



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
المدرسة العليا في الهندسة الكهربائية وطاقوية بهران
Ecole Supérieure en Génie Electrique et Energétique d'Oran

Support de cours du module

Ingénierie 1

Ce support est destiné aux étudiants de 2^{ème} année
des classes préparatoires en sciences et technologies

Réalisé par

HENDEL MOUNIA

MAITRE DE CONFERENCE A L'ESG2E

Année Universitaire 2017/2018

Avant- propos

Conforme au programme pédagogique des classes préparatoires en sciences et technologies, défini par arrêté ministériel, de l'enseignement supérieur et de la recherche scientifique, ce support de cours complété par un manuel de travaux pratiques s'adresse aux étudiants de deuxième année, module "Ingénierie 1". Ce support de cours permet d'avoir un aperçu générale sur l'étude des systèmes.

Partie 1: Destinée à permettre aux étudiants de connaitre les notions fondamentales des systèmes: composantes, interactions, finalités, environnements...etc.

Partie 2: Consacrée à la modalisation des systèmes en utilisant le langage SysML.

Partie 3: Planification de projets par représentations graphiques: GANTT et PERT.

Application 1: Modélisation structurelle, fonctionnelle et comportementale d'un système avec le langage SysML.

Application 2: Planification d'un projet avec le logiciel Excel (représentations GANTT et PERT).

Sommaire

Etudes des Systèmes.

1. Introduction.....	4
2. Présentation générale des systèmes.....	4
3. Classification des systèmes.....	5
3.1. Caractéristiques Technico-économiques.....	5
3.2. Caractéristiques Physico-économiques.....	6
4. Cycle de vie d'un logiciel.....	6
4.1 Analyse du besoin.....	7
4.2 Cahier de charge.....	7
4.3 Conception du système.....	7
4.4 Réalisation des éléments.....	7
4.5 Intégration du système.....	8
4.6 Livraison / Distribution.....	8
4.7 Modification, maintenance et adaptation.....	8
5. Le langage SysML.....	8
5.1 Diagramme d'exigences.....	9
5.2 Diagrammes Structurels.	12
5.3 Diagrammes comportementaux.....	18
6. Graphe pour projet, GANTT et PERT.....	23
6.1 Matrice des antériorités.....	24
6.2 Graphe sagittal potentiel tâches.	27
6.3 PERT potentiel tâches.....	27
6.4 Diagramme de Gantt.....	29

Application 1. Introduction au Systems Modeling Language (SysML): étude des systèmes.

Exercices et solutions.....	30
-----------------------------	----

Application 2. Planification graphique d'un projet sous Excel.

Exercices et solutions.....	34
-----------------------------	----

Bibliographie.....	46
--------------------	----

Chapitre : Etudes des Systèmes.

1. Introduction:

L'ingénierie des systèmes (SI) constitue une méthodologie bien déterminée. Elle permet l'étude, la conception, l'évaluation et l'amélioration d'un système donné dans le but d'associer une solution satisfaisante (pour toutes les parties prenantes) à un ou plusieurs besoins.

2. Présentation générale des systèmes:

Un système est un assemblage de constituants, interagissant entre eux (échange d'informations, échange de matières, échange d'énergie...etc.) de manière à répondre à un ou plusieurs besoins.

On peut distinguer deux types de systèmes:

- Les systèmes naturels, mis en place par la nature. (Exemples: corps humains, système planétaires...etc.)
- Les systèmes industriels appelés aussi systèmes artificiels, proposés par l'homme. (Exemples: voiture, tablette, vélo...etc.)

Remarques:

- Il ne suffit pas de connaître tous les composants d'un système pour maîtriser et comprendre son fonctionnement.

Par exemple il ne suffit pas de connaître les composants d'une voiture pour connaître le mode de son fonctionnement. (voir figure)

- Un système donné n'est jamais isolé du monde extérieur (ils interagissent en permanence entre eux). Ainsi, la définition d'un système nécessite la définition des interactions entre les constituants, mais aussi la définition des interactions avec les éléments du milieu extérieur (EME).

Par exemple une voiture est en relation avec le conducteur, la route et l'air, ces derniers représentent des éléments externes à la voiture. (voir figure 2)

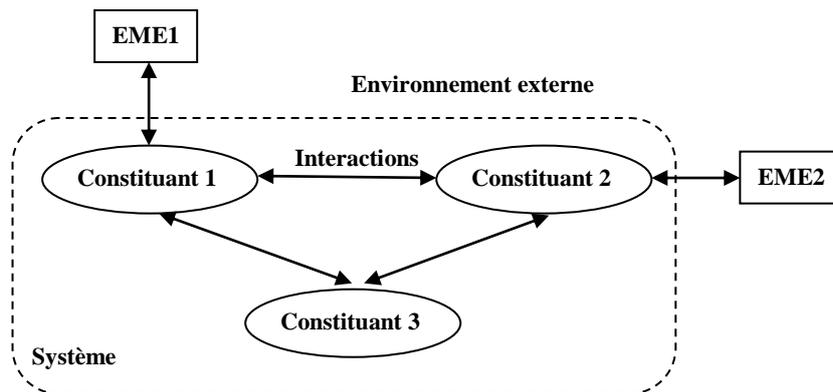


Figure 1: Constituant, Interactions, EMEs et environnement externe.

3. Classification des systèmes:

Les systèmes sont généralement classés selon les caractéristiques Technico-économiques et le domaine Physico-économique:

3.1 Caractéristiques Technico-économiques:

L'ensemble de ces caractéristiques sont étudiées au cours de la mise au point du système:

- La durée de vie.
- La fiabilité.
- Le cout.
- La quantité.

Exemple: Les rasoirs jetables ont un cout faible mais une durée de vie très limitée en comparaison aux rasoirs électriques.

3.2 Caractéristiques Physico-économiques:

Représentent le domaine d'application pour lequel le système a été conçu, dans ce contexte on peut distinguer deux catégories de systèmes:

- Les systèmes à production limitée: on retrouve ce genre de systèmes généralement dans les usines, on peut citer comme exemple: les grandes machines de production.
- Les systèmes à grande production: représentent des produits à grande diffusion, on peut citer comme exemples: les pièces de rechange des voitures, l'électroménager...etc.

4. Cycle de vie d'un logiciel:

Il est exprimé en 9 différentes étapes qui vont de l'analyse du besoin à l'élimination du système. (Voir figure 2)

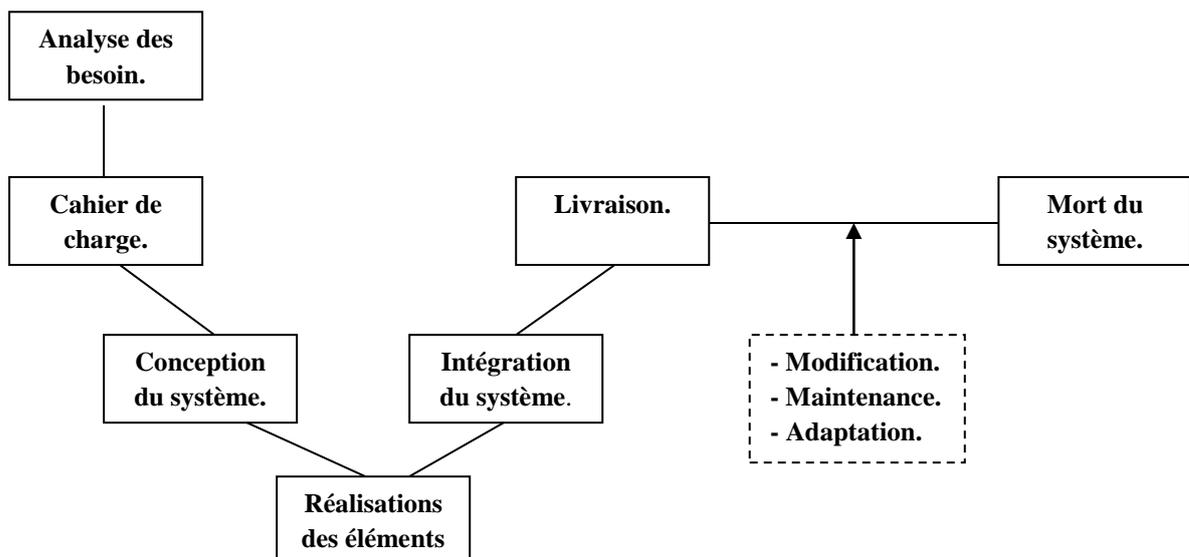


Figure 2: Cycle de vie d'un système.

4.1 Analyse du besoin:

Définition d'un besoin: « Un besoin est une nécessité ou un désir éprouvé par un utilisateur.»

Pour valider le besoin, il faut se poser les trois questions:

- Pourquoi le produit existe-t-il?
- Qu'est ce qui pourrait faire évoluer le besoin ?
- Qu'est ce qui pourrait le faire disparaître?

Exemple: La cabine téléphonique, si une cabine téléphonique permet à l'utilisateur d'avoir accès au réseau téléphonique, le besoin d'une cabine téléphonique est notablement réduit par l'apparition des téléphones portables.

4.2 Cahier de charge:

C'est le contrat entre l'entreprise et le client, il formalise avec précision les besoins du client. Il spécifie un tableau de bord qui représente le système (projet) et énumère avec précision les conditions dans lesquelles il doit être réalisé.

Les contraintes à prendre en considération sont généralement de nature: économiques, environnementales, humaines, industrielles et matérielles...etc.

4.3 Conception du système:

C'est l'étape qui consiste à définir avec précision comment doit se présenter le système. A l'issue de cette phase on détermine l'ensemble des composants ainsi que la manière dont ils seront reliés.

4.4 Réalisation des éléments:

Cette phase permet de réunir l'ensemble des constituants matériels (pièces mécaniques, moteurs...etc) et immatériels (programmes informatiques, logiciels...etc) en vue d'assemblage.

4.5 Intégration du système:

C'est la phase de montage ou la mise au point du système (assemblage de composants matériels et immatériels).

4.6 Livraison / Distribution :

Cette phase de vie peut apporter des contraintes supplémentaires au système.

4.7 Modification, maintenance et adaptation :

Cette phase constitue la mise en service du système depuis sa livraison jusqu'à la mise au rebut finale. Cette phase peut être entrecoupée de phases de modification, adaptation et amélioration et maintenance du système.

5. Le langage SysML:

La modélisation d'un système inclut, la simulation de son comportement et la communication des descriptions. On SI cette modalisation s'effectue selon trois points de vu: fonctionnel, structurel et comportemental [].

- Point de vu "Fonctionnel" (A quoi sert le système?), permet de fixer les actions effectuées par le système.
- Point de vu "Structurel" (De quoi est il composé ? et Comment est il organisé ?): consiste à définir les composants et les éléments du milieu extérieur, ainsi que les interactions entre eux.
- Point de vu "Comportemental" (quelles sont ces performances?): consiste a modeliser théoriquement le produit et son environnement, puis simuler son comportement.

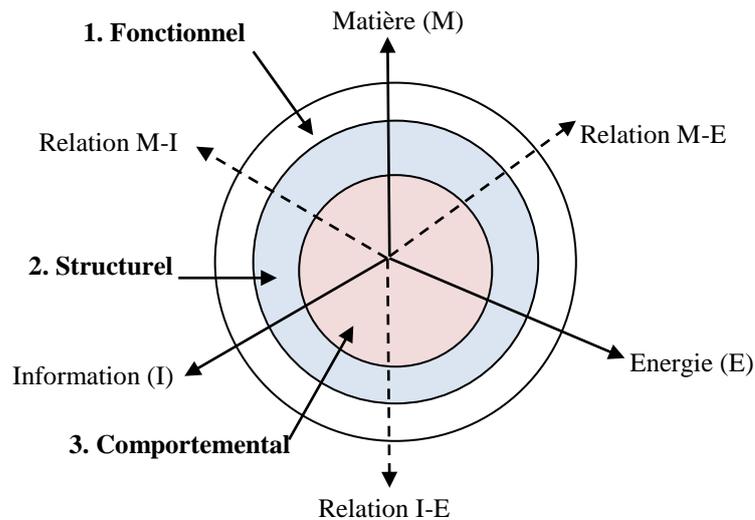


Figure 3: Les points de vu fonctionnel, structurel et comportemental.

