**Motivation**

We will start by defining what is a rescue robot and describing the issues that calls the need for a smart rescue robot. So, what is a rescue robot? A rescue robot is defined as a semi-autonomous or fully-autonomous robot that is capable of search and rescue missions in specific situations. The examples of such situations are in earthquakes, building fires and drowning situation in seas or lakes. The general purpose of the rescue robot is for assisting rescue teams making it easier to complete their mission. There are several qualities or features that makes a robot a rescue robot. In our opinion, the basic general requirements of a rescue robot must suit the following criteria:

* Must be durable enough to withstand rough terrains
* Having the ability to move through one or more types of terrain
* Having the ability to communicate or send important information to rescue teams
* Having the ability to sense robot surrounding in order to avoid obstacles

Nowadays, further research and developments has been carried to push this new technology to the market as it will solve a lot of the problems regarding the safety of people.

Although we do have safety related people such as firemen and lifeguards, problems can still arise in a rescue situation, which could cause injury to the civilian (victim) or even cause death. One of the issues that could happen in a rescue situation is that humans are prone to make errors. We as humans are prone to have a poor performance in our task when we are stressed, under a lot of pressure or lack of concentration. Since all humans have their own personal problems in their lives, that means that rescue teams also will have their own personal problems, which could affect their performance during a rescue situation. Having a rescue robot to assist the rescue team could help prevent mistakes from happening during a rescue situation, thus saving more lives in the process.

Ignoring human errors that could occur during a rescue situation, rescue teams also need a lot of time to act in an emergency to ensure the safety of the majority of the people. They would need to set up safety gears and equipment in which during that time could be used to start the rescue mission. On top of that, the rescue team are not entirely invincible to the dangers of the rescue situation. They are also human and they still have the risk of injuring themselves or death while they are trying to save other people. The main mission is to save as much lives as possible in a rescue situation, including the rescue team. With the assistance of a rescue robot, these stated problems are no longer an issue, which could save more lives during an emergency situation.

Rescue robots that possess the qualities mentioned above may be able to solve said problems during safety missions. Firstly, rescue robot that are durable can move in dangerous paths, that would normally consume more time for the average rescue teams to move, without risking lives. Other than that, the robot can also send information regarding trapped civilians that are difficult to locate. For example, in a collapsed building, the robot can move in small holes and tight areas and help rescue teams locate victims that are trapped under a rock. This in turn will help rescue teams save more lives than before, which is the ultimate priority in a rescue situation.

We strive as a group to develop a rescue robot that could solve all these stated problems and provide safety to humans generally. As a result of our development, we managed to add features to our rescue robot that could solve the problems that were stated above.

Our rescue robot has the ability to travel across both land and water. This is extremely beneficial as it is compatible for the rescue team to use, regardless of the rescue situation. Burning buildings, earthquake emergencies, drowning situations, and other rescue situation could be assisted using the same robot. This also makes the robot more versatile in handling the rescue situation.

The mobility of our rescue robot is also very reliable. Not only is it versatile to move both on land and water, it could also break small obstacles if one is present in the situation, and detect humans to aid them during the rescue situation. This is what makes our rescue robot smart and reliable. There are also other functionalities to aid the rescue team which we are going to dive in deeper in the next section.

