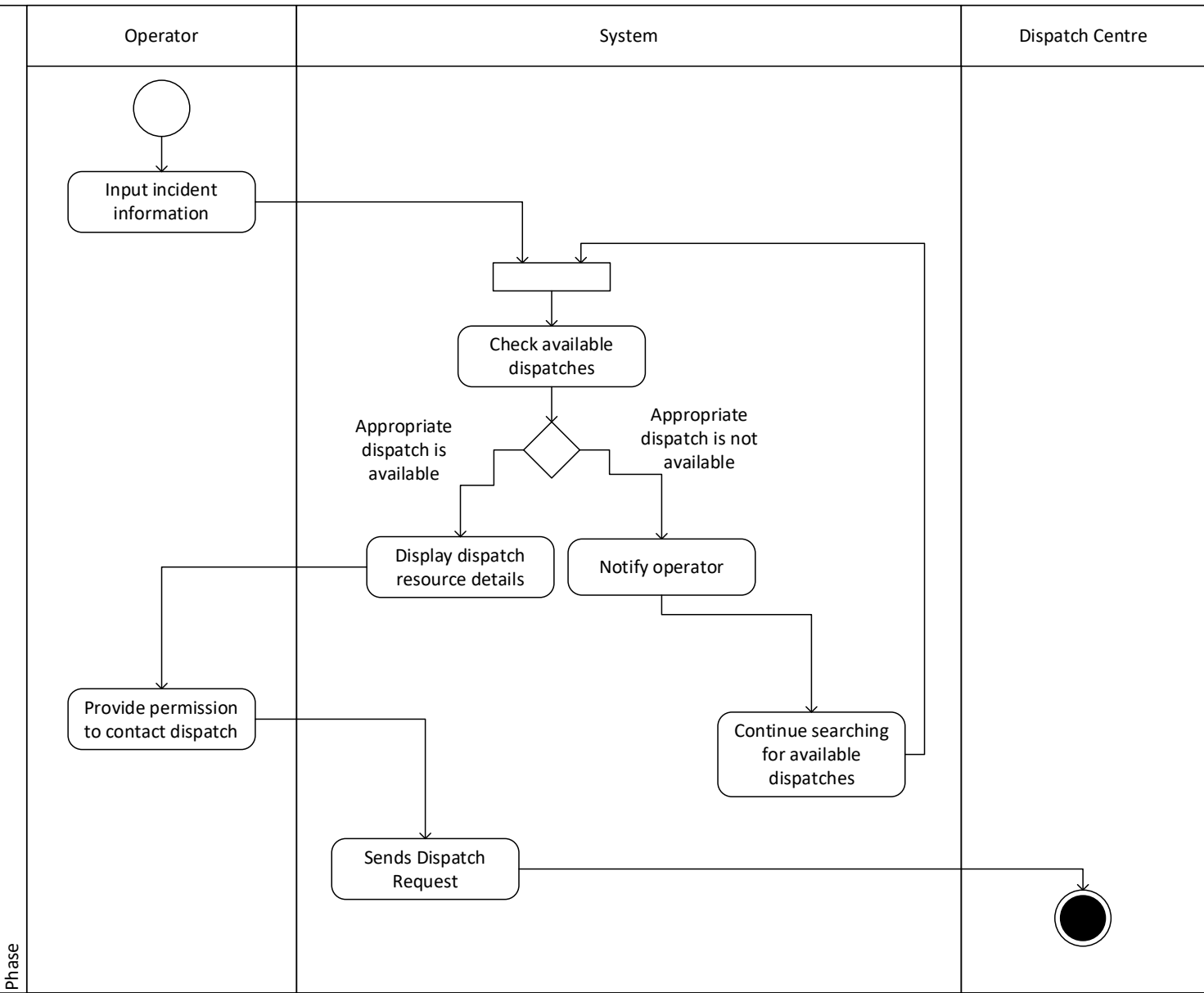
Operator

<<system>>

Dispatch Centre



Search Dispatch (Zane)



Activity Descriptions

CHOOSE DISPATCH (Kaleb)

This diagram starts with the operator entering the location and type of emergency, which the system receives and searches for dispatches closely matching the inputted parameters. The system then loops if no dispatches are available, or if there are it displays a list of appropriate dispatches. The operator then selects a dispatch and confirms their selection, which is forwarded to the Request Dispatch use case.

