

LE DIAGRAMME D'EXIGENCES

Un diagramme d'exigences est une représentation graphique permettant de définir, de lier et de gérer différents types d'exigences pour un système. Ces exigences peuvent inclure des contraintes ou spécifications fonctionnelles, non fonctionnelles et autres, définissant ce qu'un système doit faire ou comment il doit fonctionner.

Celui-ci fait partie des diagrammes de SysML. SysML (Systems Modeling Language) est un langage utilisé pour modéliser des systèmes. Il s'agit d'un dérivé du langage UML (Unified Modeling Language), adapté pour répondre à des besoins spécifiques. Il permet de représenter les aspects logiciels d'un système, mais aussi ses composants matériels, ses contraintes, ses exigences, ses performances, ainsi que les interactions entre ses différentes parties.

1. Les éléments principaux d'un diagramme d'exigences

Les diagrammes d'exigences sont composés des éléments suivants.

- **Les exigences** : elles sont représentées par un élément de type <<requirement>>. Elles sont définies par :
 - Un verbe,
 - Un identifiant (Id) représentant l'ordre affecté à l'exigence
 - Un texte qui décrit l'exigence. Ce texte doit être concis.

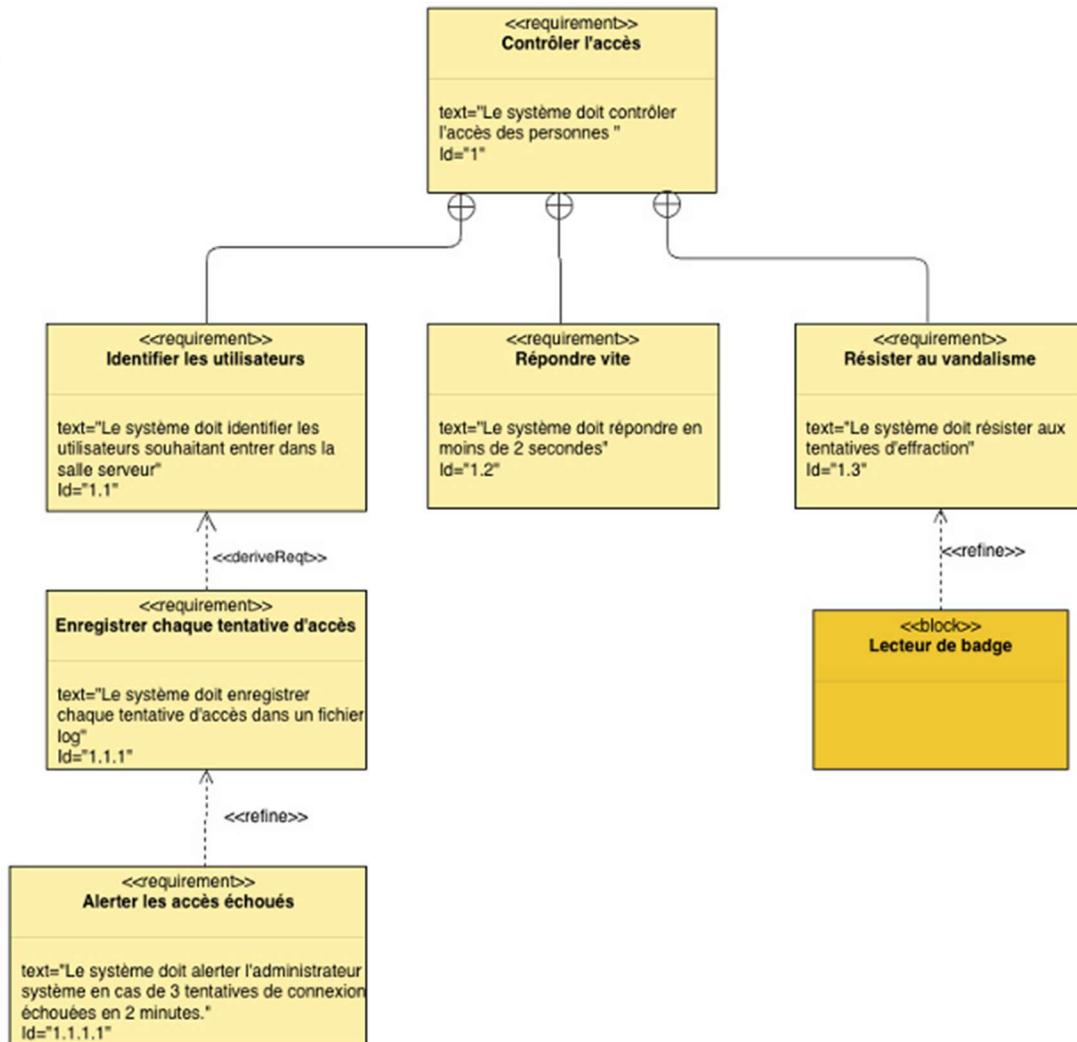
On doit y retrouver tous les éléments du cahier des charges.

