**UML Project**

**The Robot**

Below you can read the statement of the Project in Rhyme. You need to read carefully and take all the necessary notes. During each task, we will be as faithful as possible to our statement to create a UML class diagram based on a literal interpretation of it:

Task 2

We want to model a robot. A robot can be on or off. The robot knows from its sensors its current position using two-dimensional Cartesian coordinates and its orientation. Our robot can only make 90° turns, and therefore its orientation can only be North, South, East, or West.

Our robot must tell us its status (if it is on or off) and allow us to change its status. For example, it can report its position on X and Y, as well as its orientation. Our robot can also rotate 90° each time clockwise or anti-clockwise and advance a certain distance in the direction of its orientation.

Task 3

The company we work for wants to employ robots in different exploration scenarios. Each scenario has a name and is a rectangular construction that is defined by an upper corner and a lower corner expressed in X and Ycoordinates. The scenario is able to report its name and the information of its corners by returning the pair of coordinates of each one.

For the exploration of scenarios, several robots are used, such as those described in the previous task. Each robot can only go through one scenario at a time. As required, a scenario can be assigned robots and a scenario can inform which robots are in it. In turn, a robot knows the scenario it is touring.

On the other hand, for the use of robots, global tasks can be defined. Tasks consist of a description and a priority between 1 and 10. Both elements can be reported and modified for a task. Different robots can be assigned with the same task or have several tasks at the same time. Every robot must have at least one task assigned to it and must be able to assign or delete a new task. In addition, you must be able to report your tasks. From a task we can not get information about which robots are performing it.

Task 4

This time we will take into account the sensors that are part of our robots. Let's do a classification of the sensors.

All sensors have a unique identifier that they must be able to report, and each sensor must be able to transmit a flow of information, unique to each sensor, in the form of a data array.

There are only two types of sensors developed for these robots, the first being a visual sensor that corresponds to a camera. This sensor has a resolution given in megapixels that informs and, at the time of transmitting the flow of information, returns an image represented in an array of values of pixels in black and white.

On the other hand, the only other possible type of sensor is a proximity one with a minimum distance set, which is reported and can be changed. This sensor, when transmitting information, returns the distance in each orientation to the nearest obstacle (be it a wall or something else). Additionally, this sensor returns an alarm with the orientation in which there is an obstacle closer than the minimum proximity distance.

Task 5

One of the fundamental pieces of robots are their sensors. Each robot is built with various custom sensors, which are part of it. There is no limit to the number of sensors a robot can have. A robot can report which sensors it has, sensors can be added to it or removed. A sensor can inform which robot it is associated with.

At the same time, the company has decided to build scenarios with a higher level of complexity. Each scenario will be composed not only of an area, but of obstacles unique to that scenario. An obstacle is defined as a list of X and Y coordinates that make up a polygon, that are within a scenario and that can be reported. Such obstacles can be created and removed for a particular scenario, although they can only be accessed from the stage. To a scenario we can also ask you to list your obstacles.

Also, you want to model teams. Teams are made up of robots and you can add robots to different teams if you want, however, a robot can only belong to one team at the same time, or it may not be part of any. Teams have a name, which they must be able to report or modify, and it must be possible to add or remove robots from a team, as well as give a list of robots that are part of it. At the same time a robot must be able to inform which equipment it belongs to, if at all.

Finally, a scenario must now also be able to report which teams are present in it.