Le langage SysML permet de représenter ces trois points de vu au tour de neuf diagrammes et au sein d'un unique model à "Multipoints de vue".

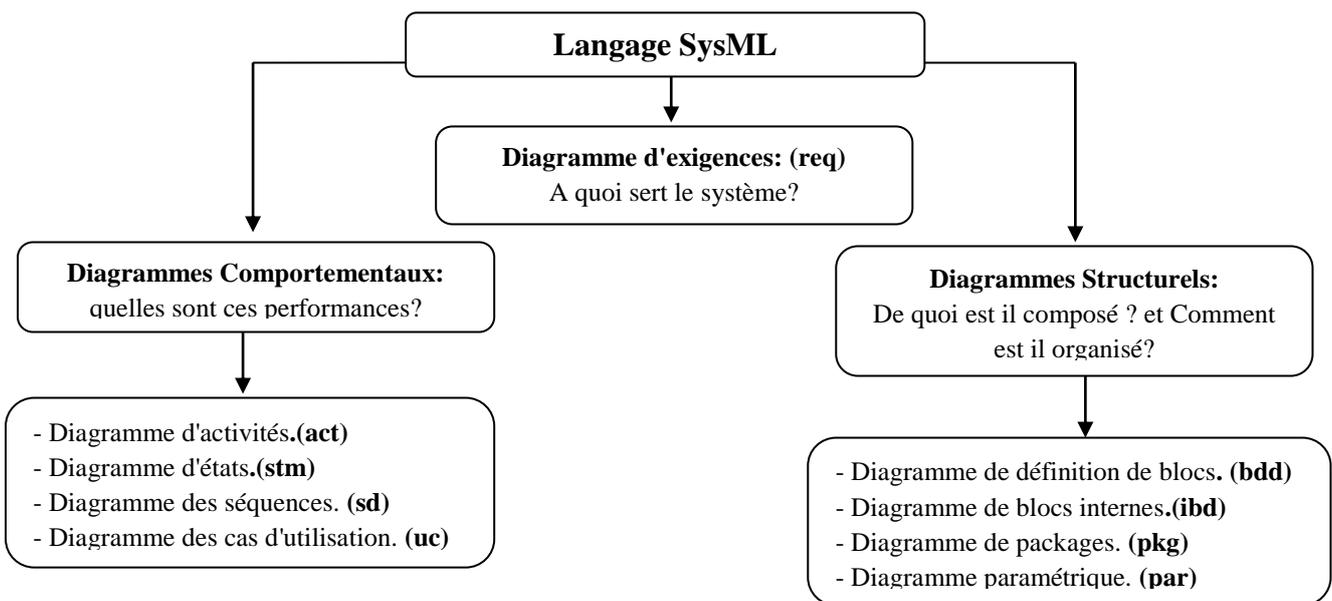


Figure 4: Présentation générale des diagrammes SysML .

5.1 Diagramme d'exigences "Requirement Diagram": permet d'exprimer les fonctions et les performances du système (élaborer le cahier de charges).

- Exigences Techniques.
- Exigences Fonctionnelles
- Exigences Economiques...etc.

a) Représentation:

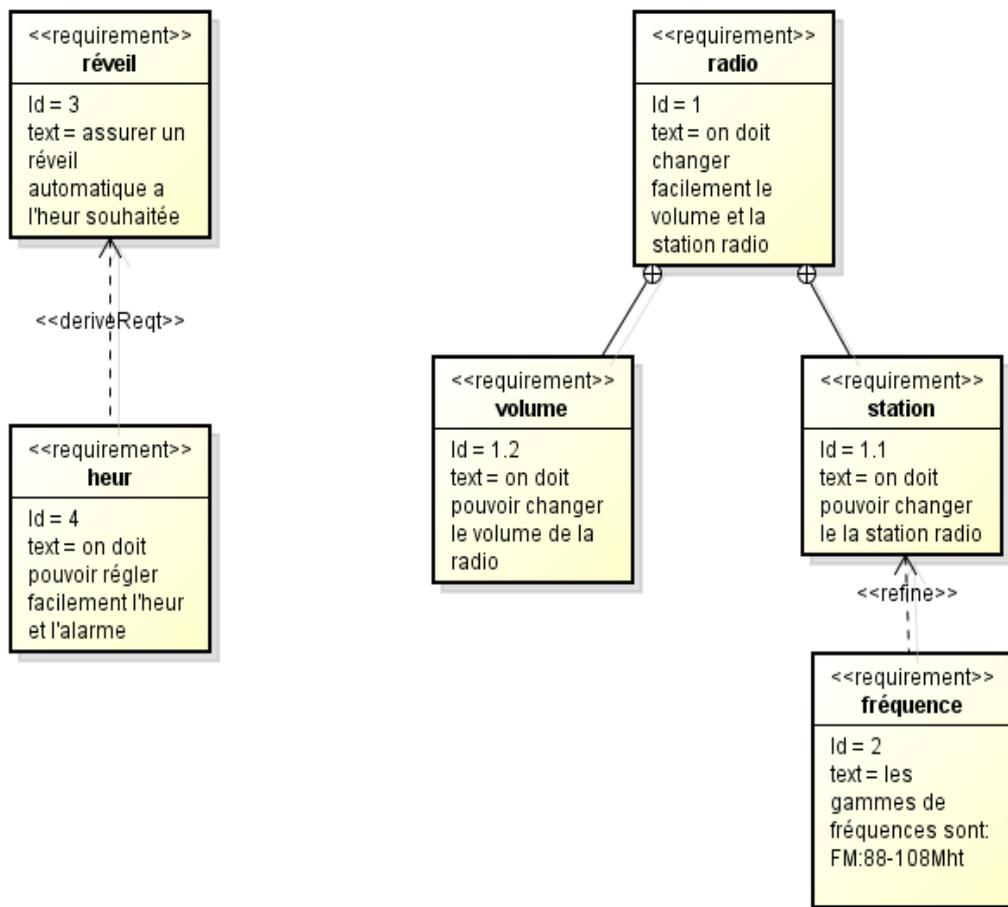
- Chaque exigence est décrite par un texte à l'intérieur du rectangle.
- Un rectangle est marqué par un stéréotype "requirement" identifié d'une façon unique.
- Les exigences sont reliées entre elles par des relations:

⊗ : Contenance: permet de décomposer une exigence composite en exigences unitaires.

⋯→ : Raffinement: ajout de précision (en général numérique).

⋯→ : Relie les exigences qui traitent la même problématique et de niveau différent.

Exemple:



Exercice 1:

Présentation: HALO est une balance d'entrée de gamme au design innovant dont les caractéristiques sont:

- Le prix est compris entre 15 et 30 Euros.
- Une gamme de 5 couleurs tendance.
- Permet de supporter une charge maximale de 3kg, avec une précision de 1g.
- Permet l'ajout d'ingrédients supplémentés au contenu du récipient.

Travail à effectuer:

- 1) Déterminer les exigences suivantes: Halo, prix, couleurs, charge, précision, Ajout.
- 3) Tracer le diagramme d'exigences ainsi que les relations en cas de nécessité.

Solution:

- 1) Déterminer les exigences suivantes:

Halo: balance doit être d'entrée de gamme au design innovant.

Prix: Le prix doit être compris entre 15 et 30 euros.

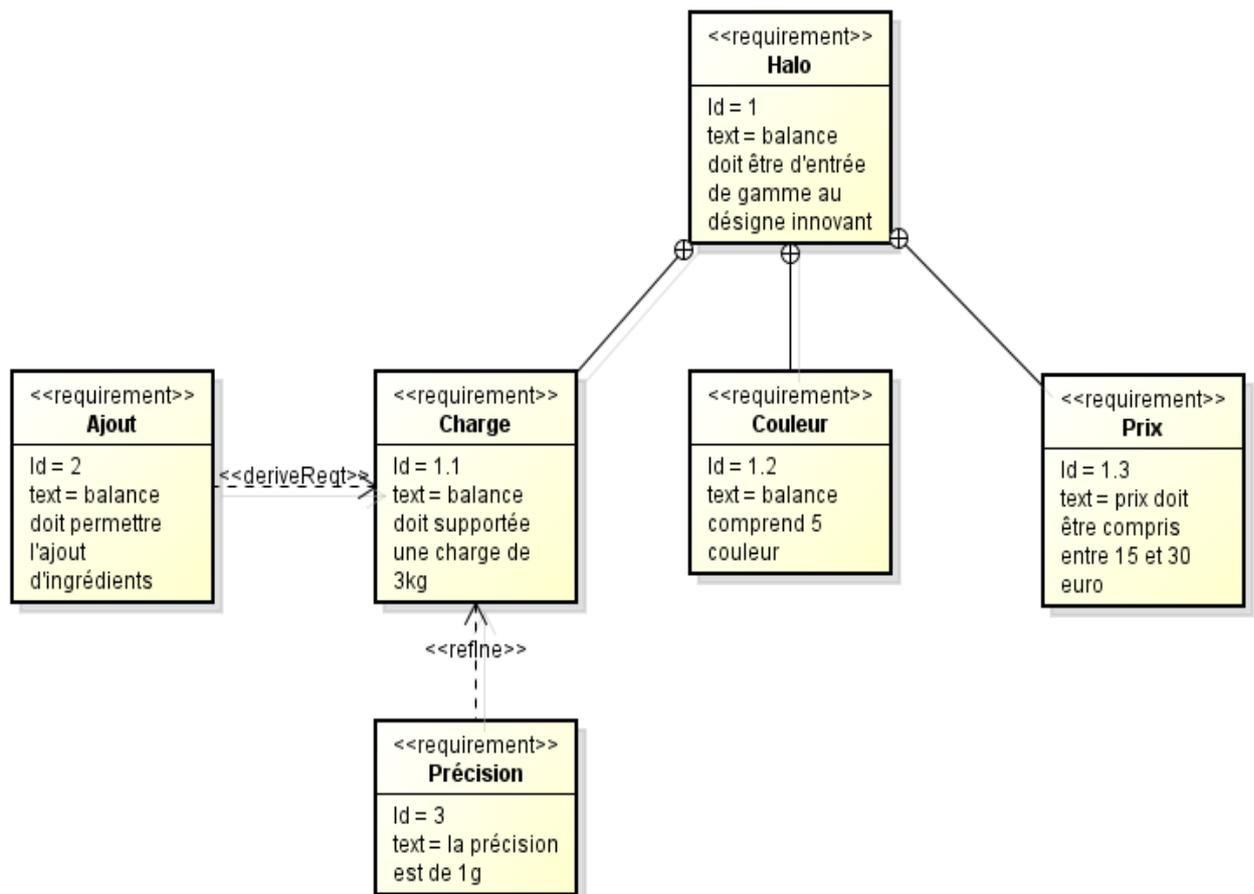
Couleur: La balance comprend 5 couleurs.

Charge: la balance doit supporter une charge de 3kg.

Précision: la précision est de 1g.

Ajout: la balance doit permettre l'ajout d'ingrédients.

- 3) Tracer le diagramme d'exigences ainsi que les relations en cas de nécessité.



Remarque: le diagramme des exigences permet aussi de ramener des éléments d'autres diagrammes et vérifier qu'une exigence a bien été prise en compte.

