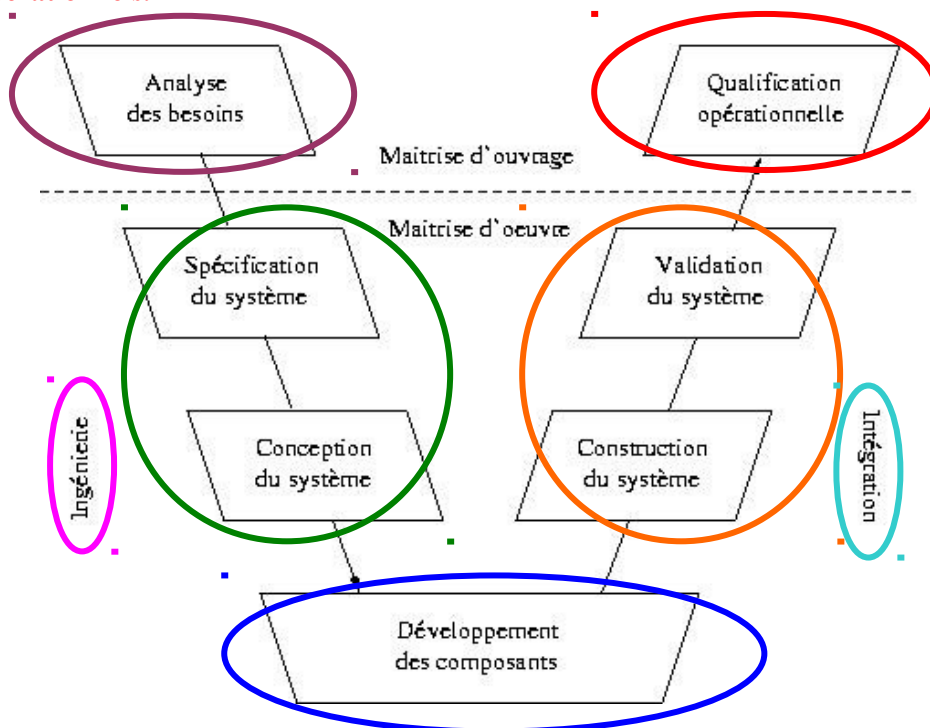


## METHODOLOGIE : INGENIERIE DES SYSTEMES

L'*ingénierie de systèmes* regroupe l'ensemble des activités de pilotage des projets de construction effective d'un système en s'appuyant sur sa décomposition architecturale produite par l'architecture de systèmes. Ce processus prend classiquement la forme d'un "cycle en V" en milieu industriel qui peut se décrire comme la succession des différentes phases d'ingénierie suivantes :

- ▶ la phase d'*analyse de besoins*, c'est-à-dire de définition de la mission du système ;
- ▶ la phase de *spécification & de conception du système* dans laquelle on définit les exigences que le système doit satisfaire pour remplir à bien sa mission, les sous-systèmes en lesquels il se décompose récursivement et les interactions qui existent entre ces sous-systèmes ;
- ▶ la phase de *développement des composants* qui composent le système et l'ensemble de ses sous-systèmes ;
- ▶ la phase d'*intégration du système*, c'est-à-dire de construction du système par l'assemblage progressif de tous les constituants du système et de test de la conformité du système aux exigences définies initialement (vérification & validation) ;
- ▶ la phase de *qualification opérationnelle*, c'est-à-dire de test de la conformité du système aux besoins opérationnels.



*Le cycle de développement en "V" d'un système industriel.*

On notera que les deux phases fondamentales suivantes du cycle en V (qui correspondent aux deux branches du V) se définissent naturellement du point de vue de la décomposition récursive d'un système :

- ▶ la phase d'ingénierie consiste en effet à définir l'arbre de décomposition du système à partir de la racine en partant d'une analyse des exigences de haut niveau, puis en spécifiant et en définissant de plus en plus précisément le système,
- ▶ la phase d'intégration consiste de son côté à parcourir l'arbre de décomposition du système des feuilles jusqu'à la racine pour interconnecter les éléments concrets correspondant aux définitions précédemment établies, tout en testant au fur et à mesure le système.