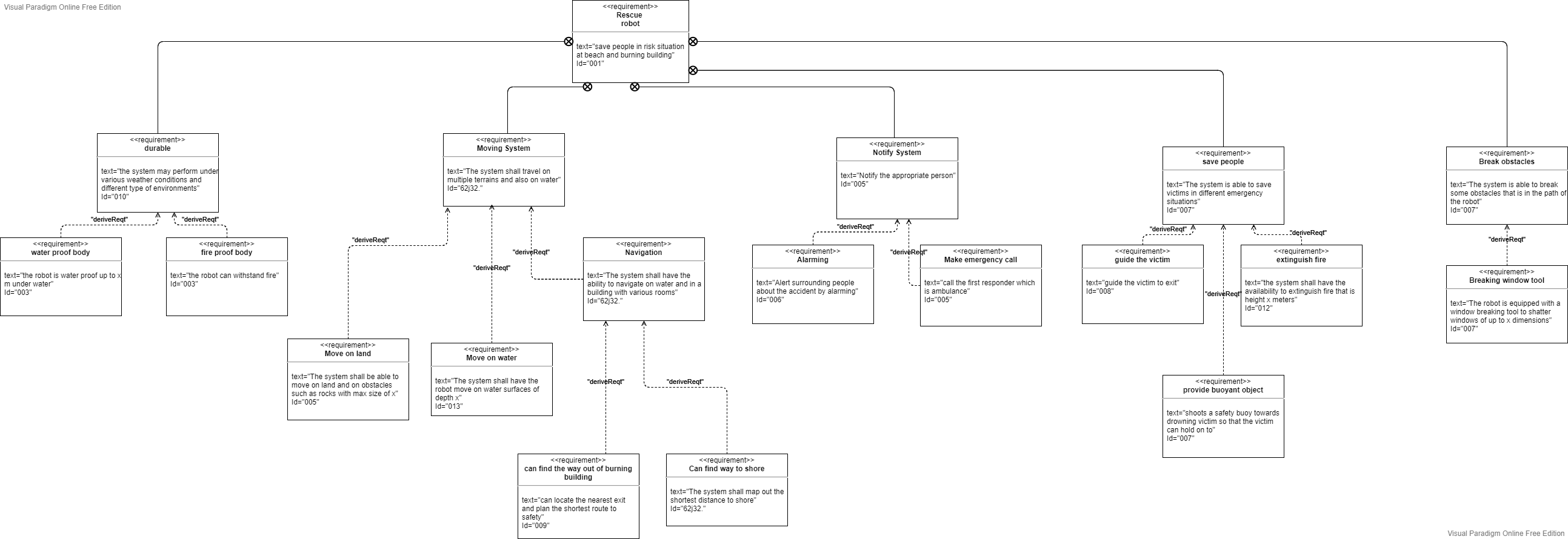
In summary, it is extremely beneficial to have the assistance of a rescue robot. With the help of a rescue robot, we can shorten the time to rescue a victim in danger, thus increasing the efficiency of a rescue mission and save a lot more lives in a critical situation. We can also avoid human errors and may even help save the lives of the rescue team if something unexpected happen.

**Introduction to diagrams**

The first and important step into the development process of our rescue robot is to precisely produce the analysis of the system context and define our problem as detail as possible. we need to have an overall description of our whole system before we start working on the robot itself. we modelled our system with the aid of SysML and UML diagrams and this is to help define the concrete scenario and to also establish boundaries between robot and the specific environment. therefore, we can determine the details of the features/abilities of the robot.

**Requirement diagram**

The first diagram that we will look into is requirement diagram. A requirement is defined as a condition or a capability needed by us user to solve a problem or achieve an objective. In order to determine which requirement is necessary to be implemented in our system and to avoid future waste, we have also included the specifics of the environment that interacts with our robot in our requirement diagram. For example, we included a breaker tool for our robot so that it can go through obstacles like windows with thickness of x cm. Further details of our final requirement diagram are as shown.

