**Glossaire**

***Option et langage choisie***

L’option que j’ai choisie est l’option 2 et elle consiste a concevoir le métamodèle pour les sites pédagogique puis un métamodèle pour les API REST en Express Js (JavaScript). Express.js est un Framework de développement d'applications web pour Node.js, qui simplifie la création de serveurs HTTP. Il fournit des fonctionnalités pour gérer les requêtes HTTP, les routes, les middlewares, les sessions, les cookies, les Template, etc.

***Présentation du langage : Métamodèle et spécifications OCL***

Le métamodèle présenté ici définit une structure de modèle pour une application de gestion de formations universitaires. Ce modèle est défini en utilisant le langage de modélisation Ecore, qui est basé sur la norme de modélisation EMF (Eclipse Modeling Framework).

Le métamodèle comporte plusieurs classes qui représentent les entités principales de l'application de gestion de formations. La classe la plus importante est la classe Département, qui représente un département universitaire. Chaque département a un nom, une liste de formations, et un responsable. La classe Formation représente une formation universitaire, avec un nom, une description, un niveau et un ensemble de spécialités. La classe Niveau représente le niveau d'une formation universitaire, avec un niveau (ex : L1, L2, M1, M2), une liste de spécialités et un responsable. La classe Spécialité représente une spécialité de formation, avec un nom et une description. La classe Semestre représente un semestre d'une formation universitaire, avec un nom et une liste d'unités d'enseignement (UE). La classe UE représente une unité d'enseignement, avec un libellé, un code, un programme, une liste de ressources et une spécialité associée. La classe Ressource représente une ressource, avec un nom, une description, un type et un lien. La classe Programmation représente une programmation pour une unité d'enseignement, avec un jour, une salle et une heure de début et de fin. Enfin, la classe Responsable représente une personne responsable, avec un nom, un prénom et un grade.

En plus de la définition de la structure de données, le métamodèle inclut également des spécifications OCL (Object Constraint Language) pour définir des contraintes sur les données. Ces spécifications OCL incluent des invariants pour s'assurer que les données sont cohérentes, par exemple que le nom d'un département est unique ou que toutes les formations ont un nom valide. Les spécifications OCL incluent également des règles pour s'assurer que les relations entre les entités sont correctes, par exemple que chaque niveau de formation a une liste unique de spécialités associées.

En somme, ce métamodèle est destiné à être utilisé comme base pour une application de gestion de formations universitaires, et inclut des spécifications OCL pour s'assurer de la cohérence des données.

***Description d’une instance de métamodèle***Dans cette instance du métamodèle "Site", il y a une hiérarchie de nœuds qui décrivent la structure organisationnelle d'un département universitaire. En particulier, on peut voir que le département "Informatique" a deux formations (Licence Fonda et Master Fonda), chacune ayant plusieurs niveaux (L1, L2, L3, M1, M2) et chaque niveau contenant un certain nombre de semestres qui ont des unités d'enseignement (UE) associées. Les UE peuvent avoir des programmations spécifiques (jour, heure, salle) et des responsables.

En outre, il est possible de voir que les niveaux de L3, M1 et M2 ont des spécialités associées. Chaque spécialité a une description et est associée à des UE spécifiques.