



INF5039 : Ingénierie Dirigée Par les Modèles

Projet IDM 2021-2022

Modélisation de portails pédagogiques et génération de code

Etudiants :

- ATEMENGUE MOAFEMBE REGIS – 12U0631
- KUIIMO KAMGO CHRISTIAN BROWNDON – 14Y229

Responsable : Dr MONTHE VALERY – Chargé de Cours

Table des matières

I.	Présentation de l'option choisie	2
II.	Présentation de notre langage	2
1.	Méta-modèle	2
2.	Description d'une instance logique du modèle.....	5
3.	Spécifications non décrites sur le méta-modèle et contraintes OCL.....	6
III.	Règles de transformation et Génération de code.....	8
1.	Transformation Acceleo	8
2.	Processus de génération de code	9
IV.	Procédure d'exploitation du projet	10
V.	Exemple de résultat pour une instance du méta-modèle	11
1.	Modèle.....	11
2.	Site Obtenu	12

I. Présentation de l'option choisie

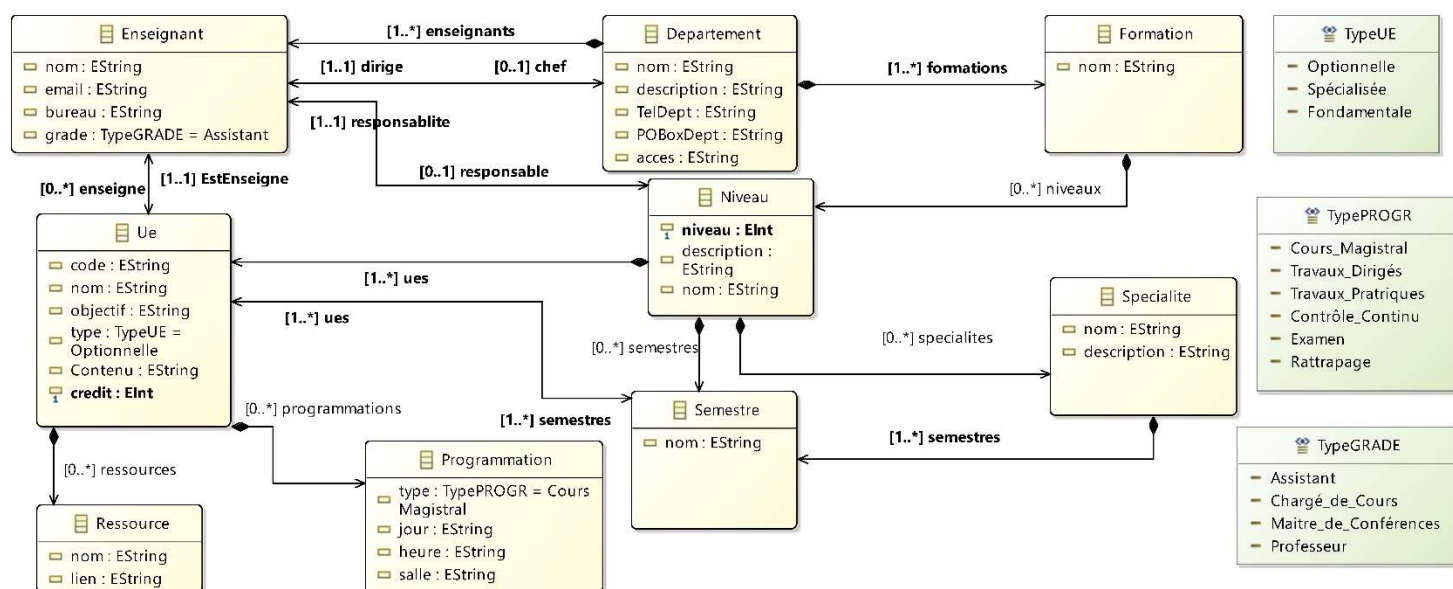
Pour moderniser la gestion du portail pédagogique, nous avons utilisé la première option qui consiste à passer de notre langage au code source du site :

- Proposer un langage
- Proposer les règles de transformation pour la génération du code
- Proposer le moteur de génération de code

II. Présentation de notre langage

1. Méta-modèle

Nous proposons un méta-modèle dont la classe principale est la classe Département. Ainsi, on se concentre sur la création d'un portail pédagogique pour un département précis. On a obtenu le méta-modèle suivant :



Nous avons exactement 9 classes et 3 types énumérés décrits comme suit :

CLASSE	ATTRIBUT	TYPE	DESCRIPTION
Département	Cette classe permet de définir le département qui est décrit sur le portail pédagogique		
	nom	EString	Permet de nommer un département, par exemple : Informatique
	description	Estring	Permet de faire une description générale du département sur son programme de formation
	TelDept	Estring	Permet d'associer les contacts téléphoniques utiles du département

