

# Projet Rover

## Introduction

Ce document a été rédigé par un équipe de développeurs du master développement logiciel à l'université Paul Sabatier.

L'équipe est composée de:

- Jordan Blanc-Poujol
- Guillaume Guinedor
- Ilja Kroonen
- Kevin Marchois

Notre équipe est composée de 4 membres avec l'aval de Mr. Bruel. En effet notre projet était déjà commencé lorsque les limites dans la taille du groupe ont été énoncées.

Dans ce document nous allons présenter le travail de modélisation réalisé sur le projet Rover, qui est un robot permettant l'exploration à distance.

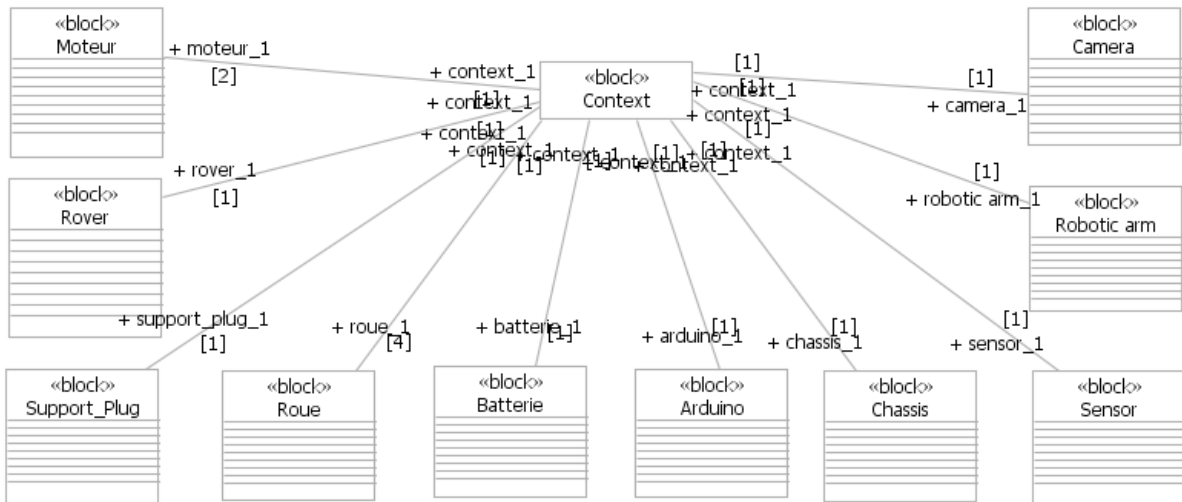
Voici les différents diagrammes qui seront présentés :

- Un diagramme de contexte (bdd)
- Un diagramme des exigences (req) du système
- Un diagramme des cas d'utilisation (uc) du système
- Un diagramme de blocks (bdd)
- Un diagramme interne de blocks (ibd)
- Un diagramme de packages
- Un diagramme de séquences (rencontre un mur)

## Sommaire

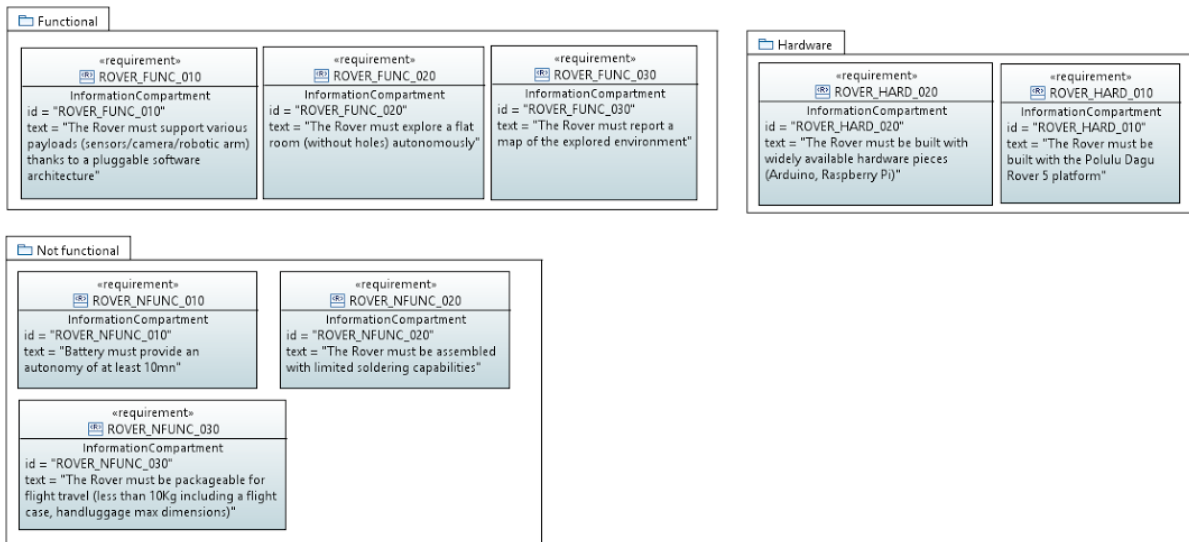
Projet Rover .....	1
Introduction.....	1
Sommaire .....	2
I) Diagramme de contexte .....	3
II) Diagramme des exigences.....	3
Requirements V1.....	3
Requirements V2.....	4
Requirements V3.....	4
Satisfy V1 .....	5
III) Diagramme des cas d'utilisation .....	6
IV) Un diagramme de blocs.....	6
V) Diagramme interne de blocs .....	7
VI) Diagramme de séquence.....	8
VII) Diagramme de séquence.....	9
VIII) Structure du modèle.....	9
Conclusion .....	15