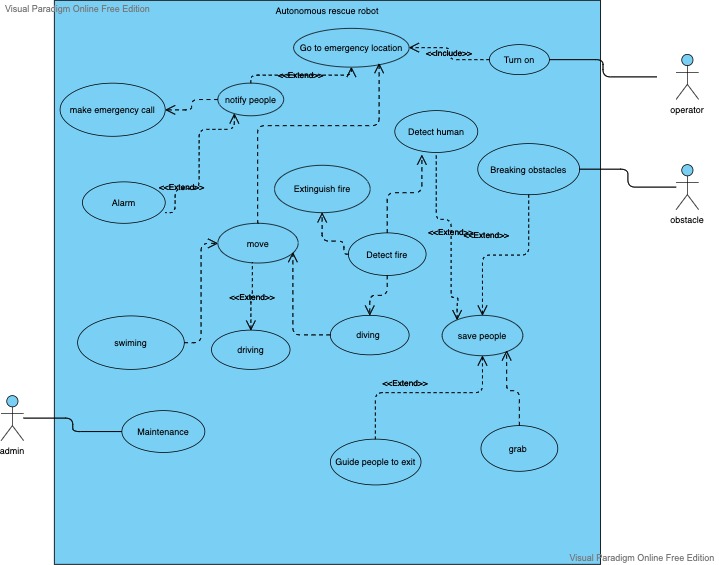


An interesting requirement to have a look at is the move system. We have included in our system, the ability for the robot to move on land with the specification that the robot can move height x meters and on water with depth of x meters. Furthermore, we also equipped our system with a navigation subsystem so that it can find the way out of burning building and find the shortest path to shore.

Another important aspect of our robot that needs to be considered is how the robot will notify the rescue team or a helping bystander when a civilian is located in the burning building or drowning in the sea. We have included a notification requirement for our robot so that when an emergency situation is recognised, the robot will sound an alarm alerting its surrounding. Other than that, the robot will also make an emergency call to alert authorities regarding the situation.

In the successive sections, we will provide the reader with scenario related diagrams such as Use case and Sequence diagrams to help visualise how the requirements are planned to be implemented in our rescue robot system. The diagrams mentioned will construct a sequential story for the application of said requirement diagram and also how the requirements act with each other and actors that interact with the system.

**Use case diagram**

To help us in the development process of the system of the rescue robot in a more effective and efficient way, we first need to define the concrete scenario that the rescue robot is involved in. The important factor in developing our system is determining the boundaries in which our robot interacts with the environment. In the first phase of our project, we have done our research and gathered information on rescue situations around the world. Using the knowledge that we gained about rescue situations, we could create a use case diagram and identify situations that our rescue robot will play a huge role when assisting the rescue team. As a result, we have decided to focus on two main rescue situations, which is burning buildings and drowning victims. The final use case diagram is showcased in figure \_\_

As you can see from the use case diagram, the operator will start the robot by turning the robot on in order for it to start operating. The robot will then move towards the emergency location in order to assist the victims of the rescue situation. Our robot will notify people nearby the emergency location with the help of its alarm. The robot will also make an emergency call in case more man power is required to deal with the current emergency situation.

In terms of mobility, our robot can move both on water and on land. It is also smart enough to detect humans, fire and obstacles. Using its intelligent sensor, the robot can navigate its way without being stuck if there is an obstacle in the way. In case it got stuck in between multiple obstacles, it can break the obstacle if it is small and fragile enough such as boulders or window glass. It can also find humans in a burning building, and save them by guiding them to the exit. When it detects fire, it has the ability to extinguish the fire and continue its journey. In a drowning situation however, the robot can grab the victim and save the drowning victim.

**Sequence Diagram**

In the following section, we continue our development process with the addition of a sequence diagram. With the sequence diagram, we can see the flow of use of the system in our robot given the context situation. More importantly, we again use the analysis of the requirement and all of the preceding diagrams that we have built so that we can design a sequence diagram for the overall system of the robot. We have developed two sequence diagrams according to their rescue situations, which is burning buildings and drowning situation.