# Projets et études de cas

## Voitures solaires radiocommandées



## Aéroglisser radiocommandé



## Quadri copter radiocommandé



## Gyropode

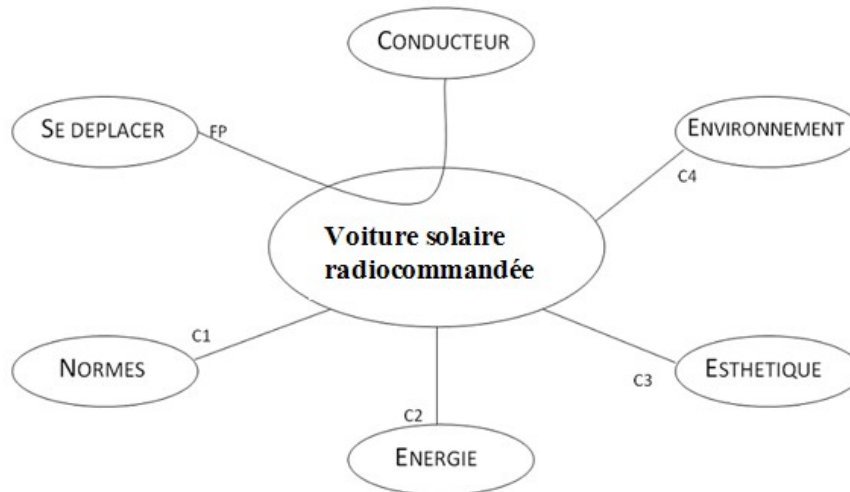


## Analyse fonctionnelle :

**Besoin à satisfaire : diagramme de prestation ou bête à corne**

**Les relations avec l'environnement : graphe des interactions ou diagramme pieuvre. Définir les fonctions de service FS que doit rendre le produit pour satisfaire l'utilisateur.**

**Distinguer dans les fonctions de services les fonctions principales FP des fonctions contraintes FC.**



**FP : Fonction Principale : Se déplacer**

**FC1 : Contrainte 1 : Respecter les normes**

**FC2 : Contrainte 2 : Utiliser l'énergie solaire**

**FC3 : Contrainte 3 : Etre esthétique**

**FC4 : Contrainte 4 : Aucun rejet de CO2**

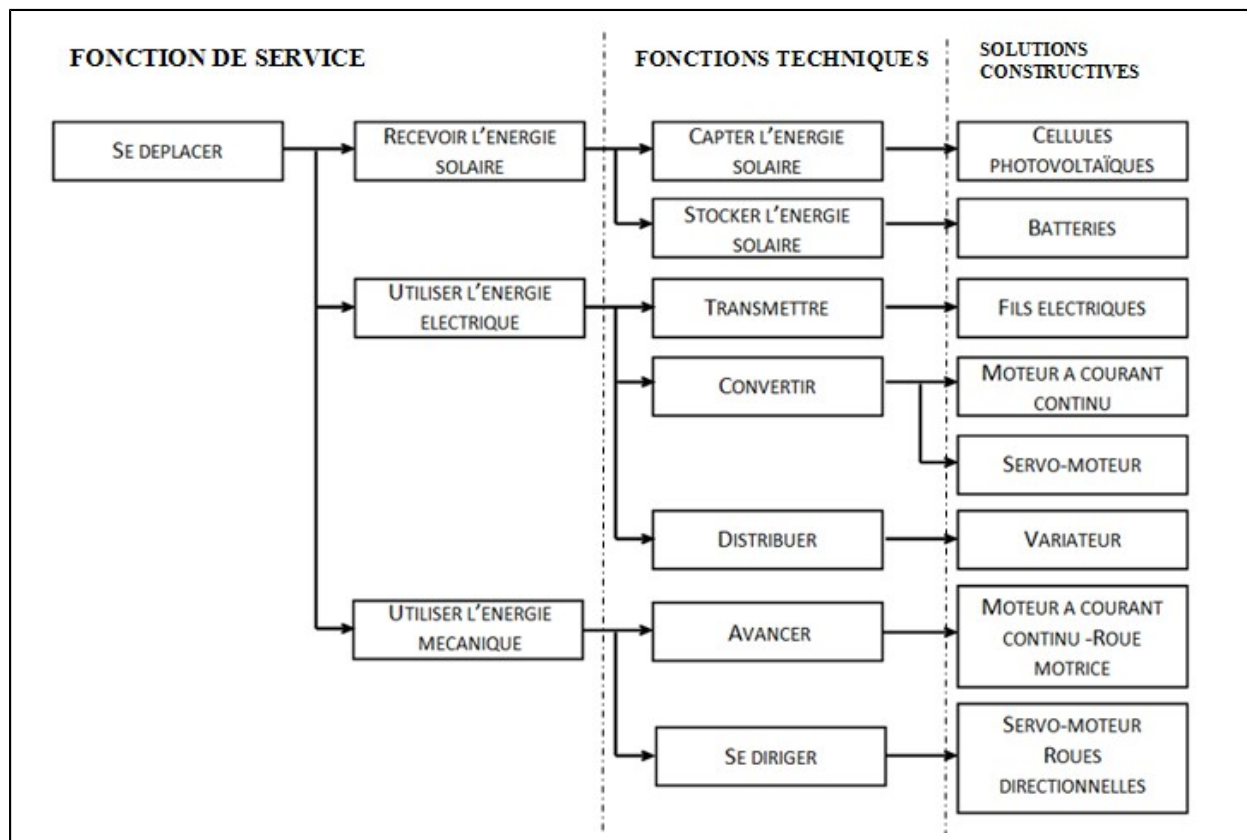
**Caractérisation des fonctions de services :** critère d'appréciation (durabilité, maintenabilité, vitesse, consommation, bruit de fonctionnement, rayon d'action, élément de confort, etc.) et niveau d'appréciation (valeur recherchée ou obtenue).