SEARCH FOR DISPATCH (Zane)

This diagram begins with the operator inputting information in regard to an incident. The system will then begin to search for any available dispatches that are appropriate for the incident outlined by the operator. If an appropriate dispatch is not available, the system will notify the operator of this, and continue to search for an available dispatch, unless the operator decides to cancel the search. If an appropriate dispatch is found, the system will display its findings to the operator and requests permission from the operator to contact the dispatcher. Once permission is given, the system will send a dispatch request to the dispatcher, including any information the dispatcher may require.

Make report (Jake)

The diagram begins with a human reporter making contact. The system will then attempt to obtain location and start a recording the operator will be informed of the reporter and begin communication the operator will then ask for location of the incident and the details of the emergency to which the human reporter replies to a fork in the road then appears in which either contact is ended or the operator inputs the incident and a new report or an update to an report realigning for both steps as the recording then ends and a log is created.

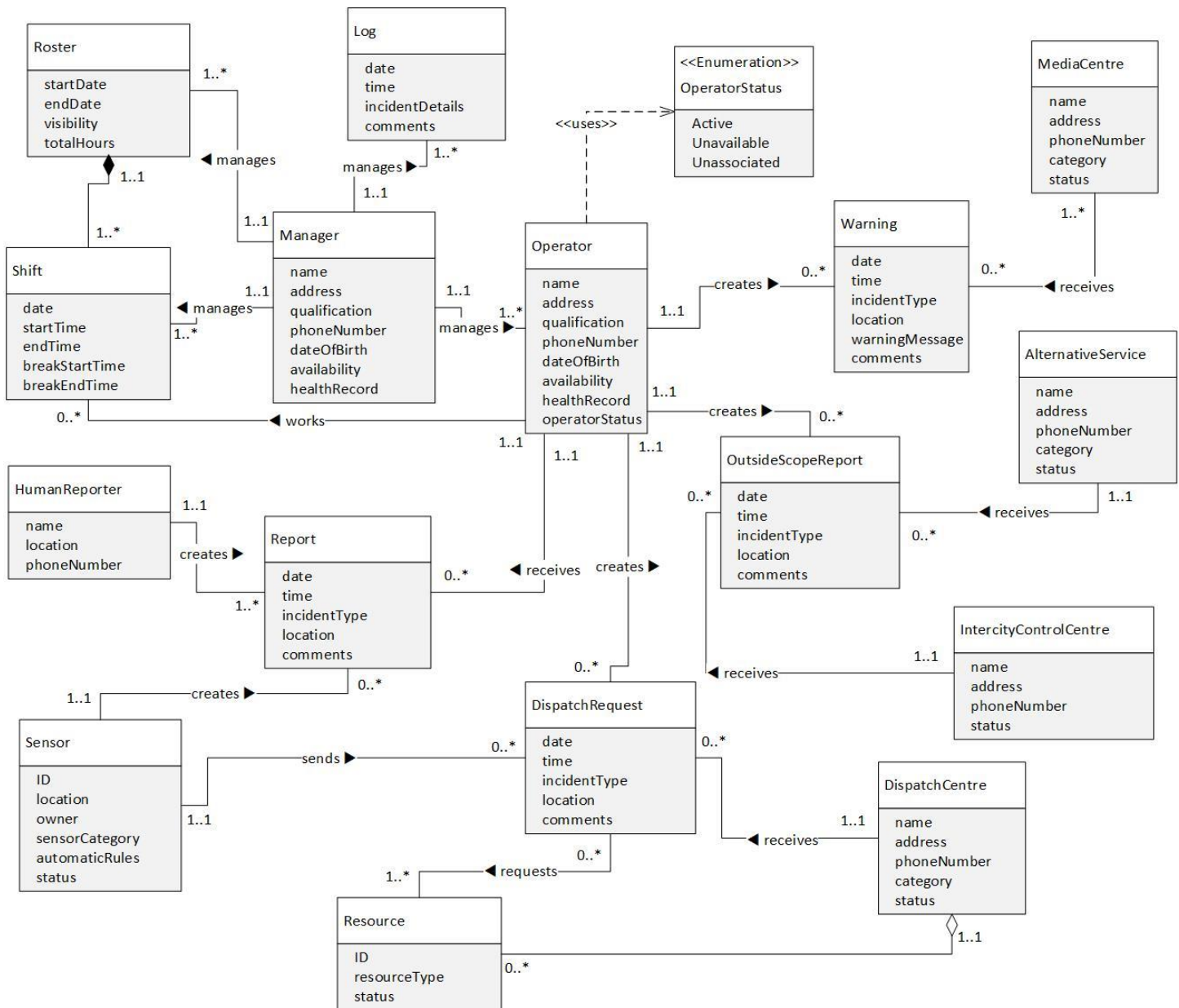
Report Alert (Deniel)

