Problem Domain Assessment of the 'Triple S' Emergency Response System

Member Names:

Lab Day:

Lab Room: EF108

Lab Time: 10:00am-12:00pm

Team Name:

Lab Academics:

Sample only:

This submission has been edited to show only partial examples of each section.

For example, the domain class diagram has most of the classes removed.

Use this to get a general idea, but do not copy. This group scored very well, but there were still some mistakes made.

Contents

2 Business Rules

Considering the crucial operations performed by Triple S and the complex nature of the proposed system, the business rules must be clearly defined. Research has been conducted into the legal, workplace, and ethical policies surrounding the system in addition to exploration of the project brief. Following on from this, the following business rules have been developed.

2.1 General Business Rules

1. Any emergency response centre must be staffed 24 hours per day, every day of the week.
2. If an emergency reporter gives a different location to the one given by the GPS reading, both locations are stored but the one provided by the caller should supersede the GPS.
3. If an operator or manager is employed as a full time, part time or casual worker, they can only work a maximum of 38 hours per week regularly.

3 Use Cases

A use case represents a system function or sequence of actions that provides observable value to an actor. Use cases are key to understanding the functional requirements of the Triple S system because they clarify the system boundaries and clarify how the Triple S system can be used. For this project, the Triple S system has been broken down into five subsystems that support semantically related use cases. These subsystems are called “sensing”, “sending”, “saving and communicating”, “managing”, and “logging”. The *sensing subsystem* is responsible for collecting and storing information about emergencies. This includes all use cases related to reporting an emergency and registering a sensor. The *sending subsystem* is responsible for requesting and dispatching emergency services to an emergency. The *saving and communicating subsystem* is responsible for coordinating communications between Triple S, emergency response services, other emergency control centres, and other services like the media. The *managing subsystem* is responsible for the internal management of Triple S operators and scheduling. Finally, the *logging subsystem* is responsible for accessing and recording all the data that Triple S is required to store.

3.1 List of Use Cases

3.1.1 List of Sensing Use Cases

Report Emergency – any person or registered sensor may report an emergency with the help of an operator. This always includes supplying two pieces of information to the system: the emergency location and type.

Find Emergency Location – the operator attempts to locate an emergency reporter or sensor by using one of many tracking services provided by the system.

> **Find Location via Radio Triangulation** – the operator attempts to locate a radio broadcast by using a radio triangulation service.
>
> **Find Location via GPS** – the operator attempts to locate a mobile phone call or GPS enabled sensor using a GPS service.
>
> **Find Location via Addressbook** – the operator attempts to locate a landline call using an address book.

3.2 Use Case Diagrams

The use cases for all five subsystems (sensing, sending, saving and communicating, managing, and logging) have been diagrammatically represented below, starting with the overall use case diagram which represents the Triple S system at the highest level of abstraction. Note that some subsystems involve use cases from other subsystems. Therefore, colour has been used to group semantically similar use cases. Red relates to emergency reporting, yellow relates to logging, green relates to dispatching, purple relates to communication, and blue relates to management.