**Remarque pour une même fonction, il peut y avoir plusieurs critères d'appréciation de nature différente.**

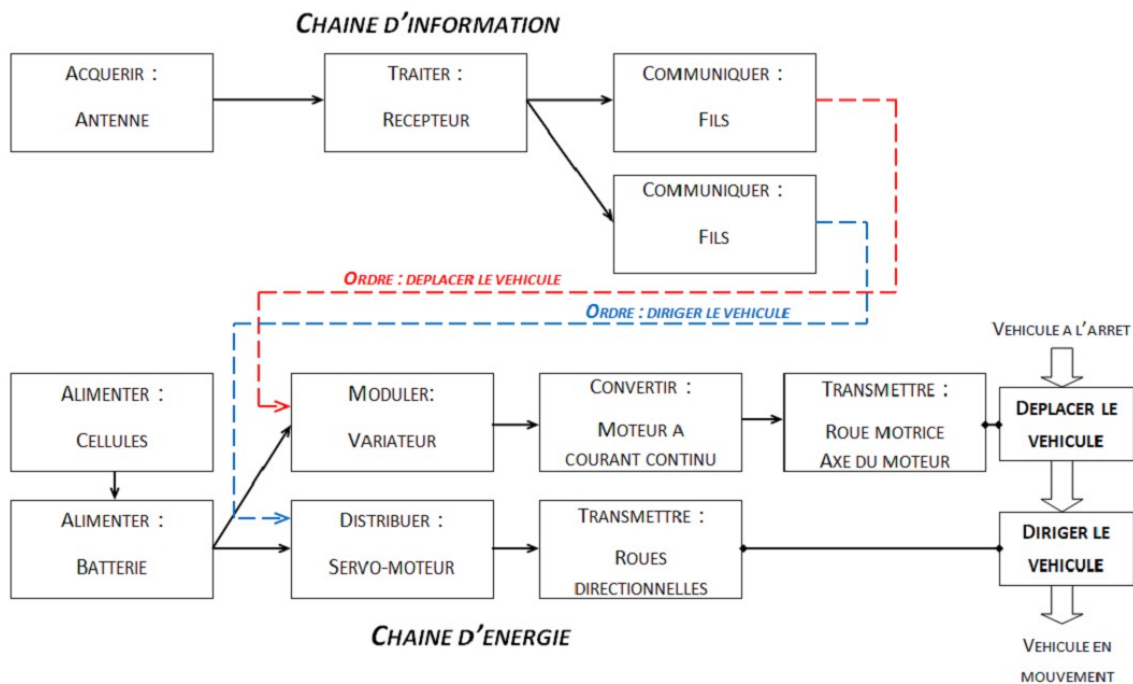
<b>Fonction</b>	<b>Critère d'appréciation</b>	<b>Niveau d'appréciation</b>
<b>FP : Se déplacer</b>	Vitesse, autonomie, monter	Slalom entre plot. Durée max d'endurance 4h, Pente de 20%
<b>FC1 : Respecter les normes</b>	Dimensions du véhicule, masse, sécurité, batterie déchargée, bande de fréquence radio	L=80cm, l=50cm et H=80cm. Garde au sol ≤ 5cm, pare-choc avt et ar ; Masse ≥ 1.5kg sans batterie Tension batterie ≤ 6,3V avec points de mesure Bande 2,4 Ghz en modulation FM
<b>FC2 : Utiliser l'énergie solaire</b>	Type de cellule photovoltaïque, surface maximum	S ≤ 2220cm <sup>2</sup>
<b>FC3 : Etre esthétique</b>	Forme, logo, couleur...	Zone de 6x6cm à l'avant et de 12cm de large et de 8cm de haut sur le côté gauche.
<b>FC4 : Aucun rejet de CO2</b>	Aucune pollution	0 g/Km de CO <sub>2</sub>