## I) Diagramme de contexte

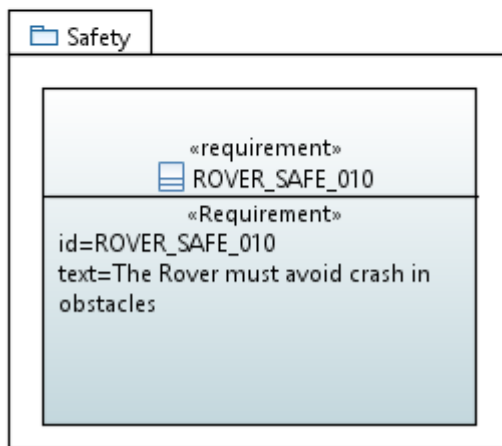
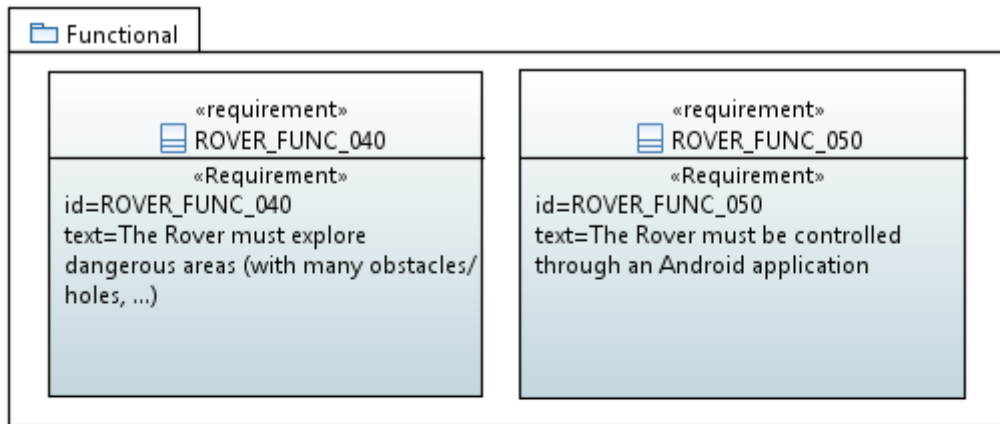


## II) Diagramme des exigences

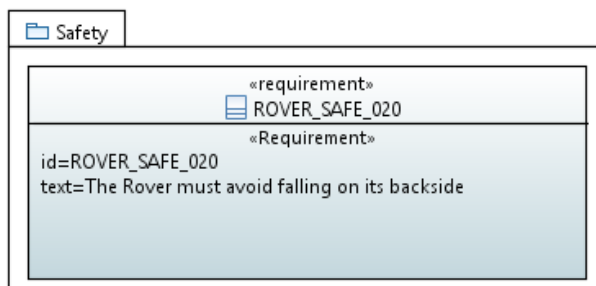
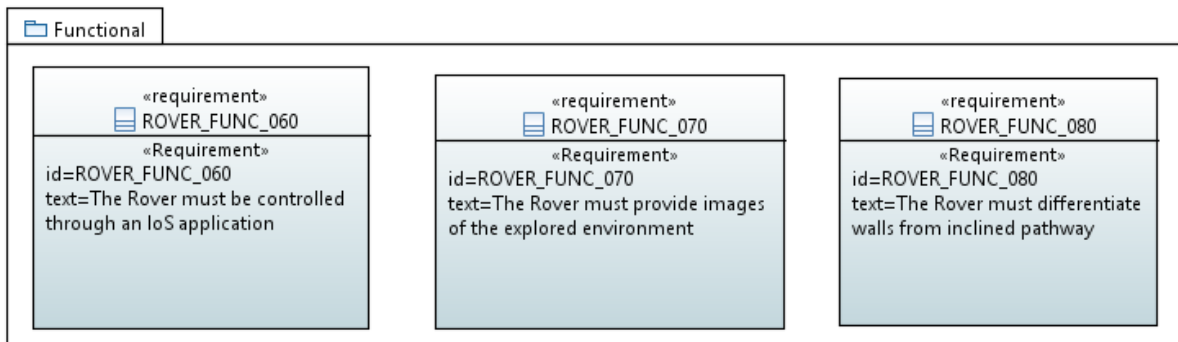
### Requirements V1