This diagram represents the activity flow for a sensor making an emergency incident report. When the registered sensor detects an emergency incident, it sends an incident report to the system. The system then identifies whether the sensor has any registered automatic dispatch rule. If there is none, the system notifies a human operator about the incident report. The operator then will look up the registered information of the sensor, and the system displays the relevant information. If the sensor has a registered automatic dispatch rule, the system will determine whether the rule has been met in this instance. If it has been met, a dispatch request is automatically sent to the relevant dispatch centre. Otherwise, the alert is ignored.

Roster Creation (Ethan)

This diagram begins with a manager signing onto the rostering system. Once the system verifies the manager it presents a calendar as well as each staff member and the hours they are working, the manager will then assign days and hours to specific workers. In the event of a staff member being assigned a shift that they have requested off or will place them in overtime the system will require authentication before progressing. Once all changes are finalized, the manager will check a box to save and upload the roster where it will be accessible for all staff to view.

Domain Class Diagram



Class	Description
Operator	An employee working under the Triple S centre who receives emergency incident reports and coordinates these reports to the appropriate entities.
OperatorStatus	An enumeration class showing the status of an operator, stating whether they are active (currently working for Triple S and available to work), unavailable (employed under Triple S but unavailable i.e. on leave), or unassociated (formerly employed under Triple S).
Manager	The individual who manages the operators and administrative operations of Triple S, including the creation and update of rosters, the management and audit of logs, and acting as a Local Emergency Operations Controller.
Roster	A roster class that depicts one fortnightly roster in Triple S, including details such as the start and end date of the fortnight, the visibility of the roster (whether operators are able to see it on their employee pages), and the number of total hours of all the shifts within that roster.
Shift	A shift worked by an operator that contains the start time, end time, and break times for the operator.
Log	A log/report that contains details of actions done by an operator, details of the incident, and the time and date when the log was created.
Report	An emergency incident report sent to an operator of Triple S from a human reporter or from a registered sensor.
HumanReporter	An individual who makes an emergency incident report to Triple S via phone or radio.
Sensor	A sensor registered to the Triple S system that can alert operators and dispatch centres about emergency incidents.
Warning	A report created by a Triple S operator that contains guidelines, advisories and notices about any major emergency incidents.
MediaCentre	Media centres, such as radio and television stations, that receive warnings from Triple S and can broadcast these warnings to the public.
DispatchRequest	A request made by an operator (or sometimes by a registered sensor) that contains a

	requested amount of resources as well as the details of an emergency incident.
DispatchCentre	The emergency dispatch centres (such as fire departments, police stations, or hospital emergency services) that receive dispatch requests from Triple S.
Resource	The resources (including human resources, vehicles, and equipment) owned by emergency dispatch centres.
OutsideScopeReport	A report created by an operator that lists details for an emergency incident that goes outside the scope of Triple S (such as incidents handled mainly by another emergency control center, or incidents that reach past the city jurisdiction of Triple S).
AlternativeService	An alternative service that handles specific emergencies not normally handled by Triple S (such as aviation emergency control centres).
IntercityControlCentre	An emergency control center found in other cities outside of the scope of Triple S.