5.2 Diagrammes Structurels:

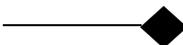
a) Diagramme de définition de blocs:

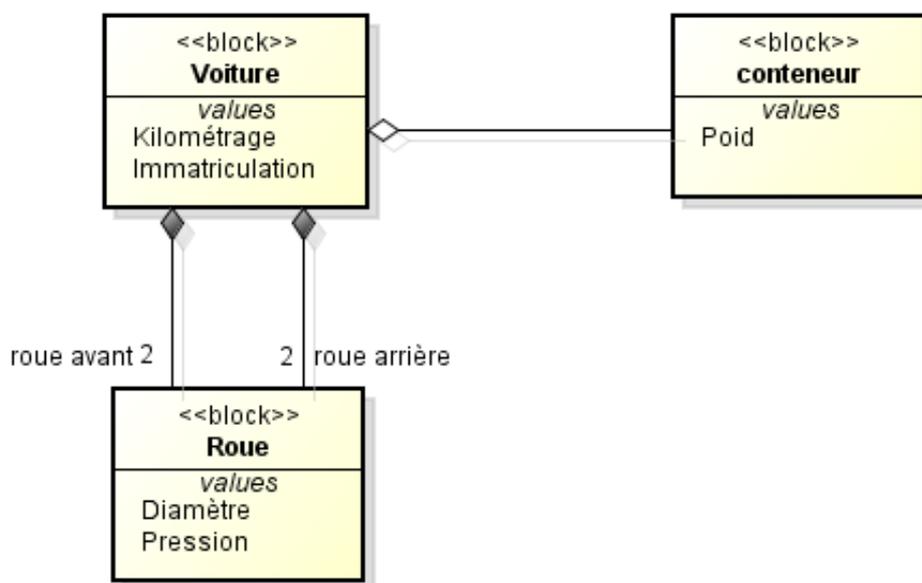
Rôle: - prendre connaissance de la structure interne du système (qui contient quoi?)
- peut représenter un composant matériel ou logiciel.

Représentation: Un bloc est représenté par un rectangle compartimenté:

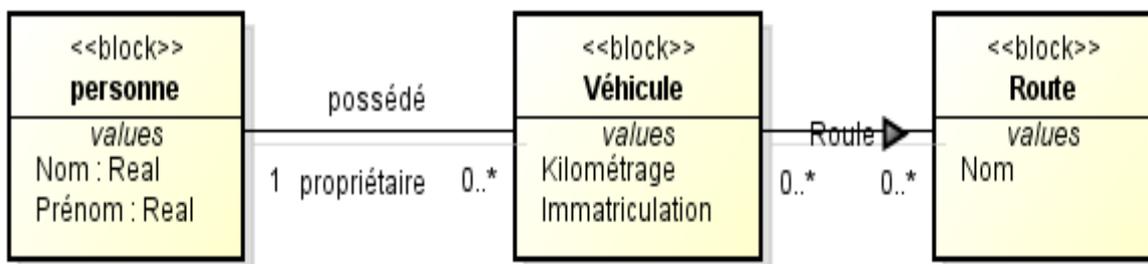
- Entête: constituer d'un stéréotype <<block>> et du nom du bloc.
- Labels: représente des valeurs, parties..etc.

Relations:

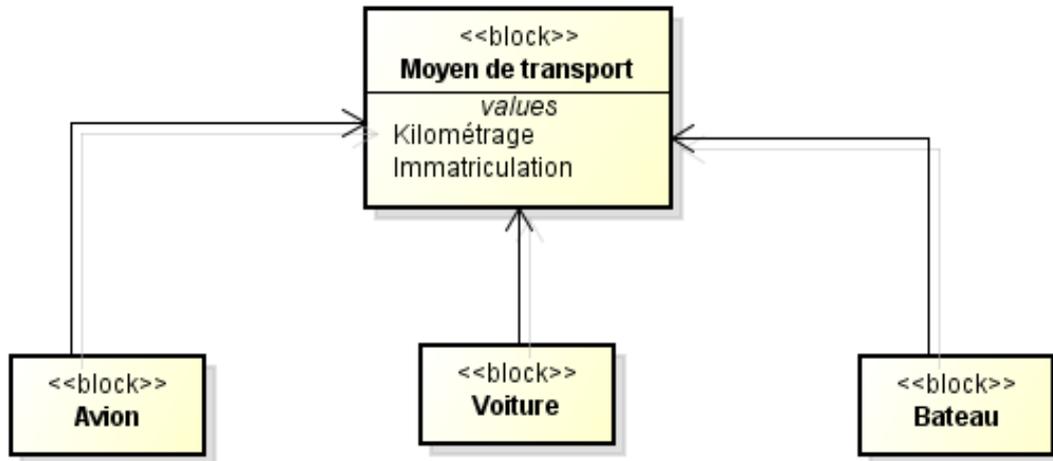
-  : Composition: Bloc conteneur a nécessairement besoin du bloc contenu pour assurer la fonction.
-  : Agrégation: Bloc conteneur n'a pas forcément besoin du bloc contenu.



-  : Association: n'implique pas la contenance, la nature du lien dépend du contexte (On peut ajouter une flèche dans le cas où la relation est unidirectionnelle).



-  : Généralisation: Factoriser des propriétés communes à plusieurs blocs dans un bloc généralisé.



b) Diagramme de blocs internes:

Rôle et représentation: Il décrit la structure interne du bloc, c'est à dire ses composants internes et les échanges entre eux. Les échanges sont assurés à l'aide:

- Port de flux:  il autorise l'entrée ou la sortie du flux (matière, énergie, ou information)
- Port standard:  il autorise l'échange d'un service, un résultat d'une opération, ou une consigne.
- Le connecteur:  ce lien relie 2 ports, dont la nature peut être précisée.

Exercice 2: La bouilloire électrique est constituée principalement:

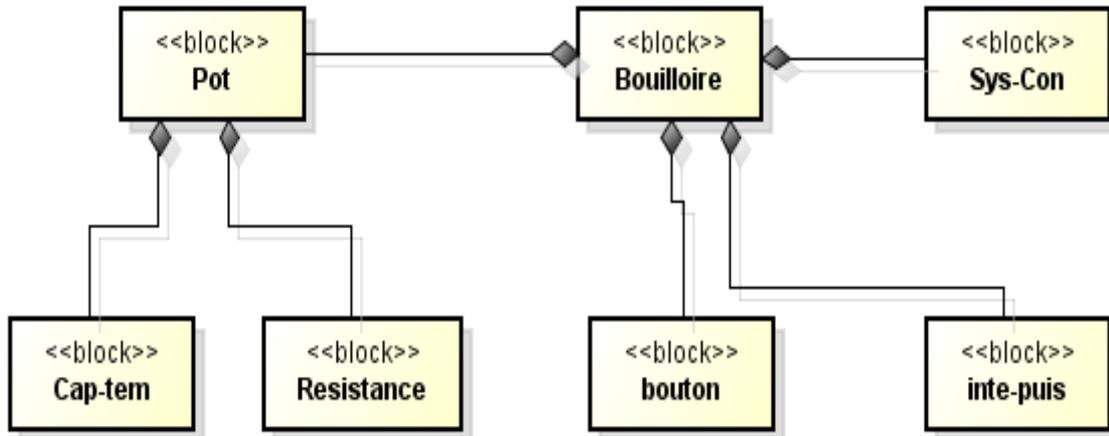
- bouton.
- système de contrôle.
- interface de puissance.
- Pot: capteur de température et résistance.

1) Tracer le diagramme de définition des blocs.

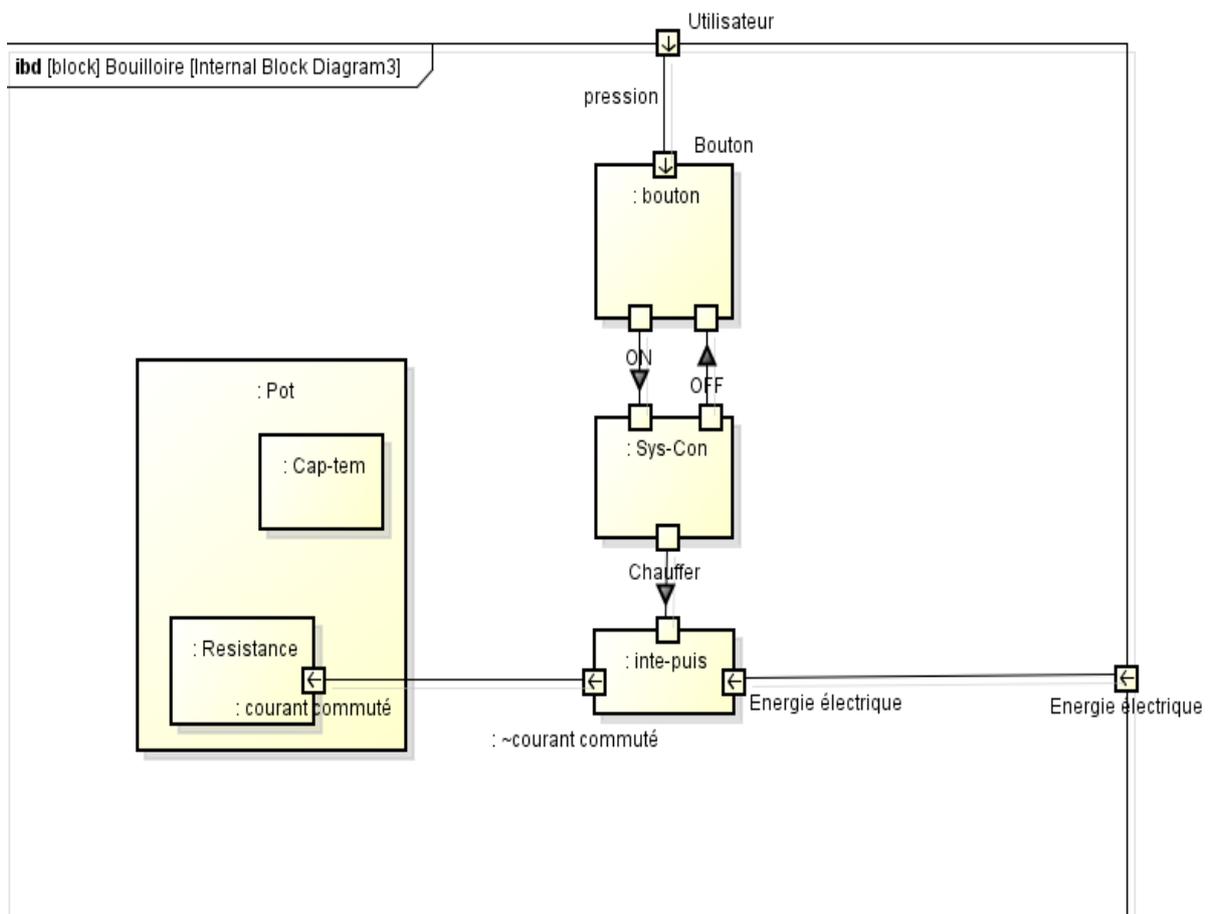
2) Tracer le diagramme de bloc interne.

Solution:

1) Le diagramme de définition des blocs:



2) Le diagramme de blocs internes:



c) Diagramme Paramétrique:

Rôle: Représente un ensemble d'équations modélisant tout ou partie du système.

Représentation: A partir d'un diagramme de définition de bloc, on réalise un assemblage d'équations pour établir un modèle mathématique modélisant tout ou partie du système.

- Les équations sont représentées par des blocs avec le stéréotype <<Constraint>>
- Une contrainte est définie par des paramètres ainsi qu'une règle.