- **Les relations** : elles permettent de positionner un lien entre les exigences. Il existe plusieurs types de lien dont notamment :
 - La **contenance** (<< composition>>) qui permet de décomposer une exigence complexe en plusieurs exigences unitaires. Cela se fait via une relation hiérarchique.
○—— Relation de composition d'exigences – l'exigence principale est décomposée en sous-exigences.
 - Le **raffinement** (« refine »), cette relation permet d'ajouter des précisions à une exigence, notamment en introduisant des données plus détaillées, quantitatives ou opérationnelles.
<<refine>> ou relation de raffinement – l'exigence est spécifiée de manière plus fine.
 - La **dérivation** (« deriveReqt ») qui consiste à relier des exigences de niveaux différents, par exemple des exigences système à des exigences de niveau sous-système.
<<deriveReqt>> ou relation d'exigence dérivée – l'exigence découle logiquement d'une autre.
Utilisé quand une exigence est déduite logiquement d'une autre.

- Les blocks sont les éléments structurels qui sont liés à une exigence par un <<refine>>.

2. Exemple de diagramme d'exigences

Système de gestion d'accès d'une salle serveur. Les exigences de ce système sont



1. L'**exigence** principale est : « Contrôler l'accès » donc id=“1”. Elle est décomposée en **plusieurs exigences** plus détaillées, pour mieux définir les spécifications fonctionnelles attendues. Les liens entre l'exigence principale et les exigences secondaires sont des **compositions**.

2. Les exigences secondaires sont :

- Identifier les utilisateurs (id=“1.1”);
- Répondre vite (id=“1.2”);
- Résister au vandalisme (id=“1.3”).

3. Décomposition de l'exigence 1.1 (Identifier les utilisateurs) :

- Enregistrer chaque tentative d'accès (id=“1.1.1”).
Le système doit enregistrer chaque tentative d'accès dans un fichier log.

Lien <**DerivReqt**>, cette exigence est dérivée de l'identification des utilisateurs.

- Alerter les accès échoués (id="1.1.1.1") : le système doit alerter l'administrateur en cas de 3 tentatives de connexion échouées en 2 minutes.

Lien <**refine**>, cette exigence affine ou spécifie davantage la précédente.

4. Le « block » lecteur de badge est lié à l'exigence 1.3 (« Résister au vandalisme ») via <>. Cela signifie que le lecteur de badge est choisi de manière à satisfaire cette exigence.

Ce diagramme permet de structurer les exigences d'un système de contrôle d'accès, de spécifier les fonctionnalités attendues (authentification, enregistrement, alerte, etc.). Il assure la traçabilité entre les besoins initiaux et les solutions techniques (comme le lecteur de badge) et prépare la vérification et la validation du système (chaque exigence doit être testée).

Le diagramme des exigences ne sert pas seulement à placer des contraintes dans le modèle. Il permet aussi de ramener des éléments venant d'autres diagrammes du modèle et ainsi de vérifier qu'une exigence a bien été prise en compte de quelque manière que ce soit, c'est-à-dire :

- *par un bloc (prise en compte par un élément structurel) ;*
- *par un élément reprenant un cas d'utilisation*