3.2.2 Sending Use Case Diagram

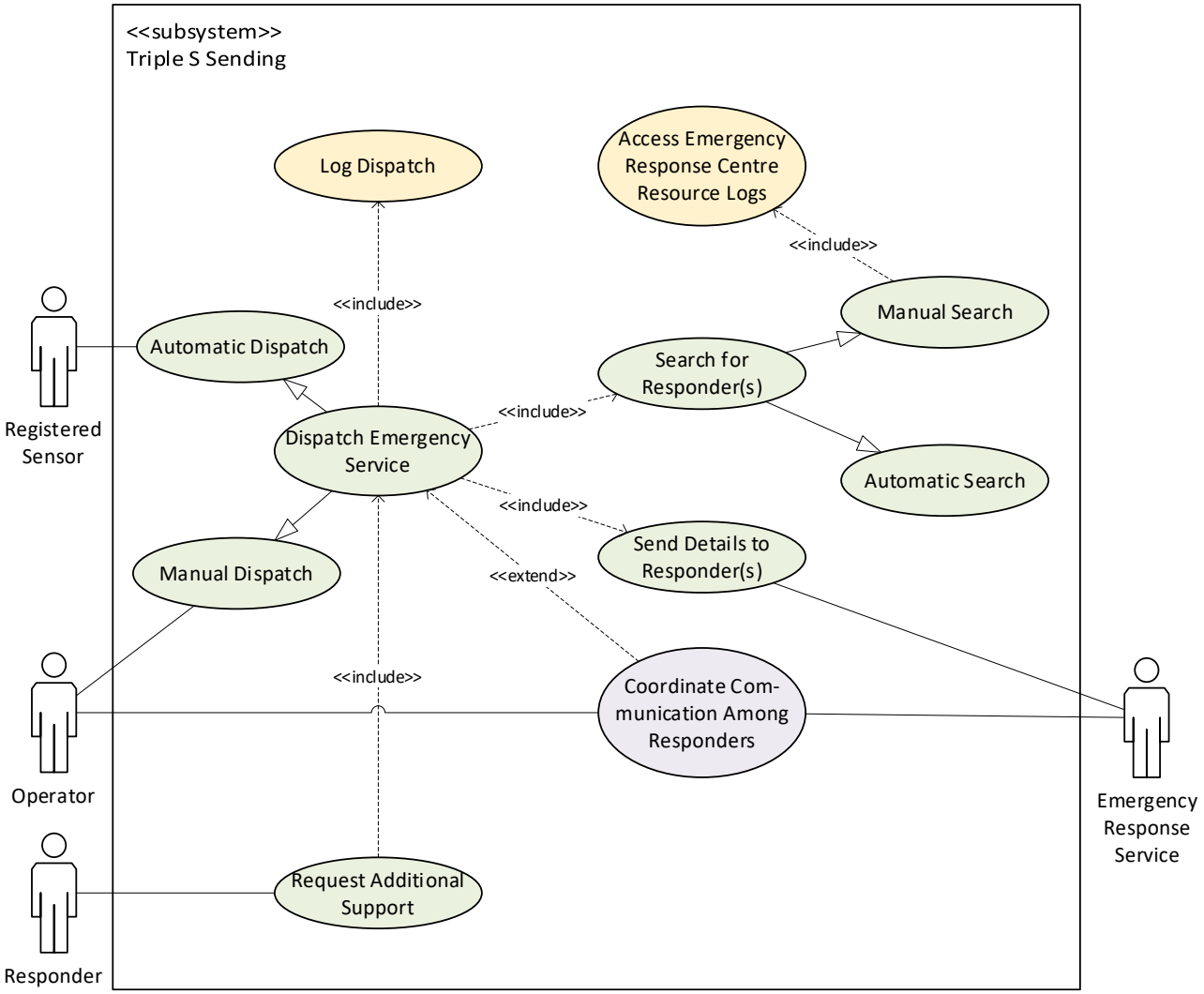


Figure 3 - Triple S Sending Subsystem

3.3 Full Use Case Descriptions

Full use case descriptions and their associated activity diagrams are required to provide a detailed model of how a use case is performed. This report has provided full use case descriptions for five key use cases as requested by the client: “Report Emergency via Phone,” “Report Emergency via Sensor,” “Register New Sensor,” “Manual Dispatch,” and “Generate Roster.”

3.3.5 Generate Roster Description

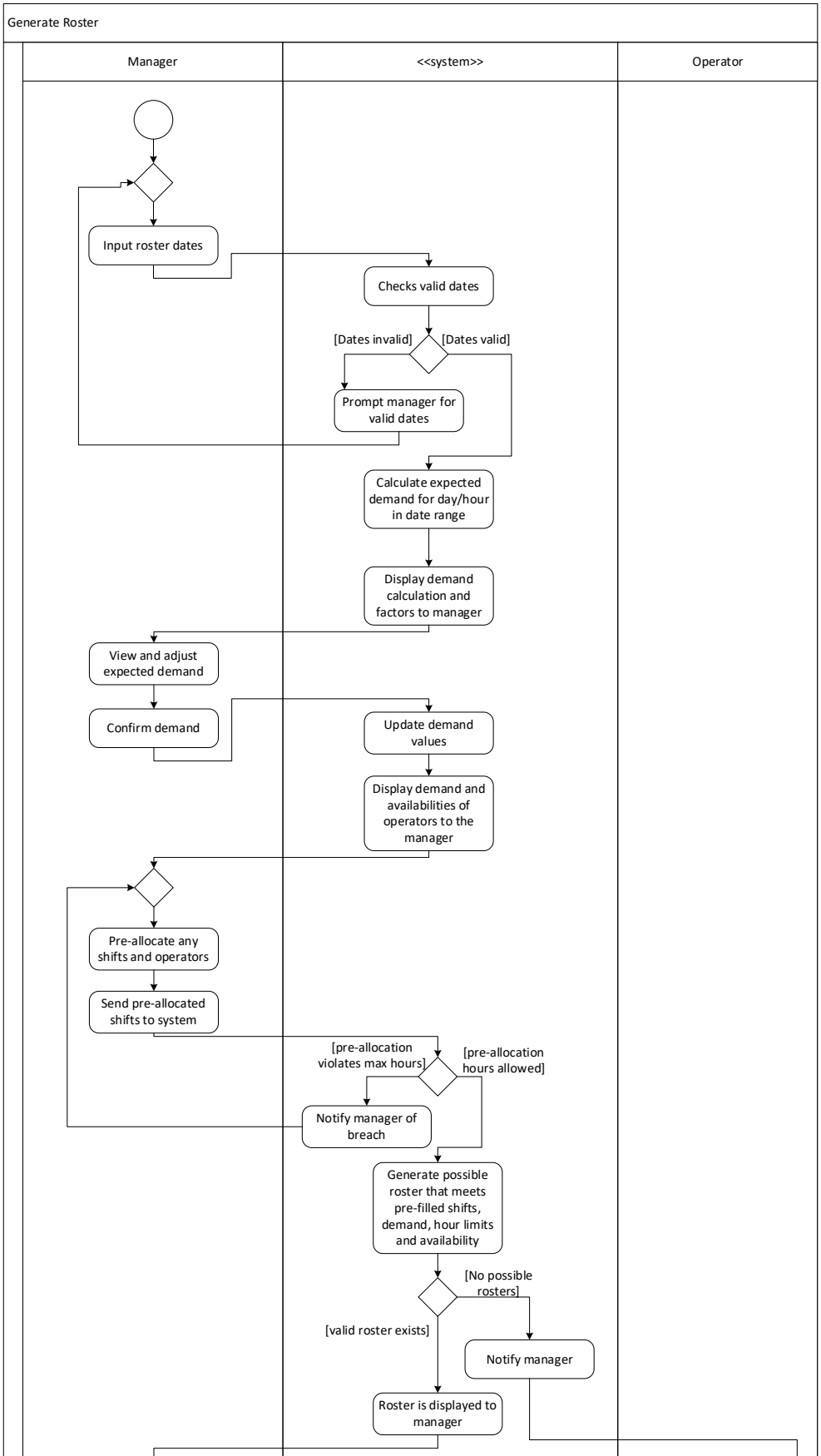
Table 5 - Generate Roster (Student Name)