Exercice 3:

Soit les équations relatives au dynamique du véhicule.

$$F = F_d + F_g \quad (F_d, F_g, F: N)$$

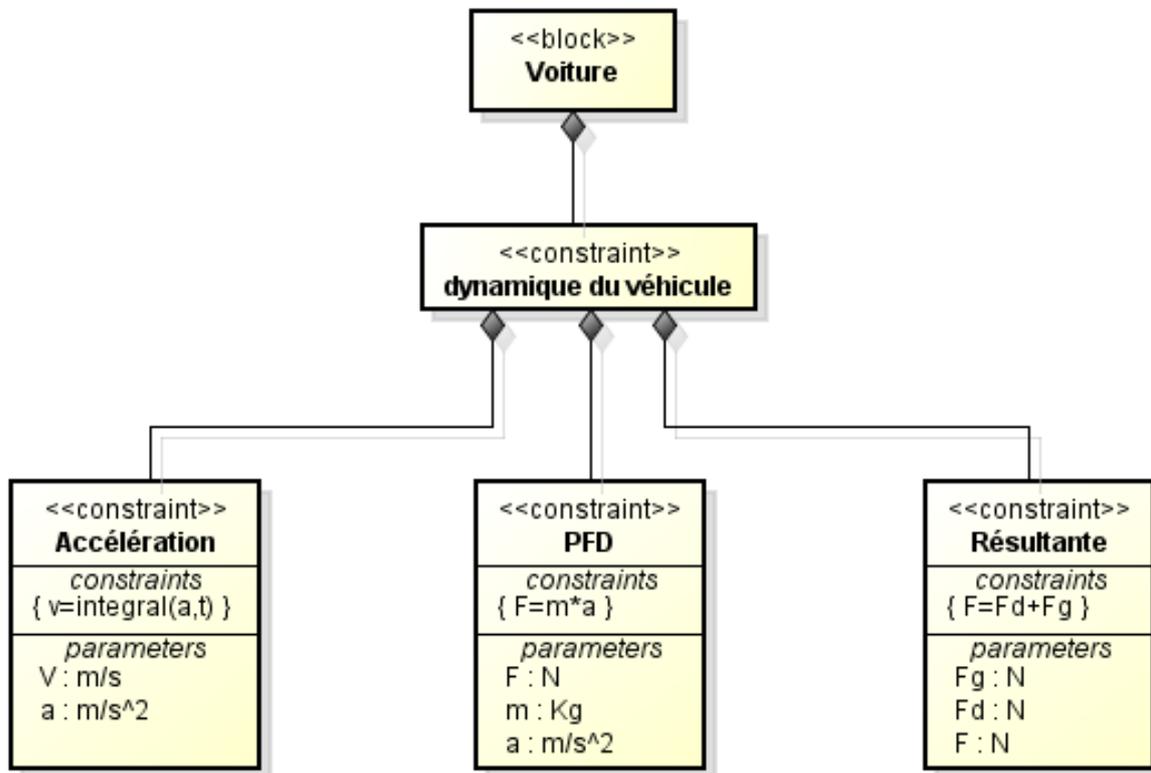
$$F = m \cdot a \quad (m: Kg, a: m/s^2)$$

$$V = \int(a, t) \quad (V: m/s)$$

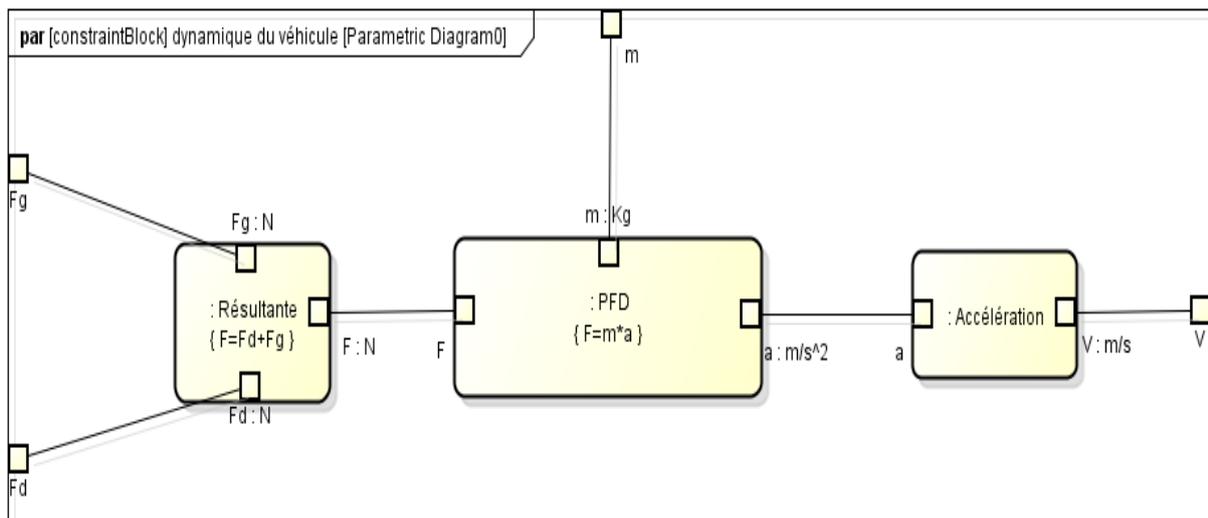
- 1) Tracer le diagramme de définition des blocs.
- 2) Tracer le diagramme Paramétrique.

Solution:

1) Le diagramme de définition des blocs:



2) Le diagramme Paramétrique:



d) Le diagramme de package:

En SysML, un élément du modèle est contenu dans un seul conteneur. Si ce conteneur est détruit ou dupliqué, les éléments qu'il contient sont également détruits ou dupliqués.

Représentation :

- Chaque package est représenté par un dossier.
- 2 relations entre packages : la contenance et la dépendance

5.3 Diagrammes comportementaux:

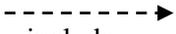
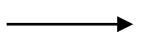
a) Diagramme de cas d'utilisation:

Rôle: Il exprime les services (use cases) offert par le système (sans spécifier comment) aux (acteurs).

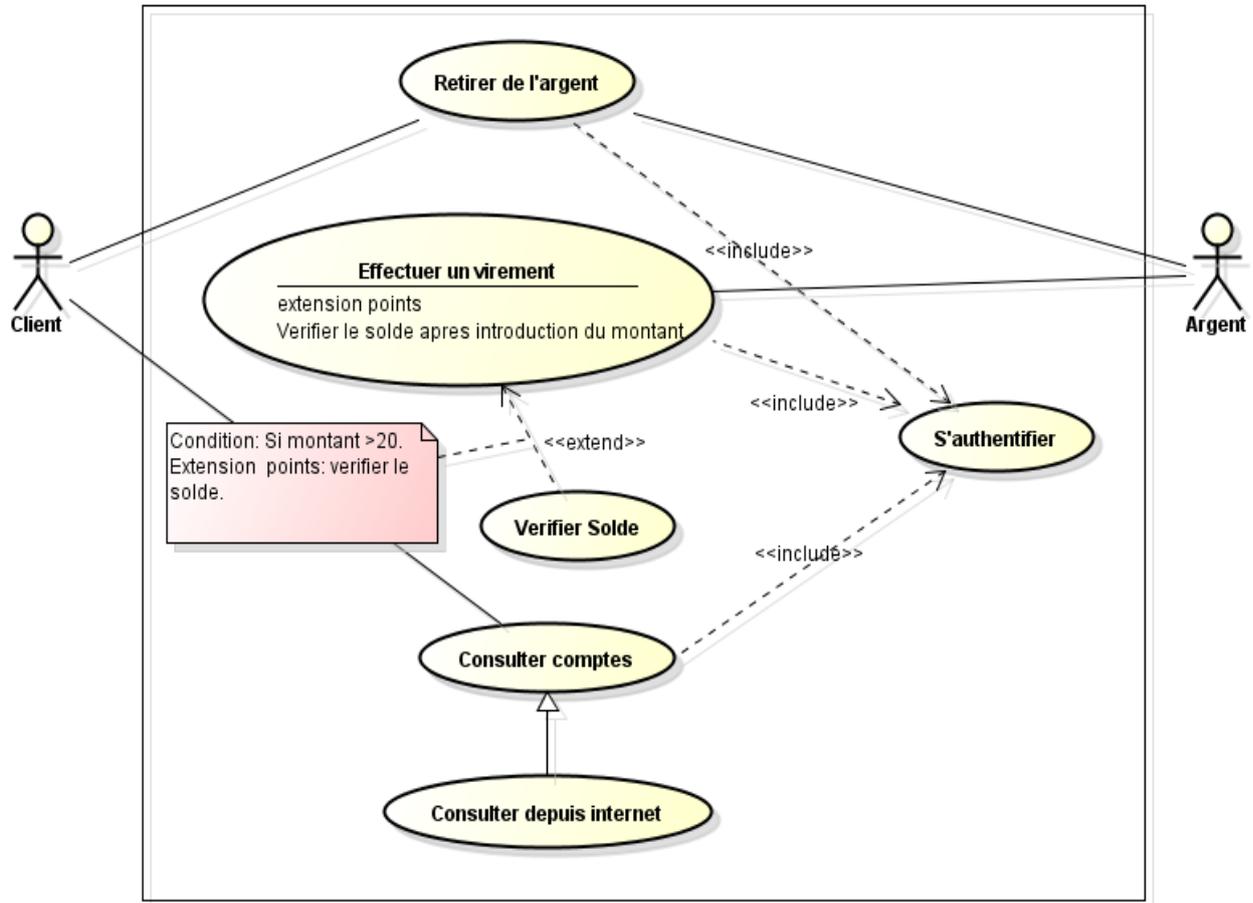
Présentation:

- Un cadre délimitant le système et contenant un ensemble de séquences d'actions (représentées par des cercles ovales) .
- A gauche on place les acteurs humains à droite les acteurs non humains.

Relation:

- L'inclusion  «include» : Le cas de base contient obligatoirement celui placé à l'extrémité de la flèche.
- L'extension  «extend» : Le cas de base contient optionnellement celui placé à l'origine de la flèche.
- Le point d'extension : Il est possible de préciser exactement à quel moment une extension est appelée
- La condition d'extension : Il est possible d'ajouter une condition (note) sous laquelle l'extension doit se produire.
- Spécialisation/généralisation  : Il définit la spécialisation d'un cas d'utilisation.

Exemple:



b) Le diagramme d'états:

Rôle: décrit les états que peut prendre le système et les transitions qui permettent les changements d'états.

Représentation:

● : Etat initial.

● : Etat intermédiaire.

⦿ : Etat Final.

Etat : Représente un moment spécifique du comportement du système qui correspond à une séquence: Entrée, phase active (ou attente), sortie.

—————→ : constitue un changement d'état allant d'un état source à un état cible.

Exercice: La bouilloire électrique (suite):

1) Soit les cas d'utilisations, les acteurs, relations d'associations suivants:

a) Cas d'utilisations:

Bouillir l'eau automatiquement.

Arrêter automatiquement le chauffage.

Arrêter manuellement le chauffage.

b) Les acteurs:

Utilisateur, l'eau et le secteur électrique.

c) relations d'associations:

Bouillir l'eau automatiquement \Rightarrow Arrêter automatiquement le chauffage.

Bouillir l'eau de manière automatique peut ne pas inclure automatiquement l'arrêt manuel du chauffage.

Définir le diagramme de cas d'utilisation.