	POBoxDept	EString	La boite postale du département
	acces	EString	Permet de présenter les conditions d'accès général pour accéder au département présenté
Formation	<i>Cette classe permet de définir l'ensemble des formations qui sont offerts dans un département</i>		
	Nom	EString	Désigne le nom d'une formation au sein du département
Enseignant	<i>Cette permet de décrit les enseignants qui sont enregistrés dans un département</i>		
	nom	EString	Les noms et prénoms d'un enseignant
	email	EString	Adresse e-mail d'un enseignant
	bureau	EString	Le nom du bureau où on peut trouver l'enseignant
	grade	TypeGRADE	Le grade de l'enseignant parmi une liste de grade prédéfinis dans un type énuméré
Niveau	<i>Cette classe permet de décrit les différents niveaux académiques que l'on retrouve dans une formation</i>		
	niveau	EInt	Numéro d'un niveau comme 1
	description	EString	Permet de faire une description générale du niveau sur son programme de formation
	nom	EString	Définit le nom du niveau comme Licence 1
Spécialité	<i>Cette classe permet de décrire les différentes spécialités de formation que l'on a dans un niveau s'il y'en a</i>		
	nom	EString	Définit le nom abrégé de la spécialité, comme GL
	description	EString	Définit le nom complet de la spécialité comme : Génie Logiciel
Semestre	<i>Cette classe permet de décrire les semestres que l'on retrouve dans un niveau ou dans une spécialité d'un niveau</i>		
	nom	EString	Définit le nom du semestre comme L1 S2
Ue	<i>Cette classe permet de décrire les unités d'enseignement qui sont dispensées dans un niveau puis qui seront associés à des semestres spécifiques</i>		
	code	EString	Définit le code de la matière comme INF5039
	nom	EString	Définit le nom complet associé à un code d'UE
	objectif	EString	Définit les objectifs généraux de l'UE

	type	TypeUE	Définit le type de l'UE parmi un type énuméré prédéfini
	Contenu	EString	Définit le plan du contenu de l'UE
	credit	Eint	Définit le nombre de crédit associé à l'UE
Programmation	<i>Cette classe permet de définir les différentes activités ou programmations associés à une UE. Il s'agit des programmations de cours, d'évaluation, de TD ou de TP</i>		
	type	TypePROG	Définit le type de programmation associé à une UE : cours, évaluation, TD ou TP
	jour	EString	Définit le jour associé à la programmation
	heure	EString	Définit l'heure associé à la programmation
	salle	EString	Définit la salle associée à la programmation
Ressources	<i>Cette classe définit les ressources associées à une UE. Il s'agit en général de fichiers pouvant être télécharger par un étudiant</i>		
	nom	EString	Définit le nom ou la description de la ressource
	lien	EString	Définit le lien vers la ressource

ENUMERATION	LITTERAL	DESCRIPTION
TypeUE	<i>Cette énumération permet de définir les types d'UE</i>	
	Optionnelle	Il s'agit d'une UE qui n'est pas obligatoire pour un étudiant
	Spécialisée	Il s'agit d'une UE qui est obligatoire dans une spécialité spécifique
	Fondamentale	Il s'agit d'une UE qui est obligatoire peut importe qu'on soit dans une spécialité ou pas
TypePROG	<i>Cette énumération permet de définir les types d'activités ou de programmation pour une UE</i>	
	Cours Magistrale	Il s'agit d'une activité de dispensation de cours
	Travaux Dirigés	Il s'agit d'une activité pour traiter des exercices

	Travaux Pratiques	Il s'agit d'une activité pour faire des exercices pratiques
	Contrôle Continu	Il s'agit d'une activité d'évaluation
	Examen	Il s'agit d'une activité d'évaluation
	Rattrapage	Il s'agit d'une activité d'évaluation
TypeGRADE	<i>Cette énumération permet de définir les types de grade pour un enseignant</i>	
	Assistant	
	Chargé de Cours	
	Maitre de Conférences	
	Professeur	

2. Description d'une instance logique du modèle

La logique de notre modèle lorsque l'on veut faire une instanciation est la suivante :

- Un département se compose d'un ensemble de formations et d'enseignants. Ainsi les formations et les enseignants sont définis dans un département.
- Un enseignant peut être chef dans un département et un département doit avoir un seul chef.
- Une formation se compose de plusieurs niveaux. Deux formations différentes ne doivent pas avoir le même nom.
- Un enseignant peut être responsable d'un niveau et un niveau doit être sous la responsabilité d'un seul enseignant.
- Un niveau se compose d'un ensemble d'unités d'enseignement, de spécialités, de semestres. Cependant, un niveau ne peut pas être composé en même temps de spécialité et de semestres. Deux niveaux ne doivent pas avoir le même nom.
- Deux unités d'enseignement ne doivent pas avoir le même code.
- Une spécialité se compose de semestres. Cette définition reste possible pour un niveau qui a des spécialités et donc le niveau ne sera pas directement composé de semestres.
- Dans une spécialité, on doit retrouver au moins une unité d'enseignement spécialisée.
- Un niveau ou une spécialité ne peut être associé qu'à deux semestres lorsque c'est possible. Ainsi, un niveau qui n'a pas de spécialité aura semestre 1 et semestre 1. Mais un niveau qui a des spécialités aura semestre 1 et semestre 2 de la spécialité X puis semestre 1 et semestre 2 de la spécialité Y.
- A un semestre, on associe des unités d'enseignement du niveau auquel le semestre est lié.
- Un enseignant peut enseigner des UE et une UE doit être enseigné par un enseignant.