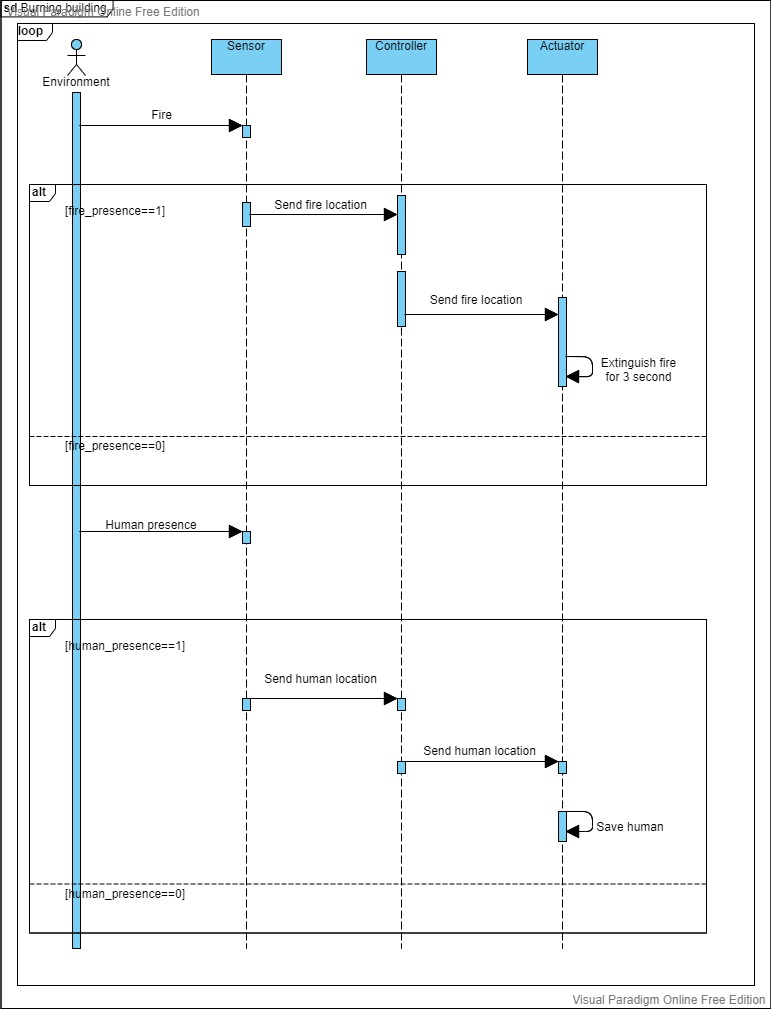


Figure x shows the sequence diagram of the burning building situation. The environment will send a trigger which is a fire in this case to the sensor of the robot. The robot will then update the fire\_presence variable depending on whether there is a fire or not nearby the robot. If the fire\_presence is 1, the sensor will send the fire location to the controller of the robot. The controller will then pass the fire location to the actuator, and the actuator will use that information to extinguish the fire for 3 seconds.

Then when the environment sends a signal that a human is nearby, it will send a trigger to the sensor to notify the robot. The sensor will then pass the human location to the controller, and the controller will continue passing the location to the actuator. The actuator then will move and save the human based on the location that it received.

The system will then be looped so that the robot keeps on updating the fire and human presence, thus helping the firemen to extinguish the fire and also helping humans if there are still humans stuck inside a burning building and guiding them to safety.