2) La bouilloire électrique contient principalement trois états

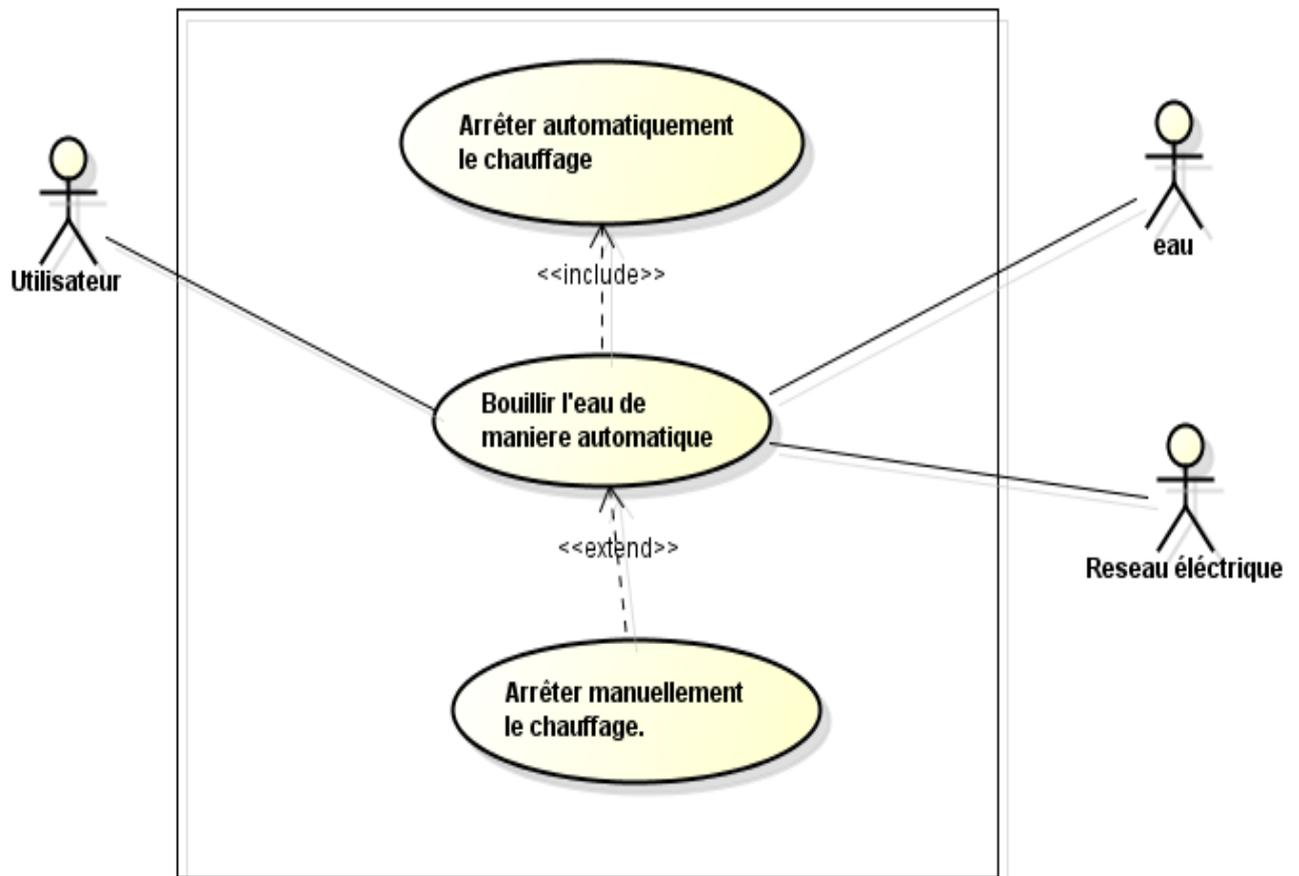
- Attente d'appuis sur ON (instruction chauffer a 0)

- Chauffer et mesurer température (instruction chauffer a 1 et temp=mesure(temp))

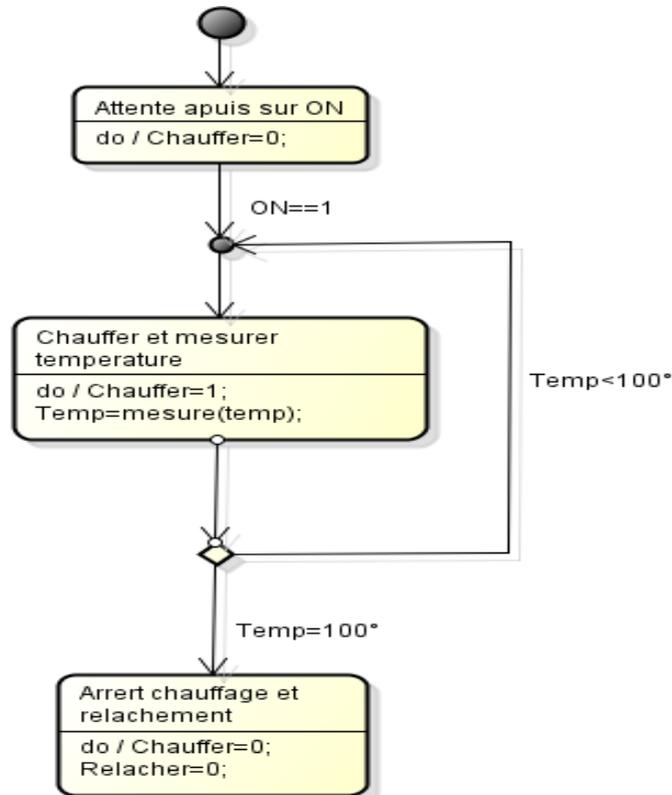
- Arrêt chauffage et relâchement(instruction chauffer a relâcher a 1).

Définir le diagramme d'états.

1) Le diagramme de cas d'utilisation.



2) Le diagramme d'états:



c) Le Diagramme de séquences:

Il permet de décrire l'enchaînement séquentiel des interactions du système en cas d'utilisation. Cela permet donc de montrer comment le système se comporte dans des scénarios de réussite comme dans des scénarios d'échec.

d) Diagramme d'activités:

Le diagramme d'activité représente les flots de données et de contrôles entre les actions. Il est utilisé majoritairement pour l'expression de la logique de contrôle et d'entrées/sorties.

6. Graphe pour projets, GANTT et PERT:

Constituent des représentations graphiques pour la planification de projets. La planification de projet va nous permettre de:

- Valider la date de fin du projet.
- Identifier les tâches critiques c'est-à-dire celles dont le retard entrainera le retard du projet.
- Anticiper le coûts.
- Coordonner les actions en cours.

L'élaboration d'une planification en utilisant "GANTT" et " PERT", s'effectue comme suit:

Exercice: Le lancement d'un nouveau produit nécessite la réalisation de tâches repérées par les lettres A à I suivantes :

Tache	Durée	Antécédent
A	5	D
B	2	G, H
C	5	B
D	4	
E	2	G, H
F	4	E, I
G	3	
H	2	D
I	6	A

- Etablissez la matrice des antériorités et classez les tâches par niveaux.
- Réalisez le graphe sagittal potentiel tâches.
- Réaliser le PERT potentiel tâches avec les dates au plus tôt, au plus tard, les marges libres et totales.

- A partir du Pert potentiel tâches réalisez le diagramme de Gantt.

6.1 Matrice des antériorités:

- Compléter les lignes en plaçant des "1" dans les colonnes où les tâches sont antérieures.

	A	B	C	D	E	F	G	H	I	N1	N2	N3	N4
A				1									
B							1	1					
C		1											
D													
E							1	1					
F					1				1				
G													
H				1									
I	1												

- Reporter la somme par ligne des "1" dans la colonne de niveau i.
- Déterminer quelles sont les tâches de niveau i (somme=0).
- Éliminer les "1" des colonnes de niveau i.

	A	B	C	D	E	F	G	H	I	N1	N2	N3	N4
A				1						1			
B							1	1		2			
C		1								1			
D										0			
E							1	1		2			
F					1				1	2			
G										0			
H				1						1			
I	1									1			
										D,G			

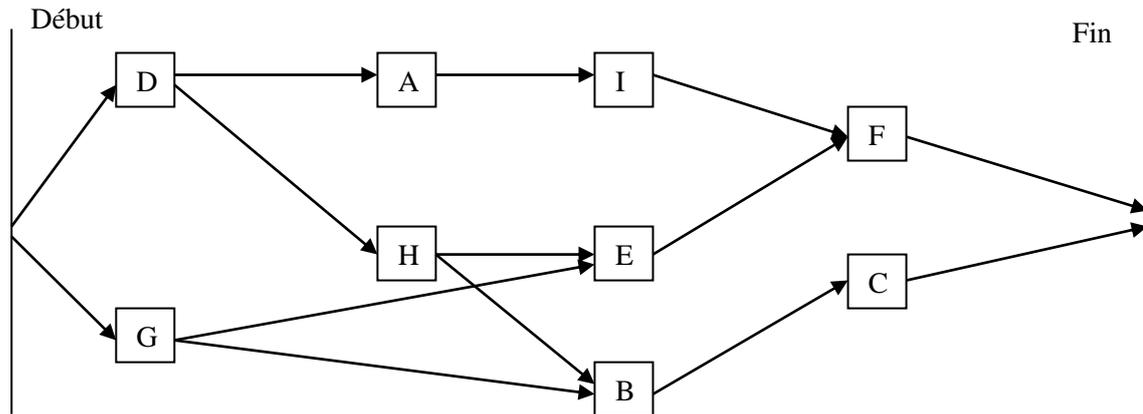
	A	B	C	D	E	F	G	H	I	N1	N2	N3	N4
A				1						1	0		
B							1	1		2	1		
C		1								1	1		
D										0	-		
E							1	1		2	1		
F					1				1	2	2		
G										0	-		
H				1						1	0		
I	1									1	1		
										D,G	A,H		

	A	B	C	D	E	F	G	H	I	N1	N2	N3	N4
A				1						1	0	-	
B							1	1		2	1	0	
C		1								1	1	1	
D										0	-	-	
E							1	1		2	1	0	
F					1				1	2	2	2	
G										0	-	-	
H				1						1	0	-	
I	1									1	1	0	
										D,G	A,H	B,E,I	

	A	B	C	D	E	F	G	H	I	N1	N2	N3	N4
A				1						1	0	-	-
B							1	1		2	1	0	-
C		1								1	1	1	0
D										0	-	-	-
E							1	1		2	1	0	-
F					1				1	2	2	2	0
G										0	-	-	-
H				1						1	0	-	-
I	1									1	1	0	-
										D,G	A,H	B,E,I	C,F

6.2 Graphe sagittal potentiel tâches:

- Les tâches représentent les sommets.
- Les vecteurs liant les sommets et orientés dans le sens du défilement (les relations de dépendance entre tâches).



6.3 PERT potentiel tâches:

Consiste à représenter chaque tache sous la forme suivantes:

(MT) marge totale	(DTA) début au plus tard	(ML) marge libre
(DTO) début au plus tôt	(D) durée	(FTO) fin au plus tôt
Nom de la tâche	(FTA) fin au plus tard	

a) Les dates au plus tôt:

$$DTO(T_i) = \text{Max}(FTO(T_{i-1})) \text{ avec } DTO(T_1) = 0$$

$$FTO(T_i) = DTO(T_i) + D_i$$

b) Les dates au plus tard:

$$DTA(T_i) = FTA(T_i) - D_i$$

$$FTA(T_i) = \text{Min}(DTA(T_{i+1})) \text{ avec } FTA(T_N) = FTO(T_N)$$

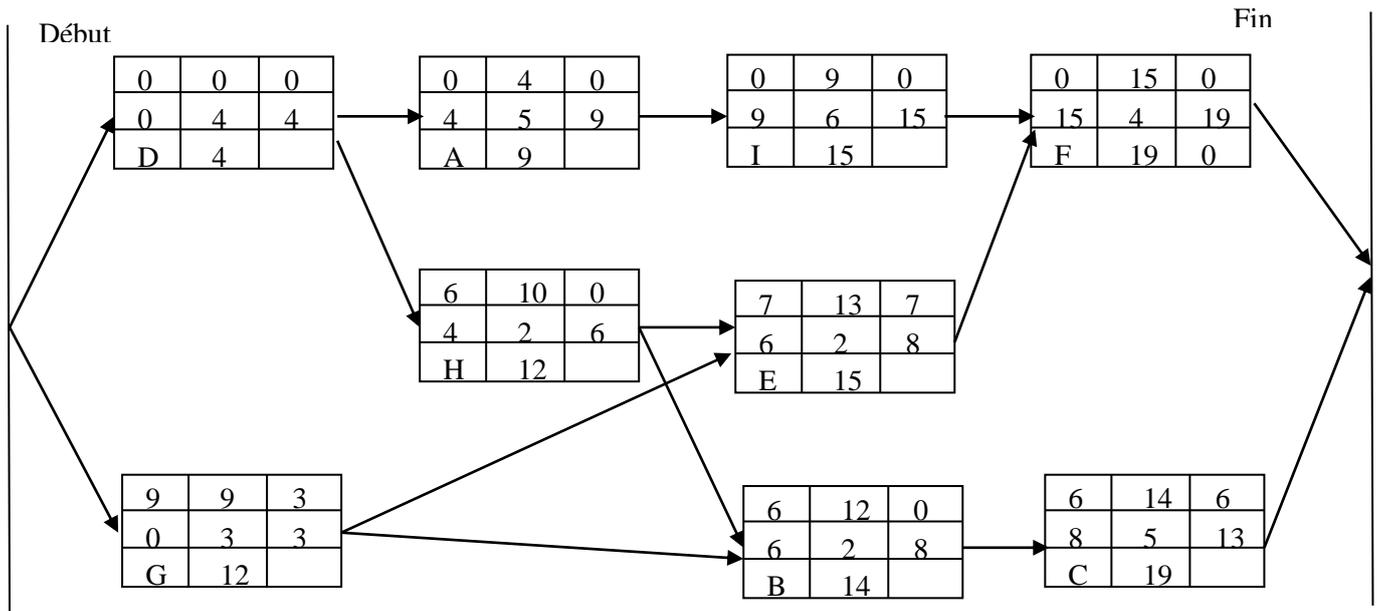
Remarque: S'il existe plusieurs tâches finales ($FTA(T_N) = \text{Max}(FTO(T_N))$).