## Diagramme de décomposition fonctionnelle du produit : FAST

Le FAST (Function Analysis System Technic) est un diagramme qui traduit chaque fonction de service en fonction technique puis matériellement en solution constructive. Il se lit de gauche à droite dans une logique du pourquoi au comment.

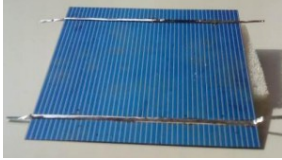


Les chaînes fonctionnelles d'énergie et d'information permettent de suivre les flux d'énergie et d'information.



## Analyse structurelle :

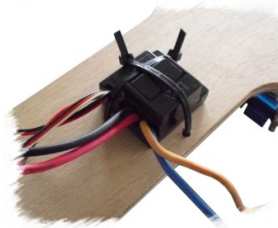
### LES CELLULES PHOTOVOLTAÏQUES



### LA BATTERIE



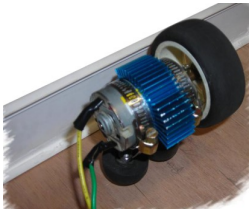
### LE VARIATEUR



### LE SERVO-MOTEUR



### LE MOTEUR



### LA RADIOCOMMANDE



### L'ANTENNE

### LE RECEPTEUR

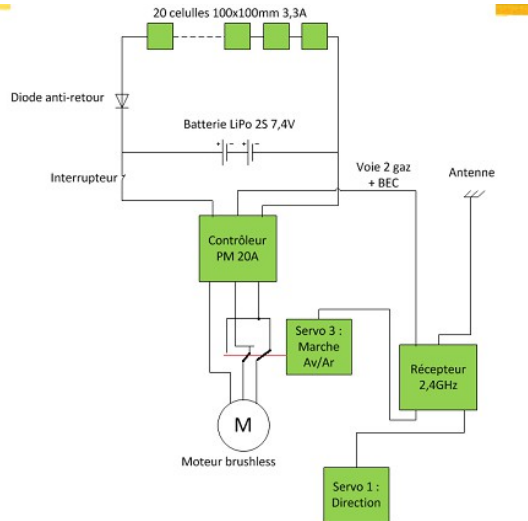




# CONSTRUCTION DU VEHICULE Par ASSEMBLAGE DES DIFFERENTS SOUS SYSTEMES

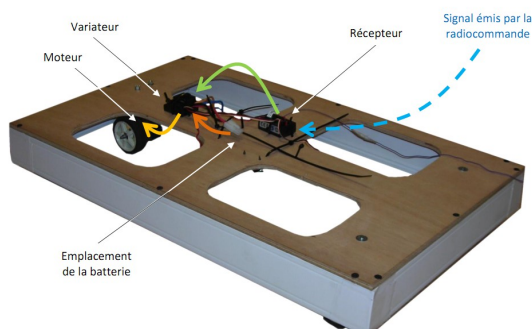


## LE SYSTEME D'ALIMENTATION



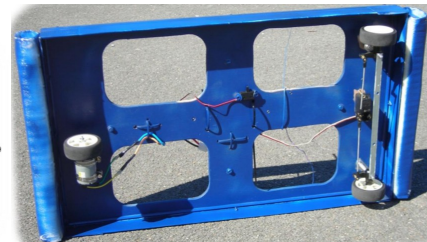
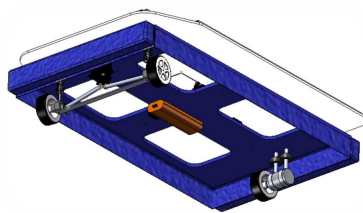
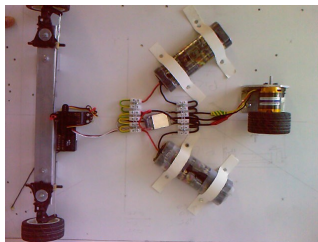
## LA GESTION DE L'ENERGIE