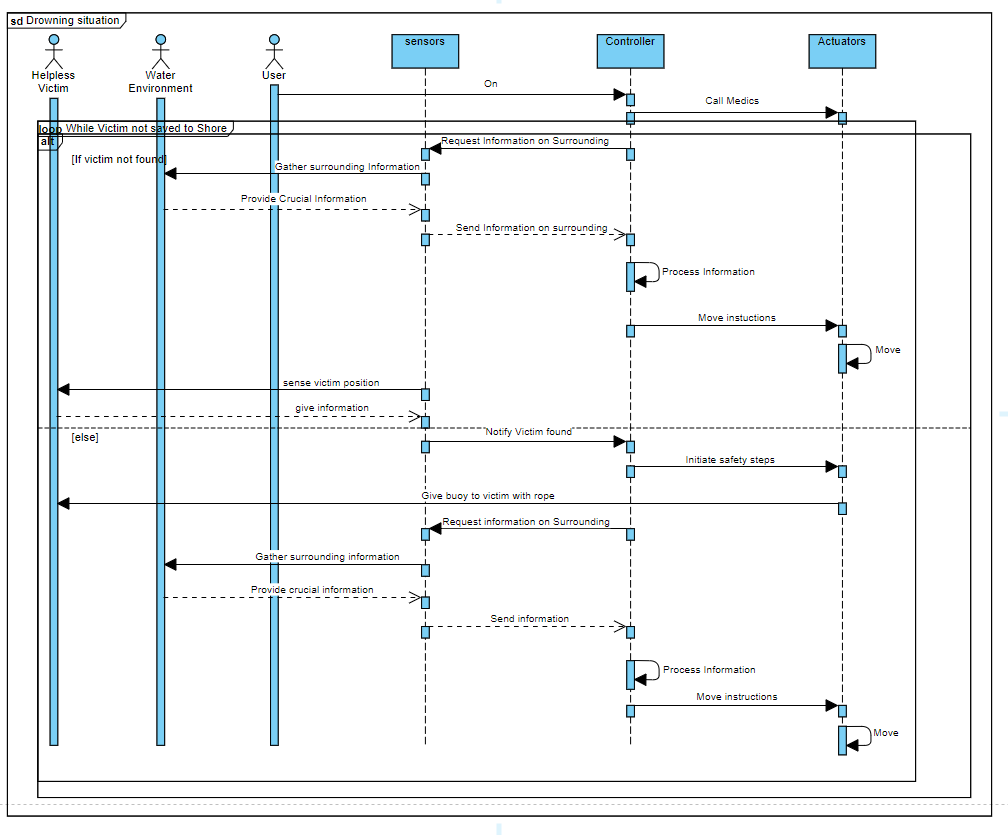
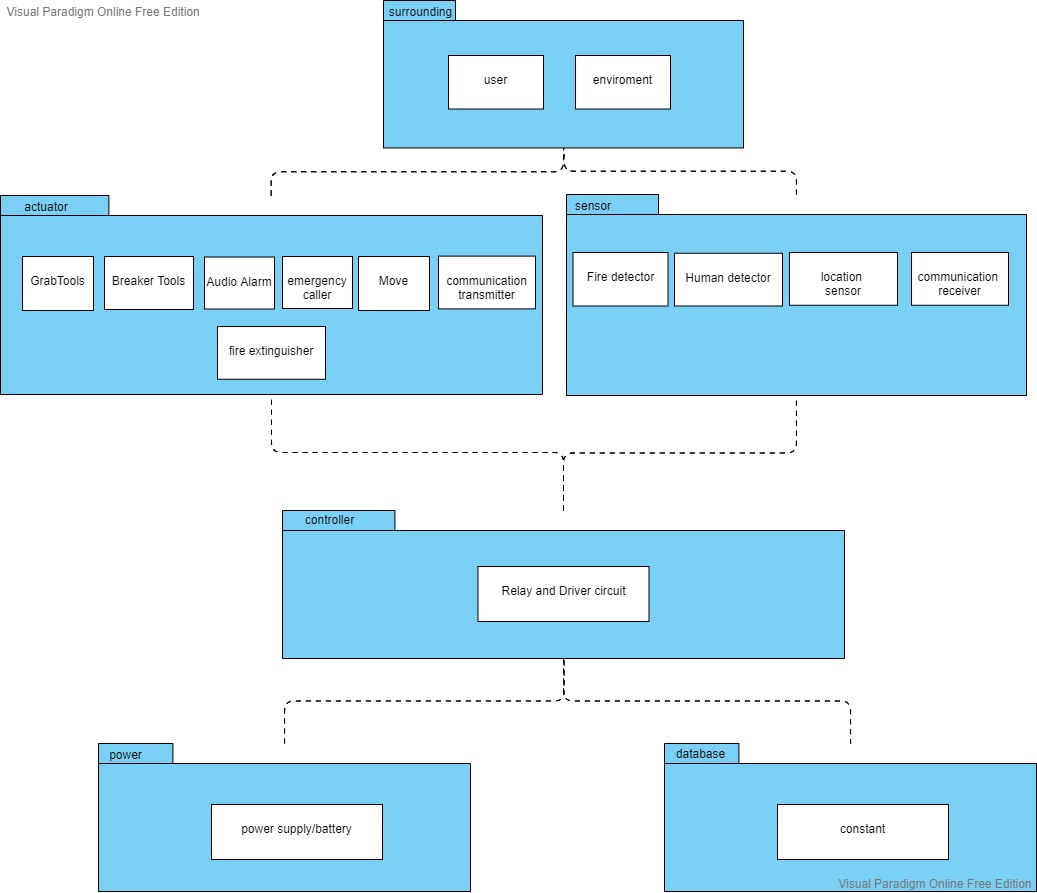