c) Marge totale:

$MT(T_i) = DTA(T_i) - DTO(T_i)$ ou bien $MT(T_i) = FTA(T_i) - FTO(T_i)$.

d) Marge libre:

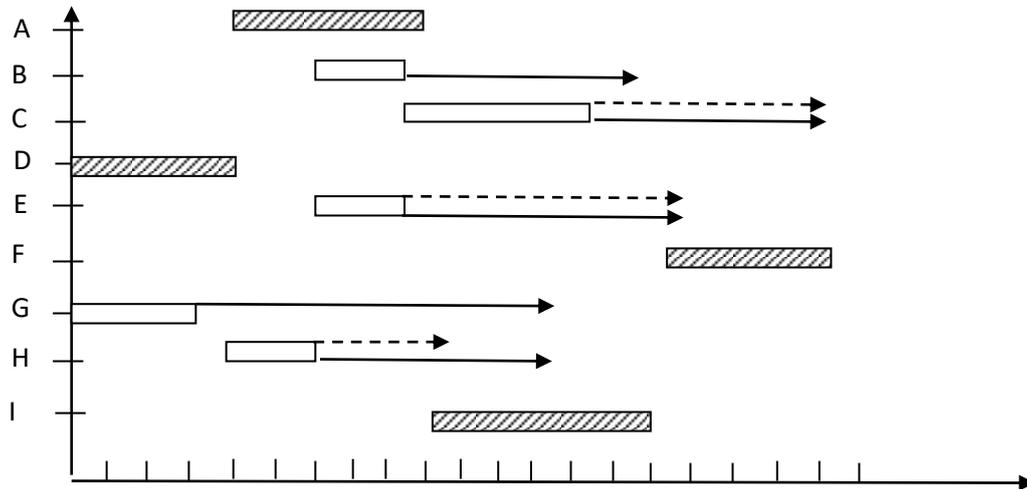
$ML(T_i) = \text{Min}(DTO(T_{i+1})) - FTO(T_i)$ avec $ML(T_N) = FTA(T_N) - FTO(T_N)$.



Remarque: Il est composé de tâches du réseau dont la marge totale est la plus faible. Un retard sur l'une des tâches du chemin critique implique un retard sur la fin du projet

6.4 Diagramme de Gantt:

- Les tâches sont représentées par des "barres" dont la longueur correspond à la durée (les liens d'antériorité doivent être respectés).



Application 1: Introduction au Systems Modeling Language (SysML). Etude des systèmes.

Rappel: Le langage SysML permet la modélisation des systèmes au tour de neuf diagrammes et selon trois points de vue:

- Point de vue "Fonctionnel" A quoi sert le système?.
- Point de vue "Structural" De quoi est il composé et Comment est il organisé?.
- Point de vue "Comportemental" quelles sont ces performances?.

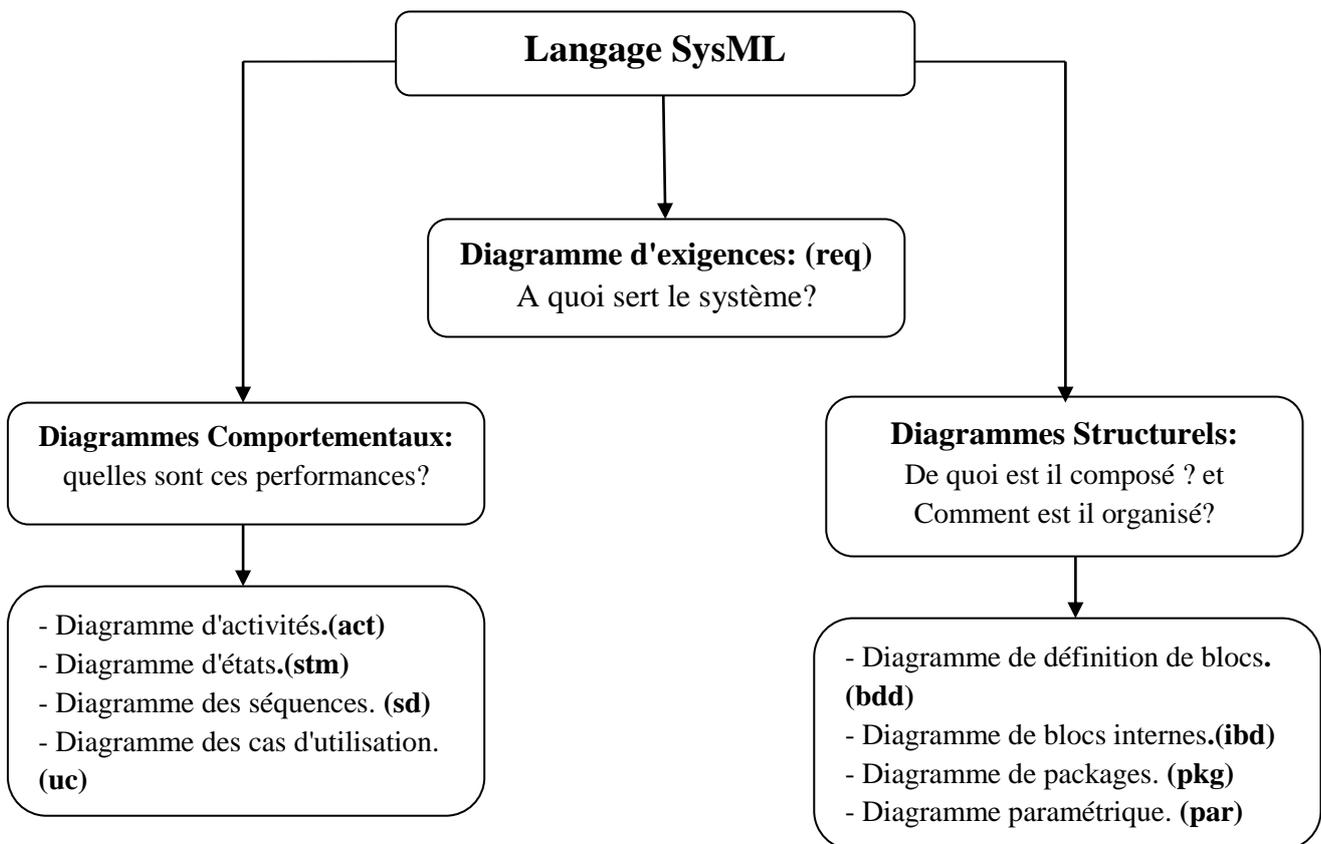


Figure1: Présentation générale des diagrammes SysML.

Exercice 1: (Aspirateur autonome TRILOBITE)

1) Représentation Fonctionnelle:

Présentation: L'aspirateur TRILOBITE est un appareil qui permet un nettoyage autonome sans l'intervention de l'utilisateur. TRILOBITE est capable:

- De retrouver sa station de recharge et de se recharger quand c'est nécessaire. Il a une autonomie de 15 minutes minimum par cycle de nettoyage.

- De se déplacer en toute sécurité dans toutes les pièces, à une vitesse maximale de 40cm/s.
- De détecter les escaliers pour ne pas tomber, et de détecter les objets et ne pas les abimer

Aussi,

- L'aspirateur offre à l'utilisateur une interface, qui lui permet de paramétrer le nettoyage et connaître l'état de la machine.
- L'aspirateur permet sans effort de vider la poussière stocker.

Travail à effectuer:

- A partir de la présentation, déterminer les exigences suivantes: Recharge, Déplacement, Vitesse, Autonomie, Gestion environnement, Paramétrage, et Entretien.
- Réaliser le diagramme d'exigence ainsi que les relations en cas de nécessité.
- Implémenter le diagramme sous Sysml.

2) Représentation Structurale:

2.1) Diagramme de définition de blocs (bdd):

Composition: L'aspirateur TRILOBITE est doté principalement:

- D'un bloc "système d'aspiration" qui est composé: d'une brosse, d'un bac à poussière, d'une turbine, d'une interface de puissance1 et , d'une interface de puissance2.
- Deux blocs "Moto propulseur" composés chacun: d'un moteur, et d'une roue.
- D'un bloc "détecteur" qui est composé: d'un détecteur d'obstacle (droit et gauche) et d'un détecteur de poussière.
- D'un bloc "station": (convertisseur DC->AC).
- D'un bloc " batterie": (convertisseurs DC->DC).
- Un bloc " IHM" qui est composé: de 4 boutons poussoirs (positions ON et OFF), et d'un afficheur graphique.

Travail à effectuer:

- Réaliser le diagramme de définition des blocs ainsi que les relations en cas de nécessité.
- Implémenter le diagramme sous Sysml.

2.2) Diagramme de blocs internes(ibd):

Nous nous intéressons dans cette partie à la représentation des échanges de matières /énergie/information/opération/service...etc, entre blocs du système d'aspiration, en se basant sur les entrées/sorties des différents blocs:

- Interface de puissance1 (relative à la brosse):

Entrées: - Alim-CC: Energie électrique CC.

- Brosse-ON: Signal logique.

Sorties: - V-Mot: Energie électrique commutée.

- Interface de puissance2 (relative à la turbine):

Entrées: - Alim-CC: Energie électrique CC.

- Turbine-ON: Signal logique.

Sorties: - V-Mot: Energie électrique AC.

- Brosse:

Entrées: - V-Mot: Energie électrique commutée.

Sorties: - Mouvement de rotation.

- Turbine:

Entrées: - V-Mot: Energie électrique AC.

Sorties: - Dépression.

- Bac à poussière:

Entrées: - Mouvement de rotation.

- Dépression.

Sorties: - Nettoyage (Aspiration).

3) Représentation comportementale:

3.1) Diagramme des cas d'utilisation (uc):

Soit les cas d'utilisations, les acteurs, relations d'associations suivants:

a) Cas d'utilisations:

- Nettoyage autonome de la pièce.

- Saisir les consignes.
- Visualiser les informations.
- Paramétrer le nettoyage.
- Arrêt manuel de l'aspirateur.

b) Acteurs:

- Utilisateur.
- Sol.

c) Relations d'associations:

- Le nettoyage autonome du sol est lancé après la validation des paramètres de nettoyage.
- Le nettoyage autonome du sol n'inclut pas forcément l'arrêt manuel de l'aspirateur.
- L'utilisateur doit paramétrer le nettoyage
- Le paramétrage du nettoyage inclut la saisie des consignes et la visualisation des paramètres.

Travail à effectuer:

- Réaliser le diagramme des cas d'utilisation.
- Implémenter le diagramme sous Sysml.