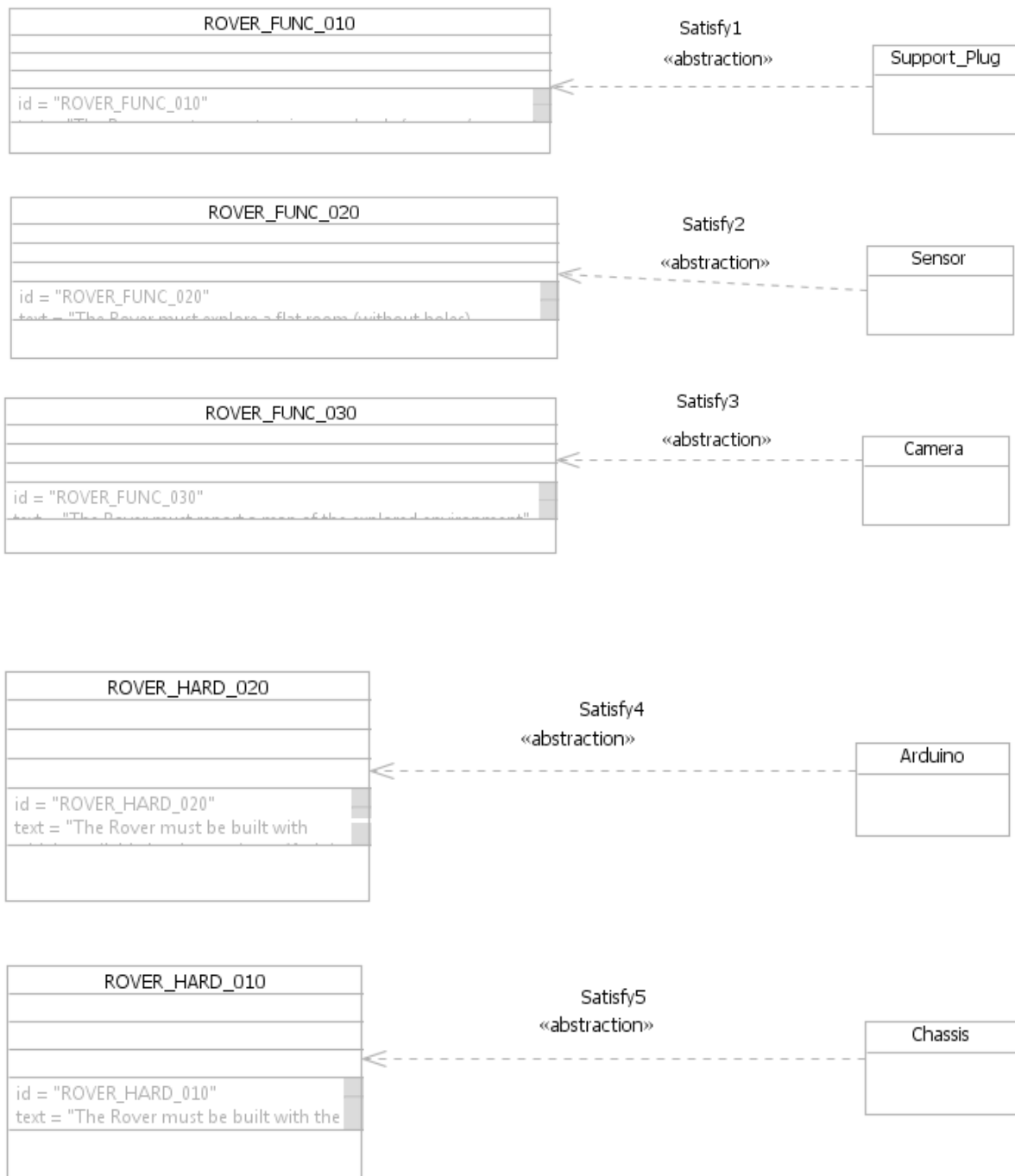
### Requirements V2

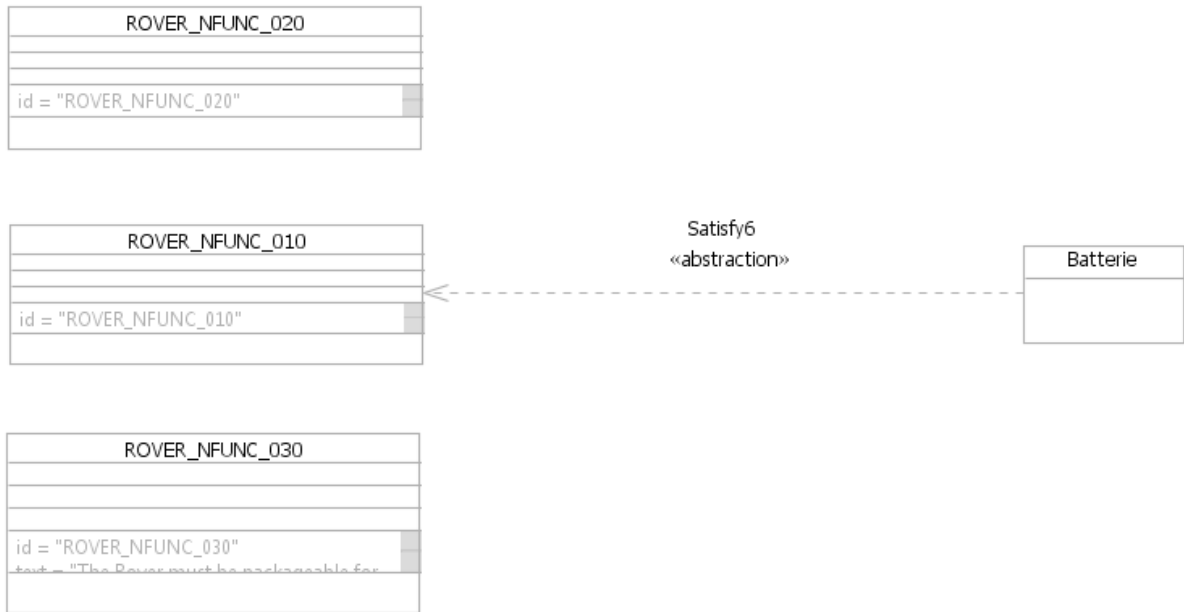


### Requirements V3

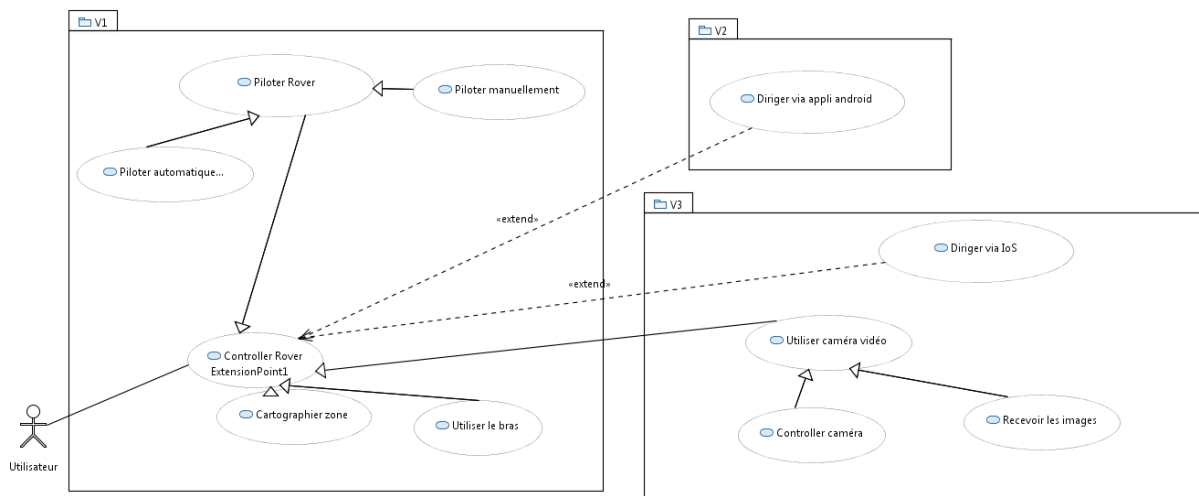