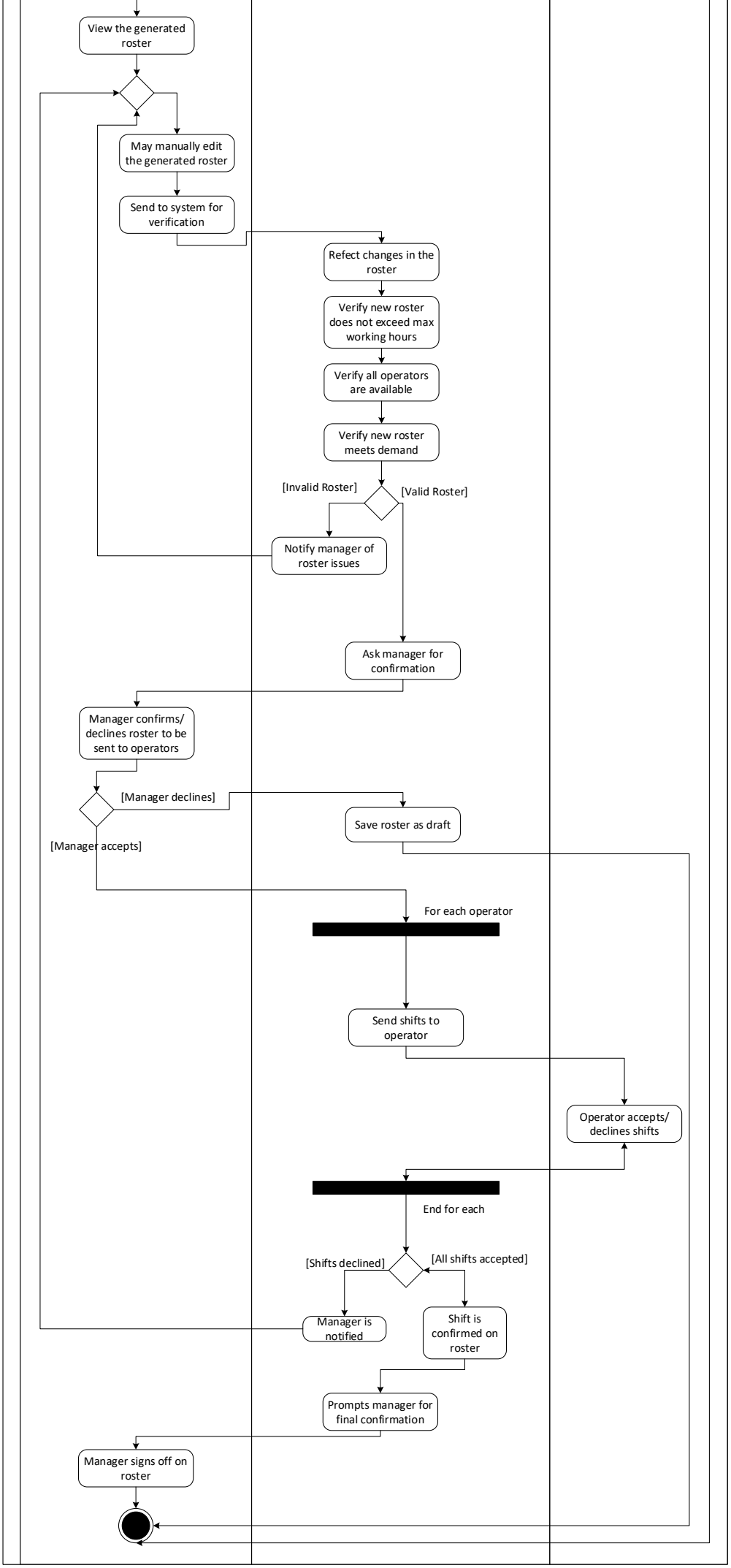
Use Case Name	Generate Roster	
Scenario	Create a new roster for upcoming dates.	
Triggering Event	Manger requests the roster be generated for a certain time period.	
Brief Description	Involves the creation of the operator roster. The process is initiated by the manager and the roster is sent out to operators for confirmation.	
Actors	Operator, Manager	
Stakeholders	Payroll -roster once generated will be sent onto payroll. Triple S Executive – Roster is of interest to ensure they are meeting demand, but also not putting people on if they are not necessary. Legal – Ensuring workers do not work more than their maximum hours.	
Preconditions	1. Operators have entered their availabilities for the roster period. 2. Demand calculation factors have been entered into demand calculator.	
Postconditions	1. Digital/physical copy of the roster will be generated. 2. One or more operators are rostered at all times. 3. Operators have confirmed their shifts. 4. Manager has signed off on the generated roster.	
Flow of Activities	Actor	System
	1. Manager inputs dates for which the roster is to be generated. 2 Manager views and can edit the expected demand for each day/hour. 3 Manager confirms the demand. 4 Manager pre-allocates shifts and operators for these shifts.	1.1 System verifies the dates are valid and in the future. 1.3 System performs calculation of expected demand for each day/hour in date range. 1.4 System displays this calculation and factors used to the manager. 3.1 System updates demand values. 3.2 System displays demand and availabilities to the manager. 3.3 System generates all possible roster options that meet demand. 3.2 System refines this by removing options that force operators to work over their maximum hours. 3.4 System displays this to manager . 4.1 Verify pre-allocated shifts do not exceed max working hours. 4.2 System generates possible roster options that meet pre-fills, demand, max hours and availability. 4.3 Roster is displayed to manager