## LE CHASSIS DU VEHICULE

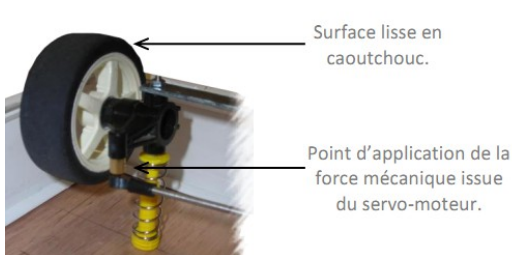


## LA COMMANDE

## L'ENSEMBLE MOTORISATION ET TRANSMISSION

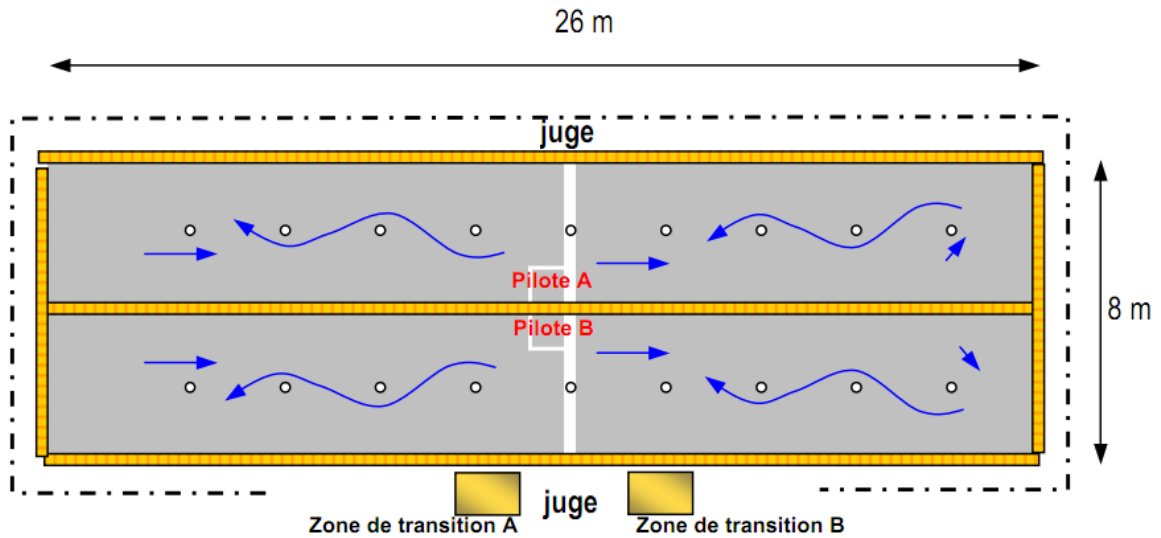


## LA DIRECTION

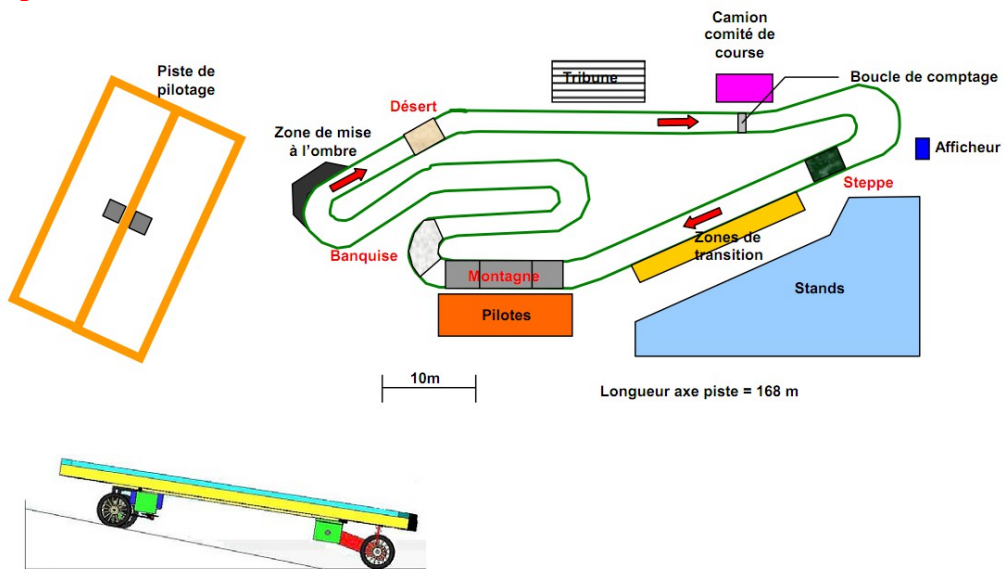


**VALIDATION DU SYSTEME** en effectuant des tests et en mesurant les écarts entre les solutions retenues et le cahier des charges fonctionnel (CdCF) qui spécifie les besoins auxquels répond le système.