Team Management and Meeting 1

SENG2130 Systems Analysis and Design

Minutes of meeting

Team__Blue Hedgehog____ Place__Zoom_____
Date/Time__3/3/2022 6.06pm_____

In attendance

All five

Apologies

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Agenda items (as needed)
- Date, time and place for next meeting
- Matters for consideration at next meeting

Action sheet

Task	Responsible	Due	Notes
Finalize the Pre-Action plan	Everyone	End of the meeting	
Outline tasks and task distribution for Gantt Chart	Everyone	Until next lab	
Finalize regular meeting time	Everyone	End of the meeting	
Discuss group dynamics and strengths	Everyone	End of the meeting	

Meeting 2

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog Place: Engineering F Meeting Room
Date/Time: 7 March 2023, 6:05 PM – 6:49 PM
Meeting length: 44 minutes

In attendance

Everyone

Apologies

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Fix pre action plan
- Discuss completed work
- Finalise Gantt chart
- Assign work for the week
- Matters for consideration at next meeting

Tasks completed

- Fix pre action plan (completed at 6:18 PM)
- Discuss completed work (completed at 6:45 PM)

Action sheet

Task	Responsible	Due	Notes
Finalise Gantt Chart	Deniel	12 th 6:00pm	Will be discussed on meeting on Sunday
Organise business Rules	Ethan	12 th 6:00pm	Make sure there are no duplicates.
Use Case Diagram draft	Jake	12 th 6:00pm	
Finish business rules	Zane	12 th 6:00pm	

Meeting 3

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog

Place: Zoom

Date/Time: 12 March 2023, 6:00 PM – 6:40 PM

Meeting length: 40 minutes

In attendance

Jake

Zane

Kaleb

Deniel

Ethan

Apologies

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Agenda items (as needed)
- Date, time and place for next meeting
- Matters for consideration at next meeting

Notes

- Review progress of tasks assigned from last Tuesday
- Assign tasks to be completed before the next meeting on Tuesday
- Discuss any comments or concerns about the

Action sheet

Task	Responsible	Due	Notes
Review new additions to Gantt Chart	Deniel	Tuesday	
Review and progress report on Business Rules	Everyone	Tuesday	
Review progress on Use Case Diagram	Jake, Kaleb	Tuesday	
Discuss changes to the business rules	Jake, Kaleb	Tuesday	

Meeting 4

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog Place: Engineering F Meeting Room
Date/Time: 14 March 2023, 6:05 PM – 6:49 PM
Meeting length: 44 minutes

In attendance

Everyone

Apologies

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Finalise use case diagram
- Address any questions
- Update Gantt chart

Action sheet

Task	Responsible	Due	Notes
Use case description draft	All	21/03/23	Kaleb might need additional time
Activity diagram draft	All	21/03/23	Kaleb might need additional time
Email Keiran Questions			- Do generalizations connect to parent use case, or can

Meeting 5

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog

Place: Zoom Date/Time: 19 March 2023, 6:00

PM – 6:40 PM

Meeting length: 40 minutes

In attendance

Jake

Zane

Kaleb

Deniel

Ethan

Apologies

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Agenda items (as needed)
- Date, time and place for next meeting
- Matters for consideration at next meeting

Notes

- Review progress of tasks assigned from last Tuesday
- Assign tasks to be completed before the next meeting on Tuesday
- Discuss any comments or concerns about the

Action sheet

Task	Responsible	Due	Notes
Review and progress report on Fully-developed Use Case Descriptions	Everyone	Tuesday	
Review and progress report on individual Activity Diagrams	Everyone	Tuesday	
Finalize Use Case System	Everyone	Tuesday	
Discuss changes to list of use cases	Everyone	Tuesday	

Meeting 6

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog

Place: Library

Date/Time: 21 March 2023,

6:05 PM – 6:40 PM

Meeting length: 35 minutes

In attendance

Everyone

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Agenda items (as needed)
- Date, time and place for next meeting
- Matters for consideration at next meeting

Notes

- Potential overlap between use cases, group came to an agreement by switching one to manual dispatch.
- Will meet on Sunday

Action sheet

Task	Responsible	Due	Notes
Use case diagram and Introduction	Ethan	Next lab	
Domain class diagram	Zane and Daniel	Next lab	
Use case diagram and subsystem.	Kaleb	Next lab	
Subsystem	Jake	Next lab	