Satisfy V1

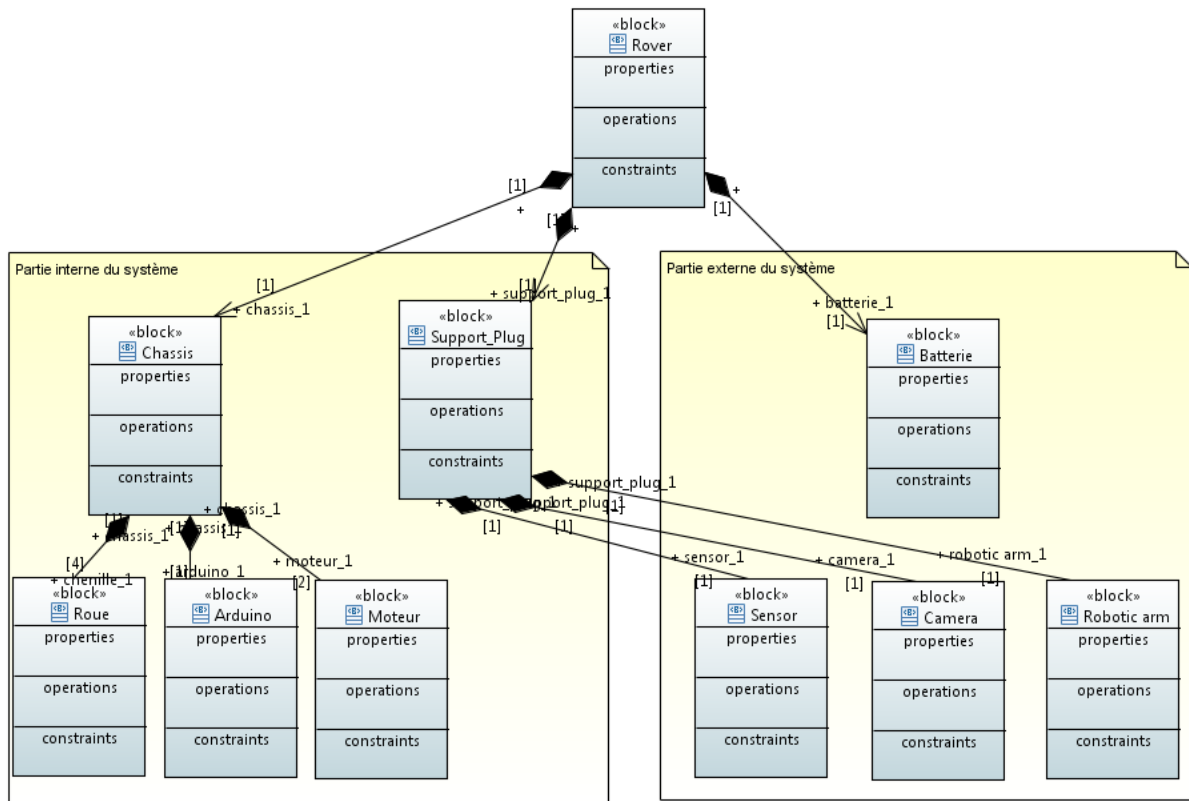




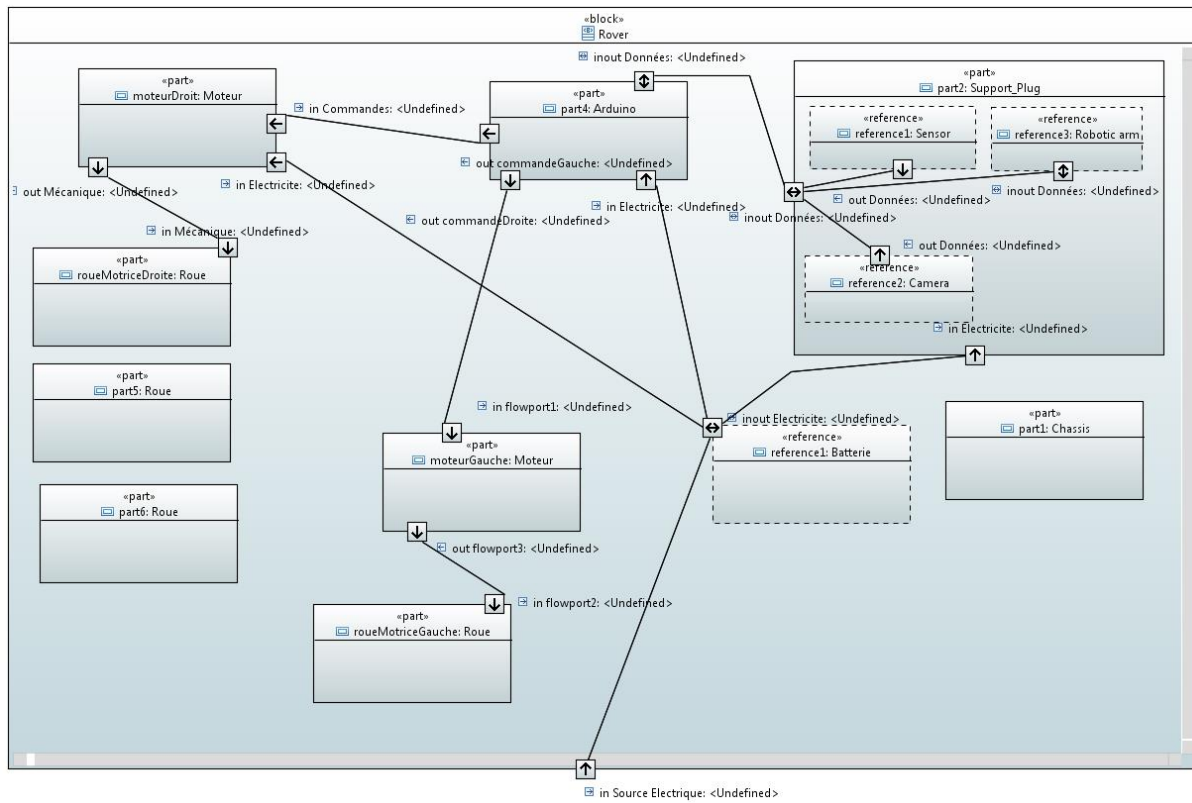
### III) Diagramme des cas d'utilisation



