Robotic Vacuum Cleaner

## Stakeholders

End user

Company that develops (software eng, mecha engineers, eletrical engineers, marketing people, safety engineers)

Resellers (Auchan.. )

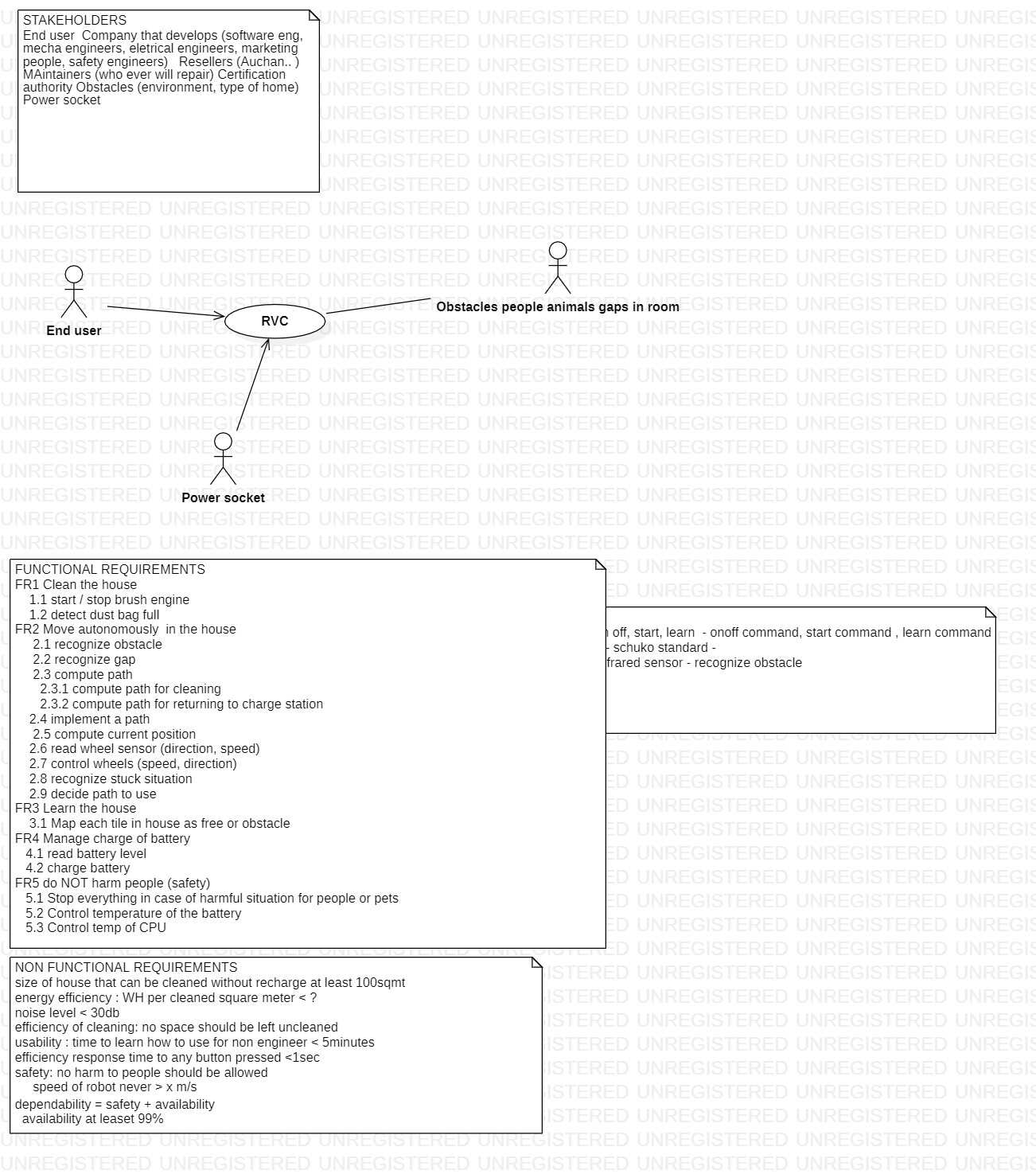
MAintainers (who ever will repair)

Certification authority

Obstacles (environment, type of home)

Power socket

## Context diagram



## Interfaces

|  |  |  |
| --- | --- | --- |
| Actor | Phys interface | Logical interface |
| End user | On off button  Start button  Learn button | On off command  Start command  Learn command |
| Power socket | Shuko standard socket | 220V 50hz |
| Obstacle | IR sensor | Recognize obstacle / not obstacle |
|  |  |  |