Figure x shows the sequence diagram of a drowning situation. The user or operator will turn the robot on for it to start its operation. It will then start its operation by calling the medics to prepare in case of an emergency after the victim has been saved. It will then enter a loop where it will use its sensors to request information of the surrounding. To be more specific, it will use the information gained from the surrounding to move towards the victim. It will then sense the victim position. If the victim isn’t found, it will loop again to move until it finds the victim.

Once the victim is found, it will start initiating safety steps, and shoots a buoy towards the victims that is attached to a rope with the robot. After the robot successfully saved the victim, it will use the same navigating mechanism to find its way back to the shore, and bringing the victim to safety.

**System structure**

The final diagram that we developed as a result of our previous diagram is our system structure diagram. The purpose of a structure diagram is to describe our system as a whole and give a clear bird’s eye view of our system. The type of system structure that we have chosen is layered application. In layered application, the components of the system are divided into horizontal layers. Each layer of the layered architecture pattern has a specific role and responsibility within the application. We will go in detail for each of the layers in the upcoming paragraph.



The system structure diagram is showcased in figure X. Our system consists of 4 main layers, which are the surrounding layer, the Y layer, the Z layer and the O layer. In the surrounding layer, we have included 2 components that will interact with our robot, which are user and environment. Both the components will act as an actor that will initiate our robot to do to operate different tasks. For example, the user will turn on the robot for it to start operating, and the environment could be a numerous number of triggers, such as fires, humans, water and obstacles, that could stimulate a response from the robot.

In the next layer, it consists of components from two part of the robot that directly interacts with the environment. There are various components in this layer, so we divided the components into two groups, which are the actuator and the sensor. In the sensor group, the components are responsible for getting information from the environment and passing it to the controller of the system. The components that are responsible for this are fire detector, human detector, location sensor, communication receiver. In the actuator group, the components are responsible for the response of the robot after it receives sets of instructions from the controller. The components that realise this purpose are grab tools, breaker tools, audio alarm, emergency caller, move, communication transmitter and fire extinguisher.

The third and arguably the most important layer of the system is the controller. It serves as the so-called brain of the system, because it is responsible for receiving the information from the sensor, managing or processing said information, and delivering sets of instructions to the corresponding components of the actuators. The controller can be realised by multiple relay and driver circuit.

The fourth and final layer is the database layer. The function of this layer is to store vital data that relates to the robot’s activity. We can realise this by a local storage drive in the robot. The information stored can be either the history of the robot’s action or the value of constant variables that has been pre-programmed such as the weight of the robot and the gravitational acceleration.

Citations

L. Zhu, G. Liu, Y. Zhao, C. Liu, M. Jiang and X. Li, "Design of Motion Control System of Rescue Robot Based on ARM," *2017 International Conference on Computer Technology, Electronics and Communication (ICCTEC)*, 2017, pp. 1183-1186, doi: 10.1109/ICCTEC.2017.00257.

A. Denker and M. C. İşeri, "Design and implementation of a semi-autonomous mobile search and rescue robot: SALVOR," *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, 2017, pp. 1-6, doi: 10.1109/IDAP.2017.8090184.