Meeting 7

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog

Place: Zoom Date/Time: 26 March 2023,

6:00 PM – 6:40 PM

Meeting length: 40 minutes

In attendance

Jake

Zane

Kaleb

Deniel

Ethan

Apologies

N/A

Absent

N/A

Agenda

- Matters arising from previous meeting
- Agenda items (as needed)
- Date, time and place for next meeting
- Matters for consideration at next meeting

Notes

- Review progress of tasks assigned from last Tuesday
- Assign tasks to be completed before the next meeting on Tuesday
- Discuss any comments or concerns about the

Action sheet

Task	Responsible	Due	Notes
Review and progress report on Introduction	Ethan	Tuesday	
Review and progress report on Domain Class Diagram	Zane, Deniel	Tuesday	
Finalize use case diagram subsystems	Everyone	Tuesday	
Review first draft of Conclusion	Zane	Tuesday	

Meeting 8

SENG2130 Systems Analysis and Design

Minutes of meeting

Team: Blue Hedgehog
5:50 PM – 6:40 PM

Place: Library

Date/Time: 28 March 2023,

Meeting length: 50 minutes

In attendance

Jake
Zane
Kaleb
Deniel

Apologies

Ethan

Absent

N/A

Agenda

- Matters arising from previous meeting
- Agenda items (as needed)
- Date, time and place for next meeting
- Matters for consideration at next meeting