## FR

FR1 Clean the house

1.1 start / stop brush engine

1.2 detect dust bag full

FR2 Move autonomously in the house

2.1 recognize obstacle

2.2 recognize gap

2.3 compute path

2.3.1 compute path for cleaning

2.3.2 compute path for returning to charge station

2.4 implement a path

2.5 compute current position

2.6 read wheel sensor (direction, speed)

2.7 control wheels (speed, direction)

2.8 recognize stuck situation

2.9 decide path to use

FR3 Learn the house

3.1 Map each tile in house as free or obstacle

FR4 Manage charge of battery

4.1 read battery level

4.2 charge battery

FR5 do NOT harm people (safety)

5.1 Stop everything in case of harmful situation for people or pets

5.2 Control temperature of the battery

5.3 Control temp of CPU

## NFR

size of house that can be cleaned without recharge at least 100sqmt

energy efficiency : WH per cleaned square meter < ?

noise level < 30db

efficiency of cleaning: no space should be left uncleaned

usability : time to learn how to use for non engineer < 5minutes

efficiency response time to any button pressed <1sec

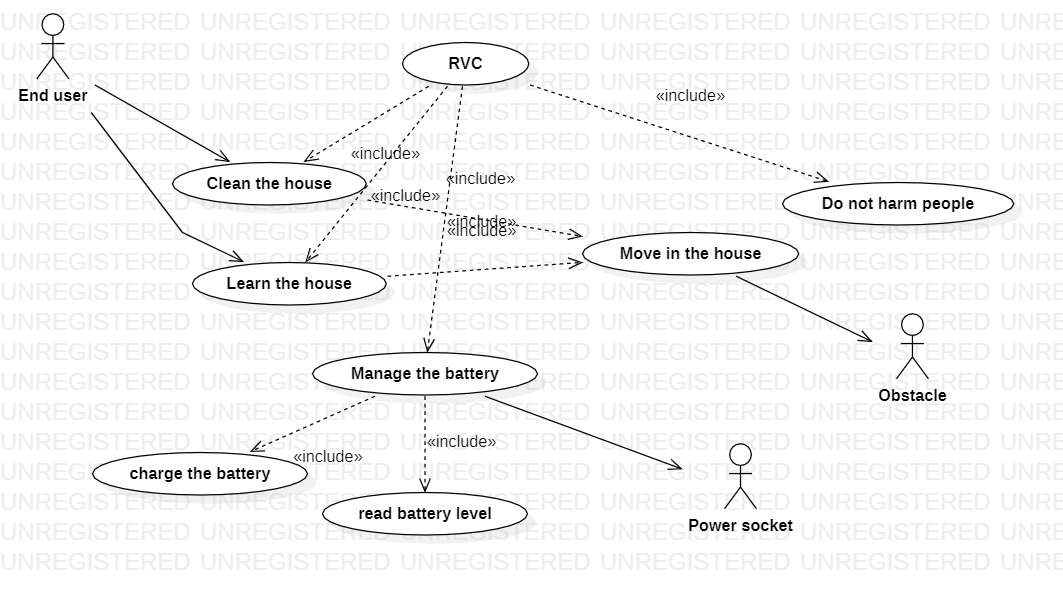
safety: no harm to people should be allowed

speed of robot never > x m/s

dependability = safety + availability

availability at least 99%

## UCD



### UC1

Use case ‘Clean the house’ UC1

|  |  |
| --- | --- |
| PRECONDITION | Robot is ON  Battery is not empty (or above a certain threshold)  Learn procedure completed (house is mapped and a cleaning path is defined) |
| POST CONDITION | Robot is back on charging station |
| ACTORS INVOLVED | End user |

Scenario UC1-1 (NOMINAL case)

|  |  |
| --- | --- |
| PRECONDITION | Robot is ON  Battery is charged and enough for cleaning all the house  Learn procedure already completed  dust bag is empty enough to clean all house – problem (OMISSION) we don’t have a dust bag sensor |
| 1 actor end user pushes ‘start’ button |  |
| 2 robot goes to tile 1 (first tile in a cleaning path) , and cleans |  |
| 3 robot goes to tile 2 (second in a cleaning path) , and cleans |  |
| … (until all tiles cleaned) |  |
| 4 robot returns to charging station |  |
| POST CONDITION | All tiles in house have been cleaned  Robot back in charging station  Battery is not empty |