	5 Manager views and can manually edit the roster if necessary. This is sent to the system for verification. 6 Manager confirms the roster to be sent out to employees. 7 Operator accepts shifts. 8 Manager signs off on the roster.	5.1 The system reflects these changes. 5.2 System verifies the roster does not violate max hours. 5.3 System verifies that all operators are available. 5.4 System verifies new roster meets demand. 5.5 System asks manager for confirmation 6.2 System sends out roster to all operators. 7.1 System keeps a record and is confirmed on the roster. 7.2 Once all operators have accepted, manager is prompted for final confirmation.
Alternate Flow	4. Manager entirely pre-allocates all shifts. a. System will perform all validations, but will have no need to generate any parts of the roster. 6. One or more operators decline shifts a. System notifies manager. b. Manager alters roster. c. Updated shifts are sent to operators.	
Exception Conditions	4. No possible roster can be generated that meets roster constraints. a. The system will display a message to the manager. b. Roster will be generated, noting which timeslots are not covered. c. System exits prior to sending out roster. 6. Manager cancels before sending roster out a. System saves a draft roster. b. System exits prior to sending out roster.	

4.5 Generate Roster (Student Name)

The roster generation step is essential to ensuring Triple S is fully staffed and capable to handle a high volume of emergency calls. When generating the roster, the demand calculation should be presented to the manager as well as the factor used. From this information, it is important to allow the manager to pre-allocate shifts to ensure that the appropriate level of experience is maintained. From here, the system fills out the remaining roster based on demand, working hours and operator availability. The manager tweaks and then will publish shifts to operators. Once all operators confirm shifts, the manager signs off on it.




View the generated roster
May manually edit the generated roster
Send to system for verification
Refect changes in the roster
Verify new roster does not exceed max working hours
Verify all operators are available
Verify new roster meets demand
[Invalid Roster]
[Valid Roster]
Notify manager of roster issues
Ask manager for confirmation
Manager confirms/ declines roster to be sent to operators
[Manager declines]
[Manager accepts]
Save roster as draft
For each operator
Send shifts to operator
Operator accepts/ declines shifts
End for each
[Shifts declined]
[All shifts accepted]
Manager is notified
Shift is confirmed on roster
Prompts manager for final confirmation
Manager signs off on roster

5 Domain Class Diagram