### Epreuve de slalom



### Epreuve d'endurance



### Problèmes rencontrés

### Points forts/faibles

Un rapport écrit sous forme de dossier technique avec le carnet de bord doit rendre compte du travail d'ensemble.

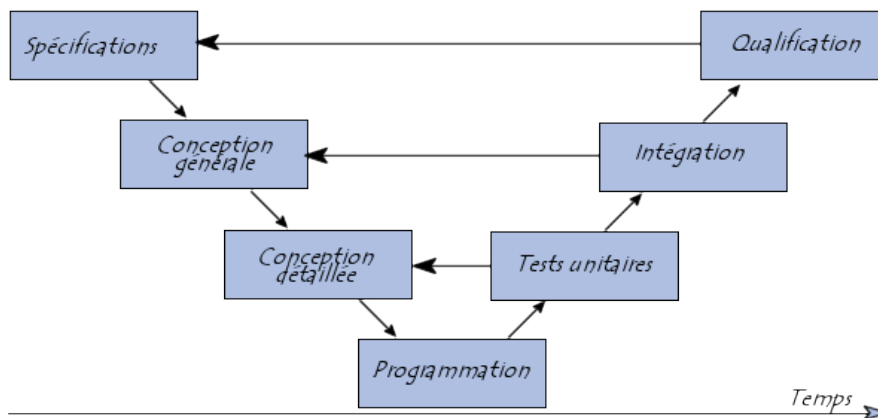
Enfin un exposé oral validera l'étude de cas ou le projet devant un jury.

## Annexe :

### Analogie avec la conception d'un logiciel

#### Cycle de vie d'un logiciel

Le « cycle de vie d'un logiciel » (en anglais *software lifecycle*), désigne toutes les étapes du développement d'un logiciel, de sa conception à sa disparition. L'objectif d'un tel découpage est de permettre de définir des jalons intermédiaires permettant la **validation** du développement logiciel, c'est-à-dire la conformité du logiciel avec les besoins exprimés, et la **vérification** du processus de développement, c'est-à-dire l'adéquation des méthodes mises en œuvre. L'origine de ce découpage provient du constat que les erreurs ont un coût d'autant plus élevé qu'elles sont détectées tardivement dans le processus de réalisation. Le cycle de vie permet de détecter les erreurs au plus tôt et ainsi de maîtriser la **qualité** du logiciel, les délais de sa réalisation et les coûts associés. Le modèle de cycle de vie en V part du principe que les procédures de vérification de la conformité du logiciel aux spécifications doivent être élaborées dès les phases de conception



Le cycle de vie du logiciel comprend généralement a minima les activités suivantes :

- **Définition des objectifs**, consistant à définir la finalité du projet et son inscription dans une stratégie globale.
- **Analyse des besoins et faisabilité**, c'est-à-dire l'expression, le recueil et la formalisation des besoins du demandeur (le client) et de l'ensemble des contraintes.
- **Conception générale**. Il s'agit de l'élaboration des spécifications de l'architecture générale du logiciel.
- **Conception détaillée**, consistant à définir précisément chaque sous-ensemble du logiciel.
- **Codage** (Implémentation ou programmation), soit la traduction dans un langage de programmation des fonctionnalités définies lors de phases de conception.
- **Tests unitaires**, permettant de vérifier individuellement que chaque sous-ensemble du logiciel est implémentée conformément aux spécifications.
- **Intégration**, dont l'objectif est de s'assurer de l'interfaçage des différents éléments (modules) du logiciel. Elle fait l'objet de *tests d'intégration* consignés dans un document.
- **Qualification** (ou *recette*), c'est-à-dire la vérification de la conformité du logiciel aux spécifications initiales.
- **Documentation**, visant à produire les informations nécessaires pour l'utilisation du logiciel et pour des développements ultérieurs.
- **Mise en production**,
- **Maintenance**, comprenant toutes les actions correctives (maintenance correctrice) et évolutives (maintenance évolutive) sur le logiciel.