Scenario UC1-2 (exception case)

|  |  |  |
| --- | --- | --- |
| PRECONDITION | Robot is ON  Battery is charged but not enough ..  Learn procedure completed  Dust bag is empty enough.. |  |
| 1 actor end user pushes ‘start’ button |  |  |
| 2 robot goes to tile 1 (first tile in a path) , and cleans |  |  |
| 3 robot goes to tile 2 (second in a path) , and cleans |  |  |
| 4 robot recognizes battery is low |  |  |
| 5 robot returns to charging station |  |  |
| POST CONDITION | Battery is nearly empty  Robot back in charging station  House is partially cleaned |  |

Scenario UC1-3 (exception case)

|  |  |  |
| --- | --- | --- |
| PRECONDITION | Robot is ON  Battery is charged  Learn procedure completed  Dust bag is NOT empty enough.. |  |
| 1 actor end user pushes ‘start’ button |  |  |
| 2 robot goes to tile 1 (first tile in a path) , and cleans |  |  |
| 3 robot goes to tile 2 (second in a path) , and cleans |  |  |
| 4 robot recognizes dust bag full |  |  |
| 5 robot returns to charging station |  |  |
| POST CONDITION | Battery is not empty  House is partially cleaned  Robot back in charging station |  |

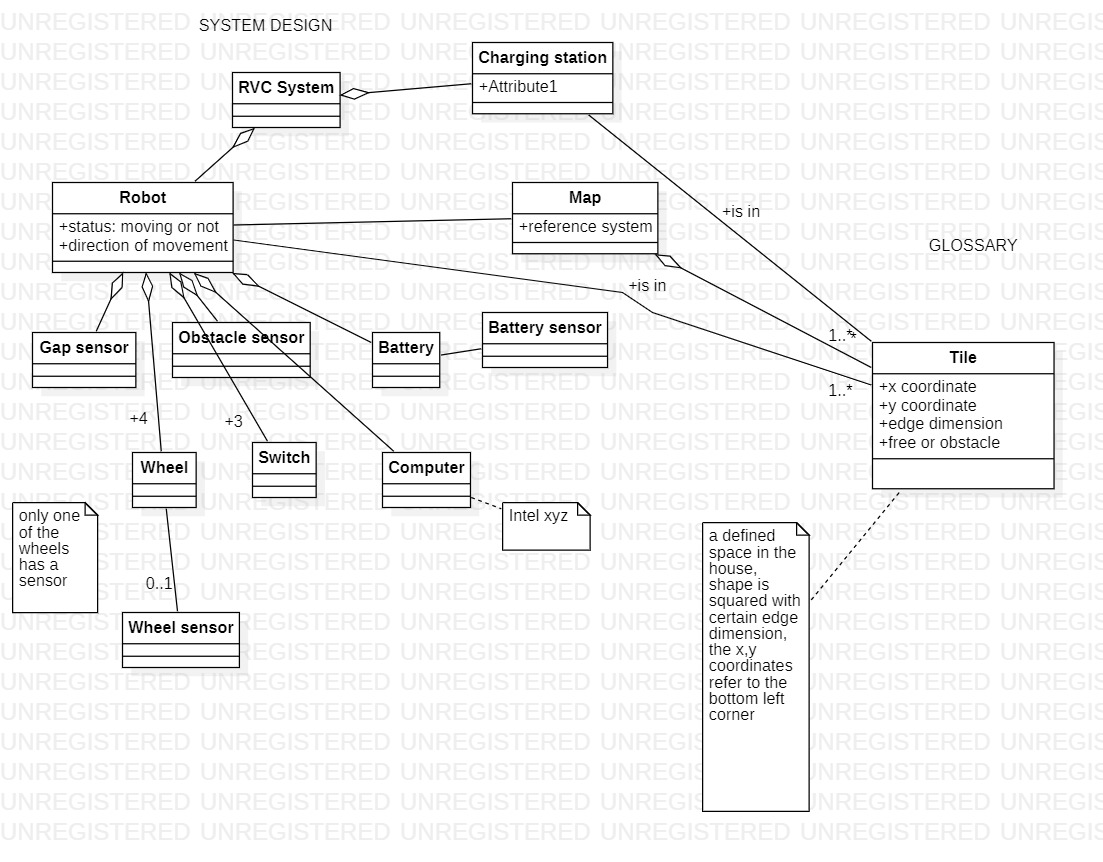
### UC2

Use case ‘Learn the house’ UC2

### UC3

…

## System design + Glossary



## Deployment diagram