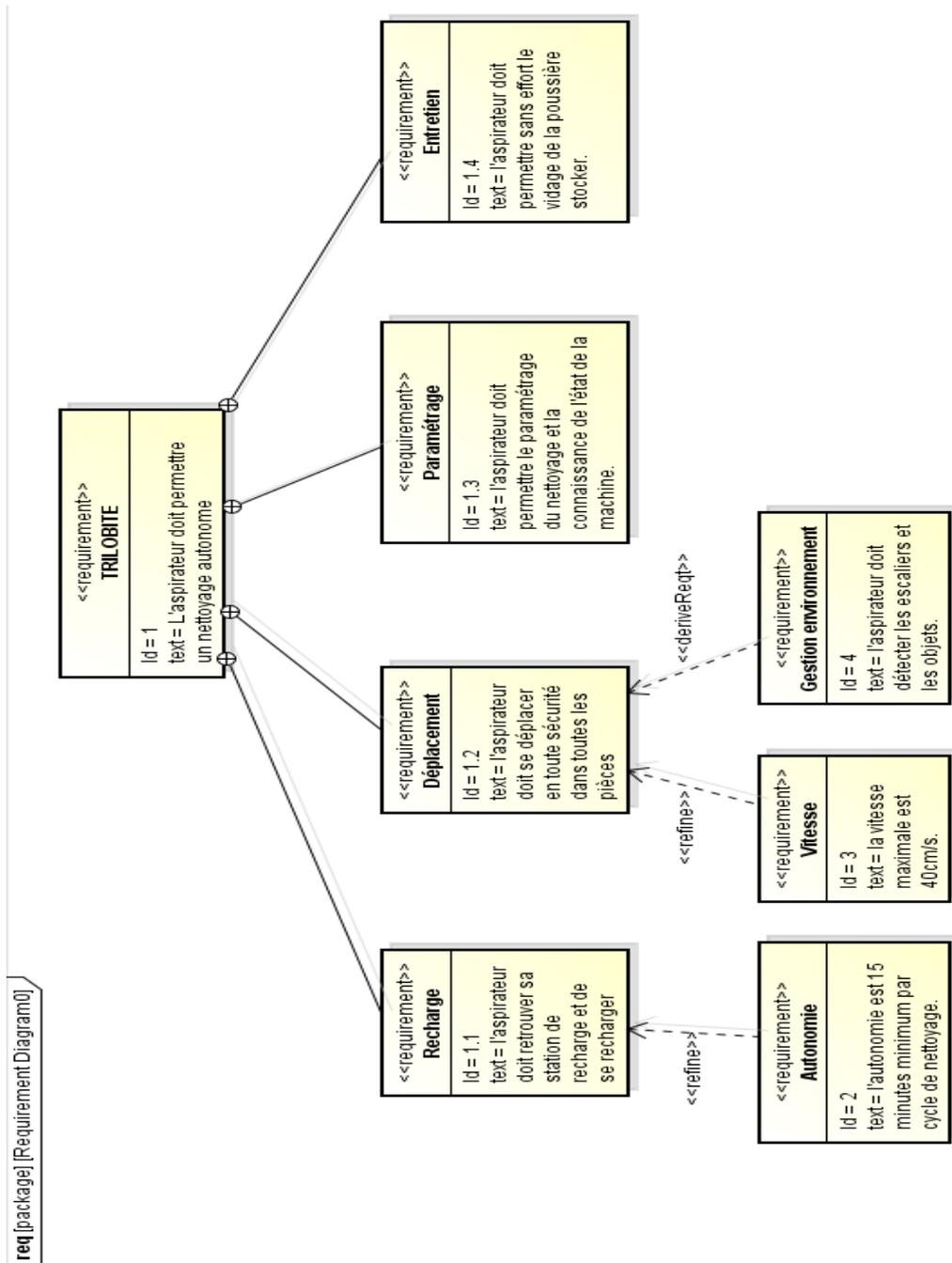
Solution Application 1.

1) Représentation Fonctionnelle:

A partir de la présentation, déterminer les exigences suivantes: TRILOBITE, Recharge, Déplacement, Vitesse, Autonomie, Gestion environnement, Paramétrage, et Entretien.

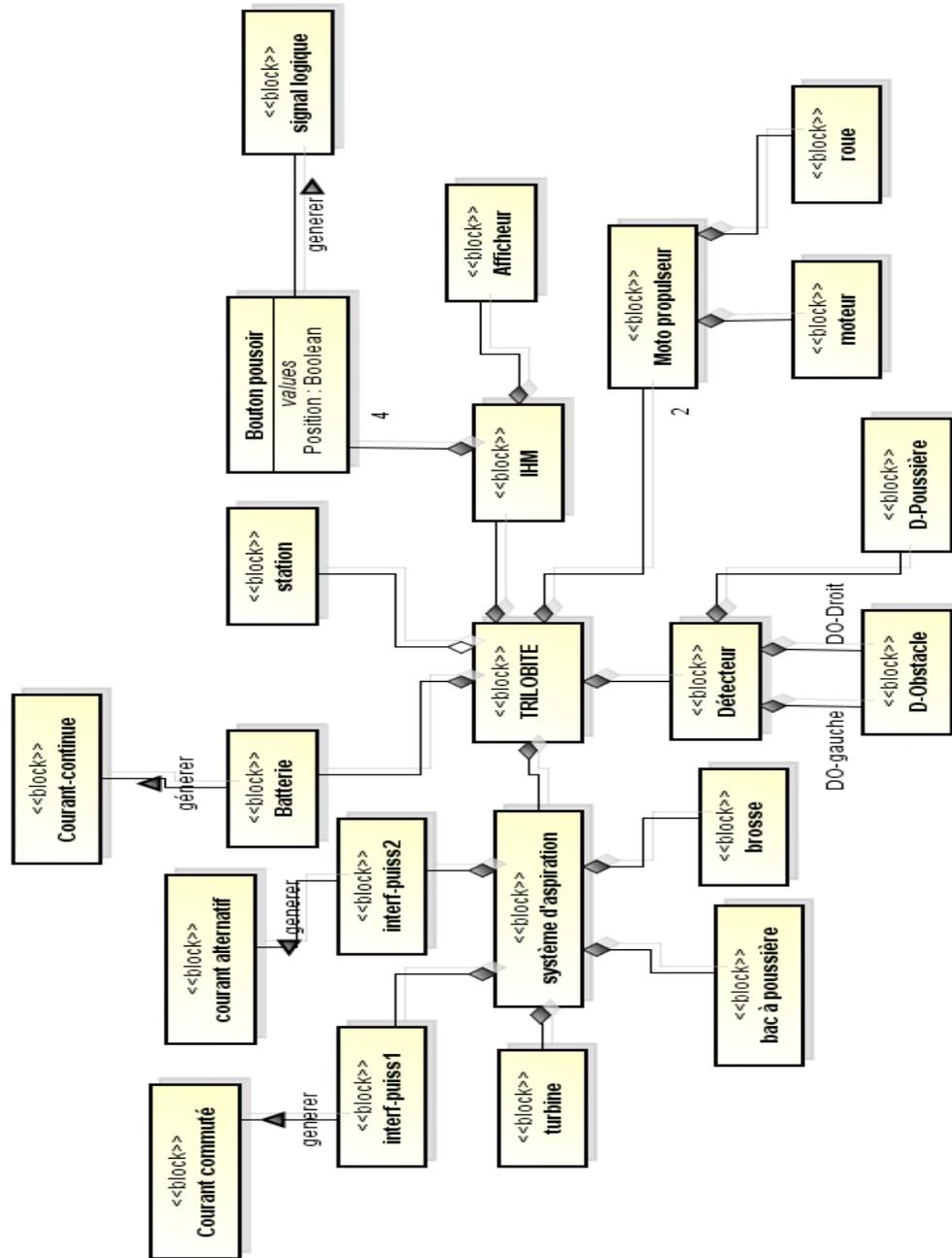
- TRILOBITE: L'aspirateur doit permettre un nettoyage autonome.
- Recharge: l'aspirateur doit retrouver sa station de recharge et de se recharger.
- Déplacement: l'aspirateur doit se déplacer en toute sécurité dans toutes les pièces.
- Vitesse: la vitesse maximale est 40cm/s.
- Autonomie: l'autonomie est 15 minutes minimum par cycle de nettoyage.
- Gestion environnement: l'aspirateur doit détecter les escaliers et les objets.
- Paramétrage: l'aspirateur doit permettre le paramétrage du nettoyage et la connaissance de l'état de la machine.
- Entretien: l'aspirateur doit permettre sans effort le vidage de la poussière stocker.

Réaliser le diagramme d'exigence ainsi que les relations en cas de nécessité et implémenter le diagramme sous Sysml

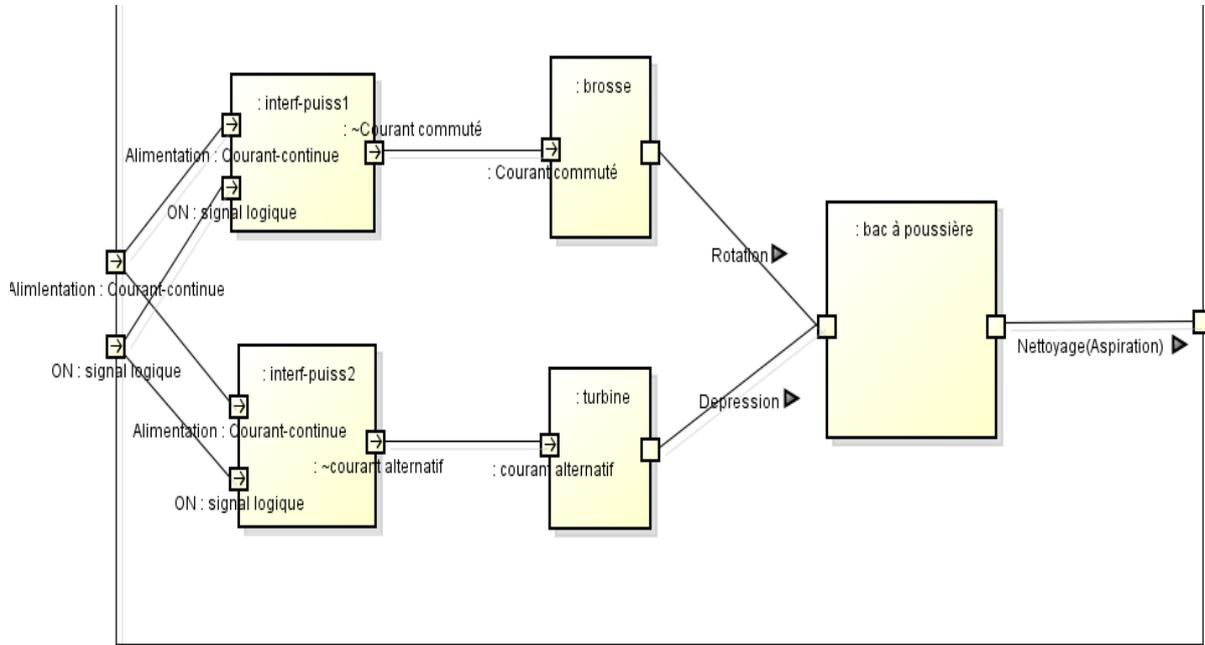


2) Représentation Structurale:

2.1) Diagramme de définition de blocs (bdd):

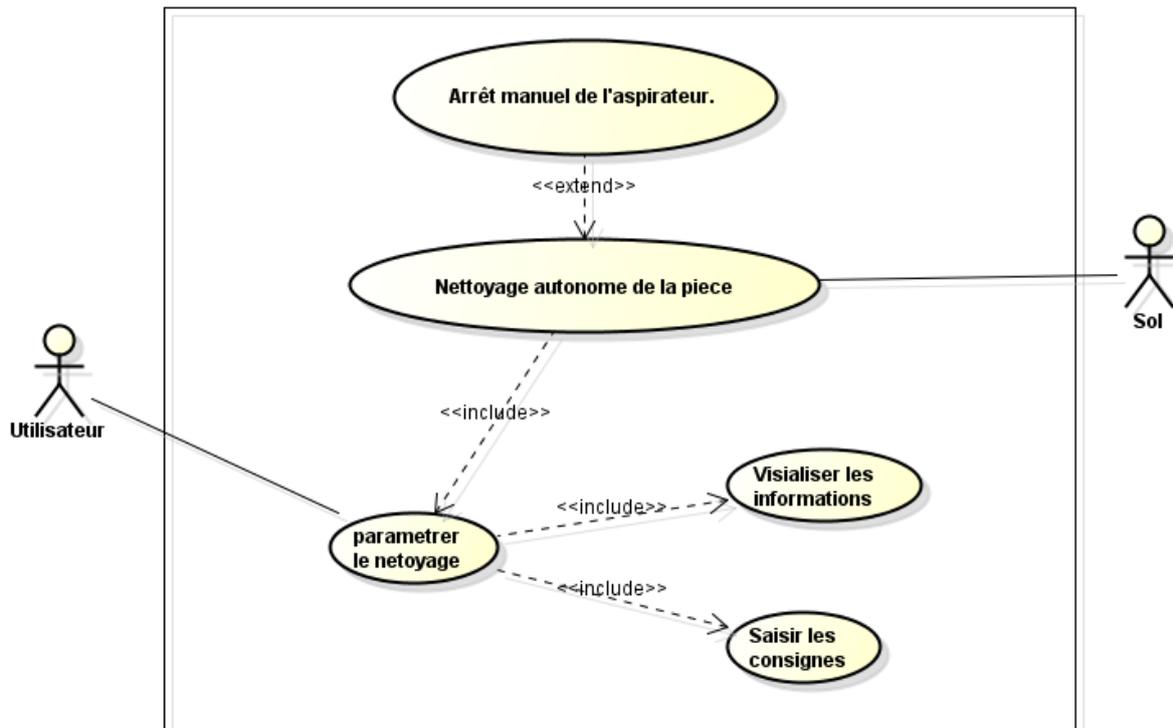


2.2) Diagramme de blocs internes(ibd):



3) Représentation comportementale:

3.1) Diagramme des cas d'utilisation (uc):



Application2: Planification graphique d'un projet sous Excel.