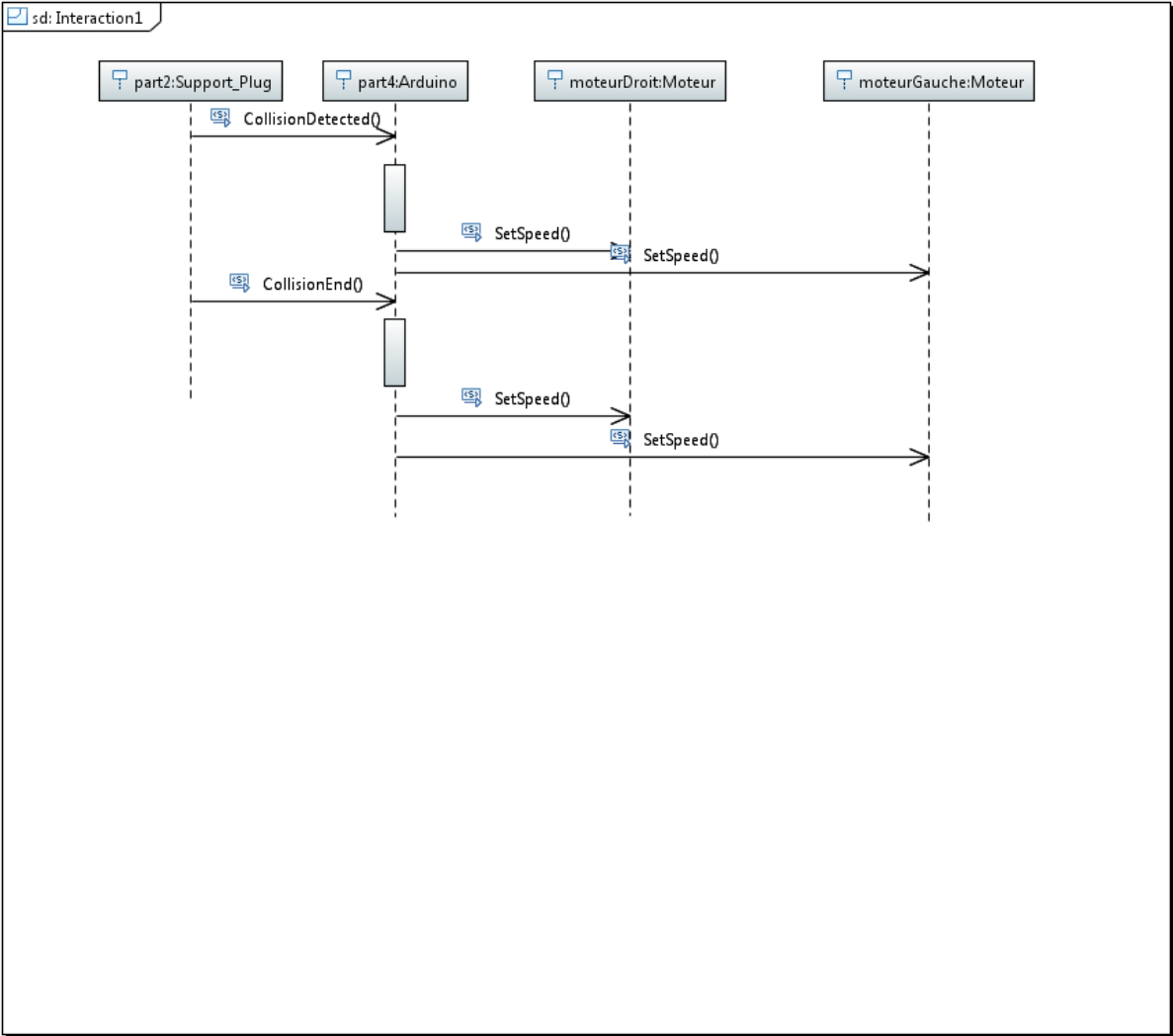
### IV) Un diagramme de blocs



### V) Diagramme interne de blocs

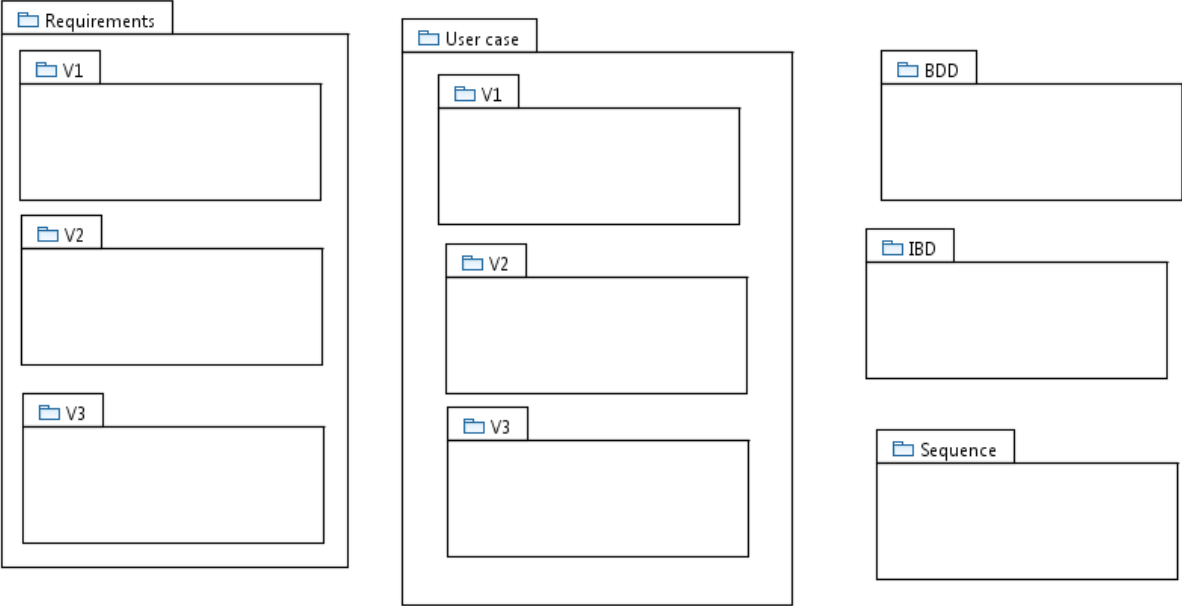


### VI) Diagramme de séquence

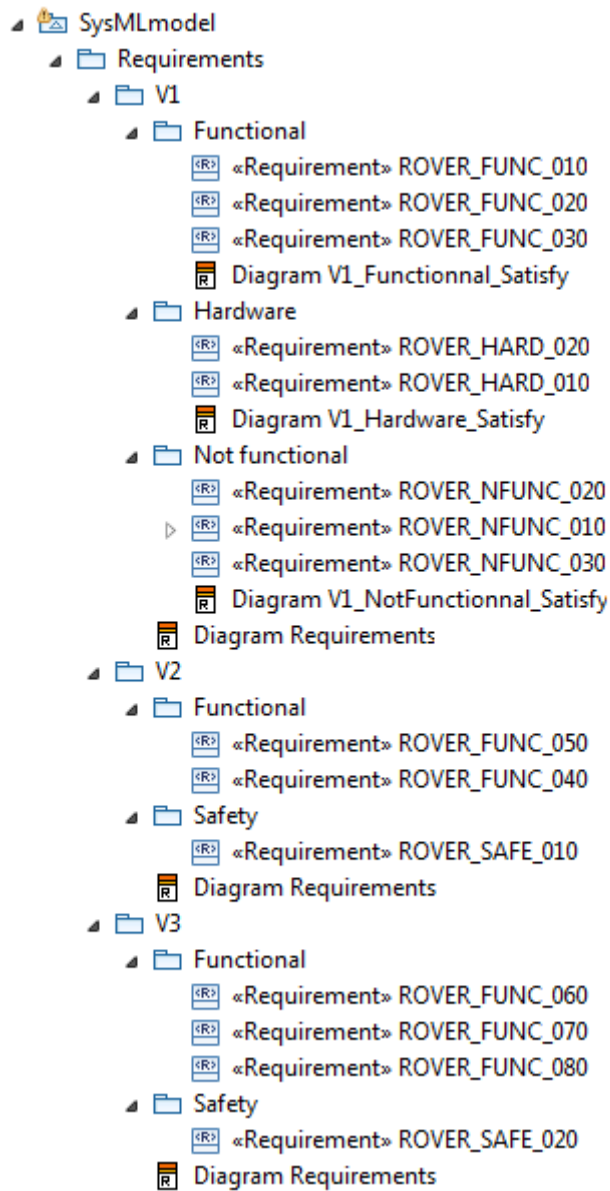


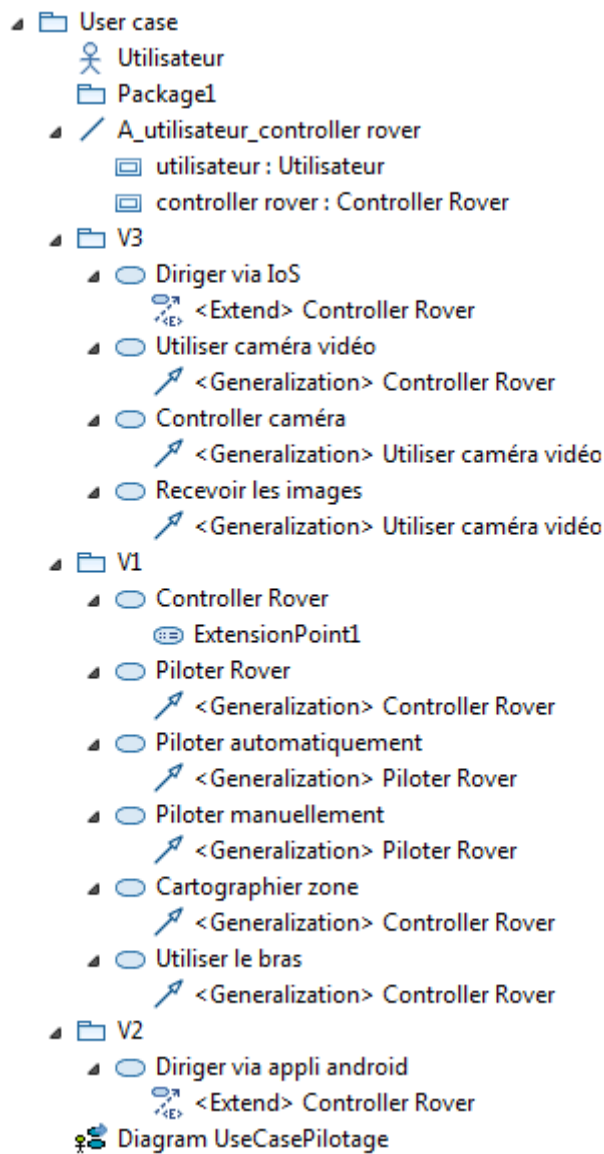


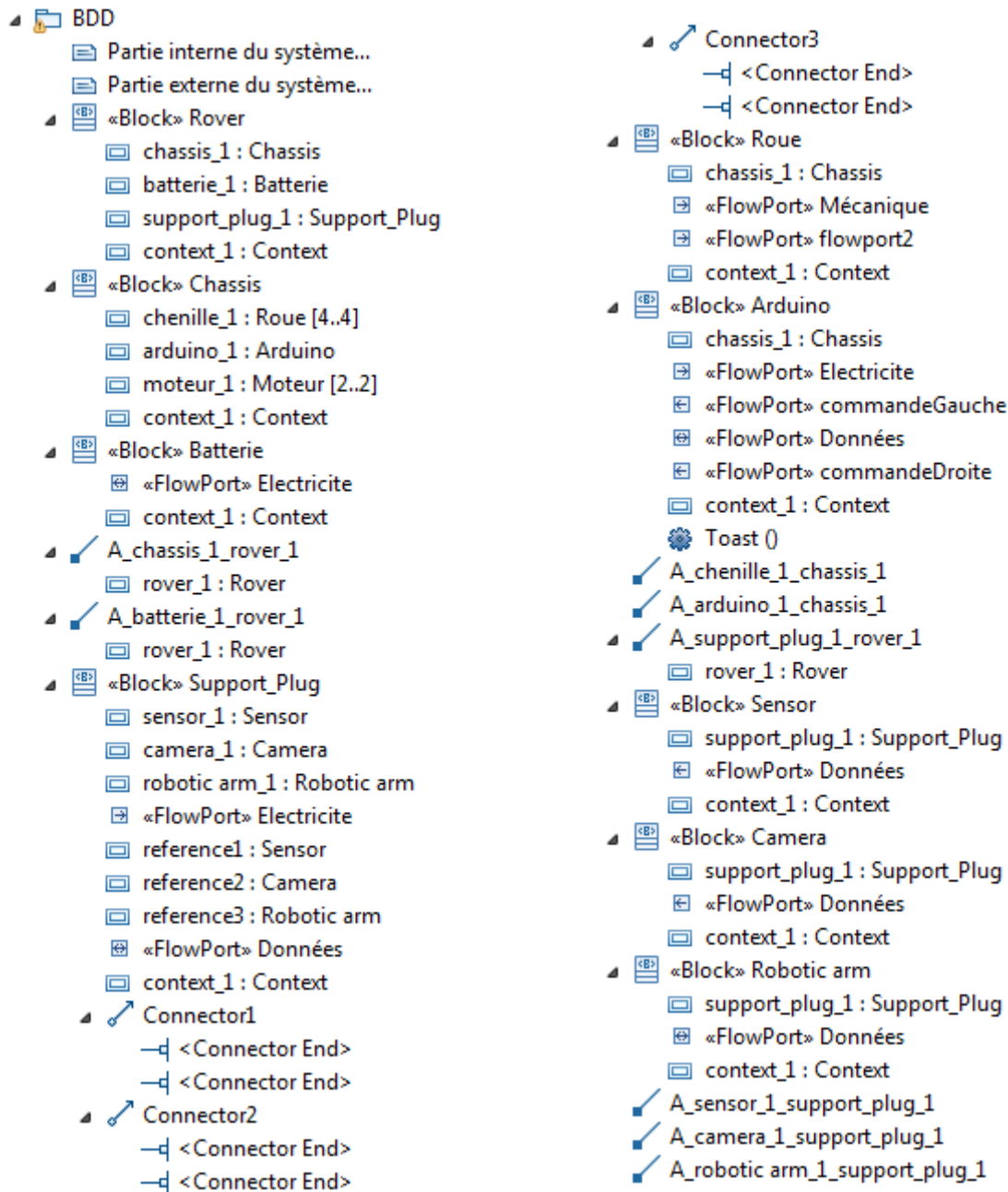
### VII) Diagramme de séquence





























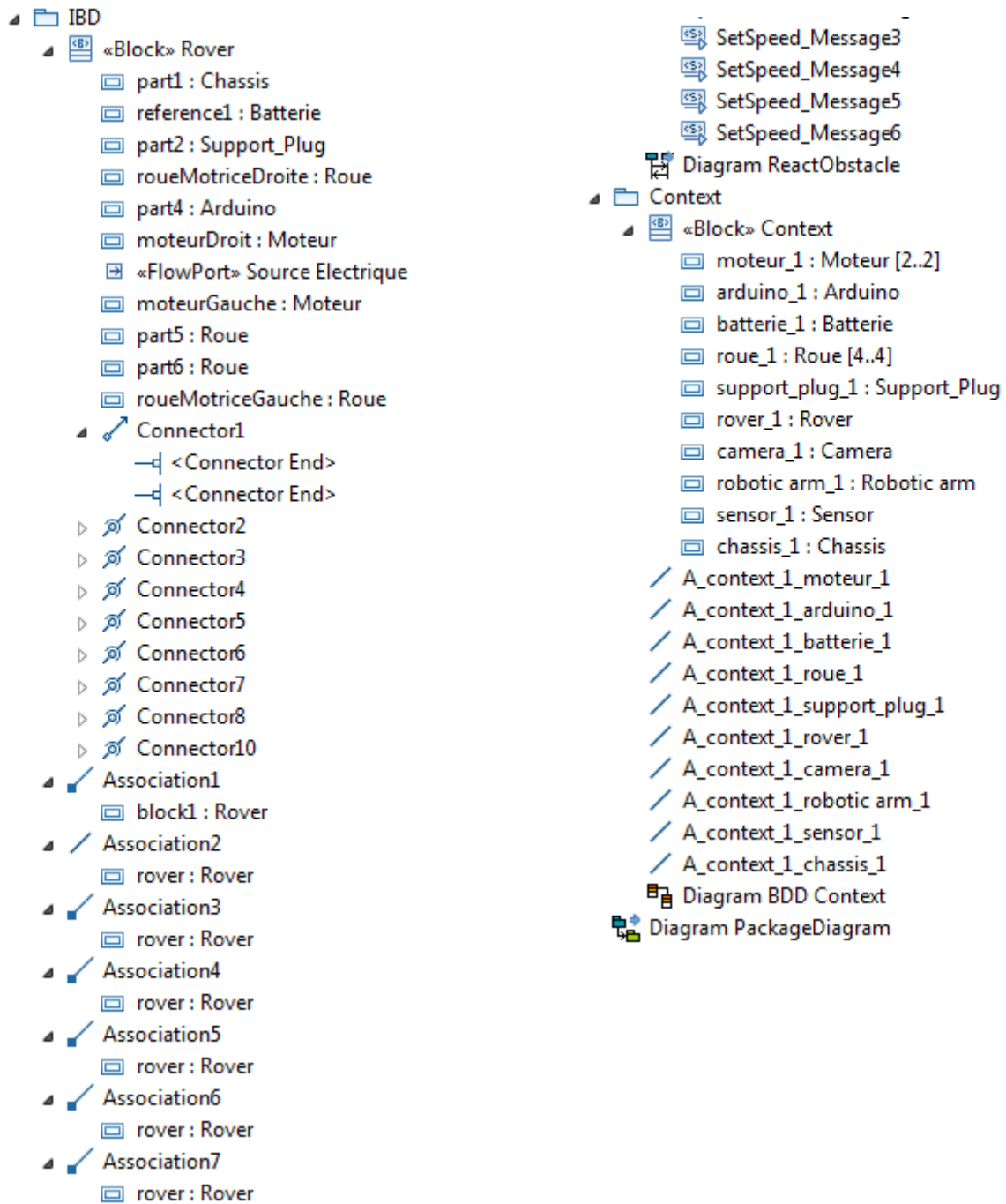
### VIII) Structure du modèle







- ▲  «Block» Moteur
  -  chassis\_1 : Chassis