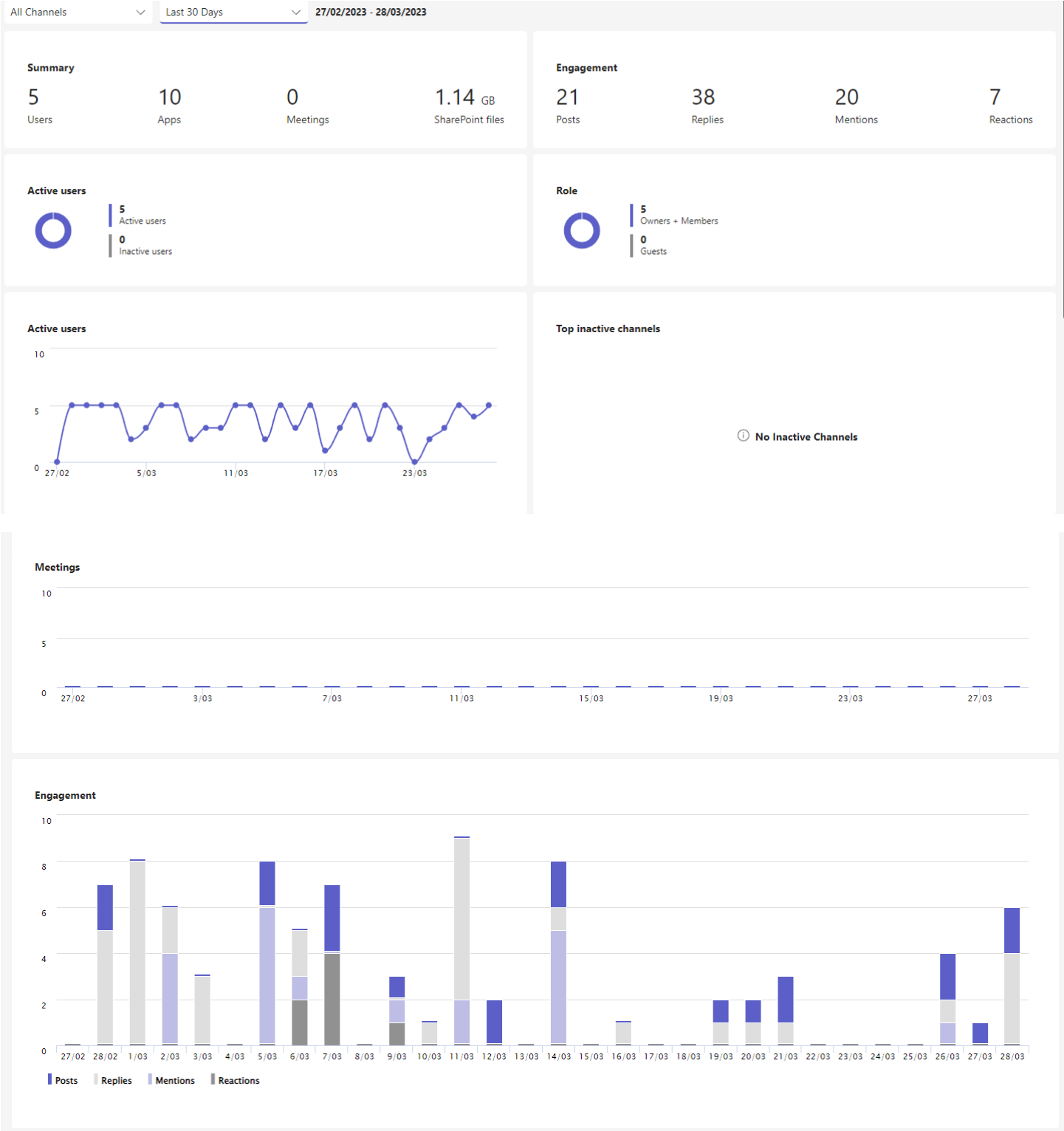
Notes

- Recommendations for the organisation
- Review spacing and relationships for Domain class diagram

Action sheet

Task	Responsible	Due	Notes
Domain Class Diagram	Deniel Zane	Thursday	
Introduction	Ethan	Thursday	
Compile into Report	Jake	Thursday	
Conclusion	Zane	Thursday	

Teams Analytics



Conclusion

To combat the long-running issues of slow responses and poor communication that have plagued the city of Nuvalis' emergency services infrastructure, Triple S has been established to coordinate and manage the operations of all emergency services under the same organisation to achieve this; a new system has been designed and laid out within this report. This system has been designed with the capabilities to conduct every aspect of the facility, with a heavy emphasis on attention to detail and precision, while also taking care to minimise obscurity where possible.

The system is intended to handle most of Triple S's necessary operations and functions, including creating and managing staff rosters, receiving, storing, analysing and distributing emergency report information and dispatching appropriate emergency services workers to valid incidents. As safety is a critical aspect of this organisation, and any failure may harm a person or group of people, it is paramount that errors are always avoided. As such, the system is designed to accommodate for and remain operational during any potential irregularities encountered during operation.

This report has outlined a comprehensive collection of lists, diagrams and tables concerning the developed system to minimise obscurity and effectively convey the system's functionality and uses. Firstly, a collection of business rules that outline the relevant work, health and safety, ethics, security, and privacy legislation and standards that apply to Triple S and serve to provide the users of this system with clear instructions on how to operate smoothly and without issue. Next, several use cases were created, including explanations, descriptions and diagrams documenting various crucial functions that the system will do. Use case descriptions for the functions involving the creation of a weekly roster, reporting of alerts, creation of a report, and the dispatching of appropriate emergency workers have been emphasised, as these functions are among the most important to be carried out by the system. Activity diagrams visualising the intended flow of these use cases have also been created. Finally, a domain class diagram was constructed to demonstrate a clear understanding of the system's relationships between various features and functions.

This report has achieved the goals outlined, including the extensive, detailed and precise requirements. By prioritising detail and clarity, the system has been outlined to avoid assumptions and allow for consistent and correct use by staff. As indicated previously, business rules outline clear instructions to users of the system on how to use the system without issue. Use case diagrams and their associated descriptions achieve a similar goal: supply further detail as associated use cases indicate the natural flow of a particular function and potential alternative flows. Finally, the provided domain class diagram indicates the relationships between system features and how they are intended to interact. This report has detailed a firm overview of the intended roles and functions of all system areas, including the positions staff of Triple S hold within the overall system and how to interact with it correctly.

References

1. Fire and Rescue NSW Act 1989.
2. Health Services Act 1997.
3. Telecommunications (Consumer Protection and Service Standards) Act 1999.
4. Telecommunications (Emergency Call Service) Determination 2019.
5. Work Health and Safety Act 2011.