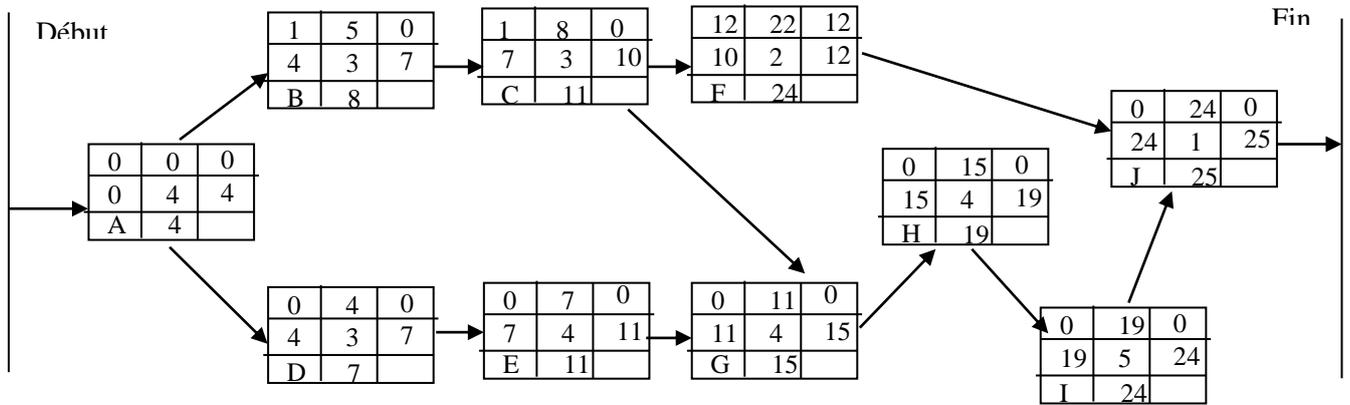
Exercice 1:

L'organisation d'un événement (fête, commémoration etc...) nécessite la réalisation de tâches repérées par les lettres A à J suivantes :

Lettre	Nom de la tâche	Durée	Antécédent
A	Définition du budget	4	
B	Sélection thème, date, lieu	3	A
C	Embauche traiteur	3	B
D	Annonce interne	3	B
E	Annonce de presse	4	D
F	Sélection menu	2	C
G	Location des équipements	4	C, E
H	Embauche personnel	4	G
I	Préparatifs	5	G
J	Evénement	1	I, H, F

- 1) Etablir la matrice des antériorités et classer les tâches par niveaux.
- 2) Réaliser le graphe sagittal potentiel tâches.

Soit le PERT potentiel tâches suivant:



3) A partir du Pert potentiel tâches, réaliser le diagramme de Gantt en spécifiant: les tâches critiques et non critiques, les marges libres et totales.

Solution Application 2.

1) Etablir la matrice des antériorités et classer les tâches par niveaux.

	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A																	
B	1																
C		1															
D		1															
E				1													
F			1														
G			1		1												
H							1										
I							1										
J						1		1	1								

	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0						
B	X										1						
C		1									1						
D		1									1						
E				1							1						
F			1								1						
G			1		1						2						
H							1				1						
I							1				1						
J						1		1	1		3						
											A						

	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0	-					
B	X										1	0					
C		X									1	1					
D		X									1	1					
E				1							1	1					
F			1								1	1					
G			1		1						2	2					
H							1				1	1					
I							1				1	1					
J						1		1	1		3	3					
											A	B					

	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0	-	-				
B	X										1	0	-				
C		X									1	1	0				
D		1									1	1	0				
E				X							1	1	1				
F			X								1	1	1				
G			1		1						2	2	2				
H							1				1	1	1				
I							1				1	1	1				
J						1		1	1		3	3	3				
											A	B	C, D				

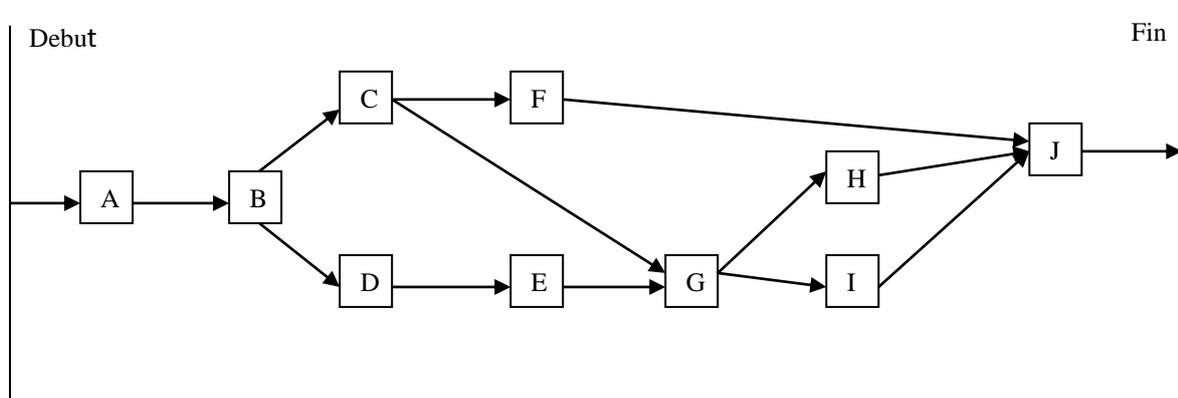
	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0	-	-	-			
B	X										1	0	-	-			
C		X									1	1	0	-			
D		1									1	1	0	-			
E				X							1	1	1	0			
F			X								1	1	1	0			
G			1		1						2	2	2	1			
H							1				1	1	1	1			
I							1				1	1	1	1			
J						X		1	1		3	3	3	3			
											A	B	C, D	E, F			

	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0	-	-	-	-		
B	X										1	0	-	-	-		
C		X									1	1	0	-	-		
D		X									1	1	0	-	-		
E				X							1	1	1	0	-		
F			X								1	1	1	0	-		
G			X		X						2	2	2	1	0		
H							X				1	1	1	1	1		
I							X				1	1	1	1	1		
J						X		1	1		3	3	3	3	2		
											A	B	C, D	E, F	G		

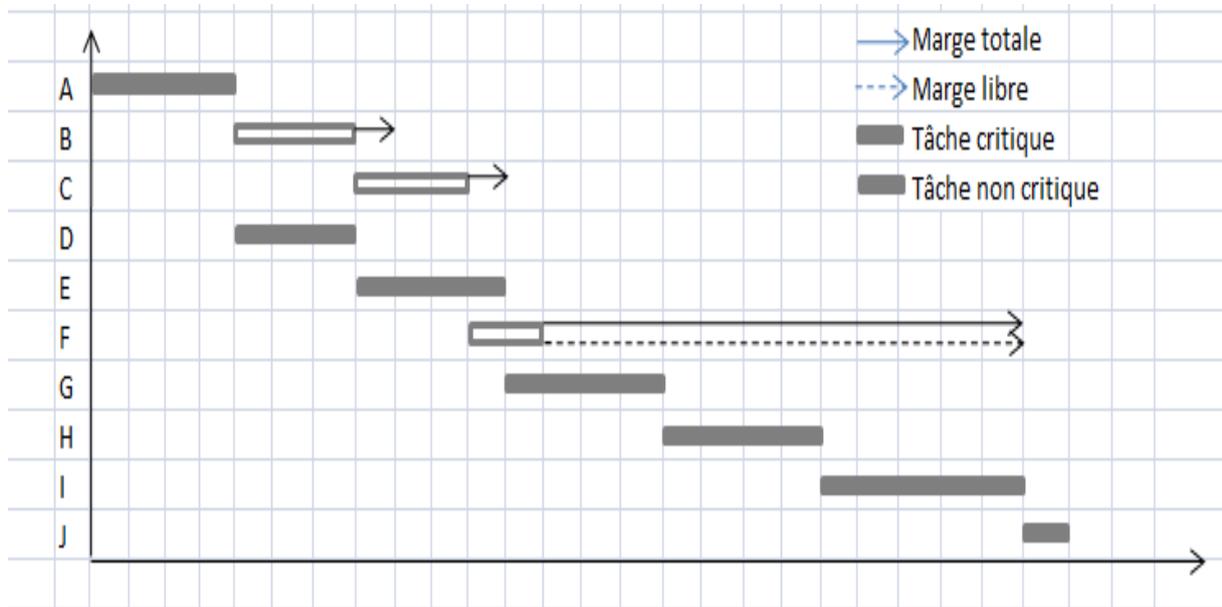
	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0	-	-	-	-	-	
B	X										1	0	-	-	-	-	
C		X									1	1	0	-	-	-	
D		X									1	1	0	-	-	-	
E				X							1	1	1	0	-	-	
F			X								1	1	1	0	-	-	
G			X		X						2	2	2	1	0	-	
H							X				1	1	1	1	1	0	
I							X				1	1	1	1	1	0	
J						X		X	X		3	3	3	3	2	2	
											A	B	C, D	E, F	G	H, I	

	A	B	C	D	E	F	G	H	I	J	N1	N2	N3	N4	N5	N6	N7
A											0	-	-	-	-	-	-
B	X										1	0	-	-	-	-	-
C		X									1	1	0	-	-	-	-
D		X									1	1	0	-	-	-	-
E				X							1	1	1	0	-	-	-
F			X								1	1	1	0	-	-	-
G			X		X						2	2	2	1	0	-	-
H								X			1	1	1	1	1	0	-
I								X			1	1	1	1	1	0	-
J						X		X	X		3	3	3	3	2	2	0
											A	B	C, D	E, F	G	H, I	J

2) Réaliser le graphe sagittal potentiel tâches.



3) A partir du Pert potentiel tâches, réaliser le diagramme de Gantt en spécifiant: les tâches critiques et non critiques, les marges libres et totales.



Bibliographie

http://www.lyc-hoche-versailles.ac-versailles.fr/IMG/pdf/sysML_cours.pdf

http://www4.ac-nancy-metz.fr/cpge-pmf-epinal/Cours_TD_SII/Modelisation_fonctionnelle_systeme.pdf

GÉRARD CASANOVA et DENIS ABÉCASSIS, " Gestion de projet - calcul des dates et calcul des marges".

GÉRARD CASANOVA et DENIS ABÉCASSIS, " Gestion de projet - réaliser le diagramme de PERT"

GÉRARD CASANOVA et DENIS ABÉCASSIS, " Gestion de projet - diagramme de Gantt"