    -  «FlowPort» Electricite
    -  «FlowPort» Mécanique
    -  «FlowPort» Commandes
    -  «FlowPort» flowport1
    -  «FlowPort» flowport3
    -  context\_1 : Context
  - ▲  A\_moteur\_1\_chassis\_1
  - ▲  Association1
    -  support\_plug : Support\_Plug
  - ▲  Association2
    -  support\_plug : Support\_Plug
  - ▲  Association3
    -  support\_plug : Support\_Plug
  -  CollisionDetected
  -  CollisionDetected
  -  SetSpeed
  -  CollisionEnd
  -  «Satisfy» <Abstraction> Satisfy1
  -  «Satisfy» <Abstraction> Satisfy2
  -  «Satisfy» <Abstraction> Satisfy3
  -  «Satisfy» <Abstraction> Satisfy4
  -  «Satisfy» <Abstraction> Satisfy5
  -  «Satisfy» <Abstraction> Satisfy6
  -  Diagram BDD\_Rover



## Conclusion

Ce petit projet nous a permis de découvrir SysML et a été une bonne mise en pratique de ces connaissances, ainsi que la découverte d'un nouvel outil Papyrus qui pourrait s'avérer utile dans nos stages.

En effet l'utilisation de SysML nous a permis d'intégrer un nouvel aspect de la programmation que nous n'avions pas pris en compte jusqu'à lors, l'aspect physique du projet. Nous avons appris à ne plus penser uniquement « logiciel », et à intégrer des éléments extérieurs au logiciel comme les capteurs, les roues.

L'utilisation de SysML nous a aussi donné la possibilité de créer des diagrammes dynamiquement. En effet par exemple la construction du BDD nous a permis de construire l'IBD en important directement les différents éléments du BDD.