- Une UE peut se composer de ressources et de programmation d'activités.
- Une UE spécialisée doit toujours être rattaché au semestre d'une spécialité.
- Une UE fondamentale dans un niveau qui a des spécialités doit être lié à un semestre de chaque spécialité.

3. Spécifications non décrites sur le méta-modèle et contraintes OCL

A partir de la description faite plus haut, on se rend compte que certaines spécifications ne sont pas définies à partir du diagramme. On a : (Les spécifications présentées en tiret dans une numérotation sont équivalentes et non donc pas besoin d'être définies toutes les deux)

1. Deux formations ne doivent pas avoir le même nom

Contexte : Formation

invariant FormNomDiff: *Formation.allInstances() -> forAll(p1, p2 | p1 <> p2 implies p1.nom <> p2.nom);*

2. Deux niveaux ne doivent pas avoir le même nom.

Contexte : Niveau

invariant NivNomDiff: *Niveau.allInstances() -> forAll(n1, n2 | n1 <> n2 implies n1.nom <> n2.nom);*

3. Deux unités d'enseignement ne doivent pas avoir le même code.

Contexte : Ue

invariant UeCodeDiff: *Ue.allInstances() -> forAll(ue1, ue2 | ue1 <> ue2 implies ue1.code <> ue2.code);*

4.

- Si un niveau est directement lié à des semestres alors il ne doit pas être directement lié à des spécialités.

Contexte : Niveau

invariant SiSemPasSpe: *semestres->size()>0 implies specialites->size()=0;*

- Si un niveau est directement lié à des spécialités alors il ne doit pas être directement lié à des semestres.

Contexte : Niveau

invariant SiSpePasSem: *specialites->size()>0 implies semestres->size()=0;*

5. Si un niveau n'est pas lié à des spécialités, alors il doit avoir uniquement deux semestres.

Contexte : Niveau

invariant SiSemAlors2: *specialites->size()=0 implies semestres->size()=2;*

6.

- Une spécialité dans un niveau doit avoir uniquement deux semestres.

Contexte : Spécialité

invariant NbreSem2: *semestres*->*size*()=2;

- Si un niveau est lié à des spécialités, alors il doit avoir un nombre de semestres égal au double du nombre de spécialité.

Contexte : Niveau

invariant SiSpeNsemDSpe: *specialites*->*size*()>0 **implies** *self.specialites.semestres*->*size*()=2**self.specialites*->*size*();

7. Une spécialité doit avoir au moins une unité d'enseignement spécialisée. On exclut le cas des unités spécialisées par semestre par ce qu'on peut avoir des niveaux ayant des spécialités mais sans matière spécialisée comme le cas où la seule matière est le mémoire.

Contexte : Spécialité

invariant UeSpeAM1: *self.semestres.ues* -> *exists*(ue | ue.type=TypeUE::Specialisee);

8. Une unité d'enseignement spécialisée doit être rattaché à un seul semestre qui est lui-même rattaché à une spécialité. Par soucis d'implémentation, elle a été découpée comme suit :

Une unité d'enseignement spécialisée doit être rattachée à un seul semestre.

Contexte : Ue

invariant UeSpe1Sem: *self.type*=TypeUE::Specialisee **implies** *self.semestres*->*size*()=1;

Une unité d'enseignement spécialisée d'un niveau doit être rattachée uniquement à des semestres inclus dans une spécialité

Contexte : Niveau

invariant UeSpeLieSemSpeNiv:
 self.ues -> *forall*(ue |
 ue.type=TypeUE::Specialisee **implies**
 ue.semestres -> *forall*(sem | *self.specialites.semestres* -> *includes*(sem))
);

9. Si un niveau n'a pas de spécialité alors une unité d'enseignement de ce niveau doit être liée à un seul semestre.

Contexte : Niveau

invariant PasSpeUe1Sem: *specialites*->*size*()=0 **implies** *self.ues* -> *forall*(ue | ue.semestres->*size*()=1);

10. Si un niveau est lié à des spécialités alors une unité d'enseignement fondamentale de ce niveau doit être liée à un seul semestre de chaque spécialité du niveau. Ceci traduit le fait que dans un niveau qui a des spécialités, une unité d'enseignement fondamentale doit être liée à un nombre

de semestre égal au nombre de spécialité ; en respectant le fait que ces spécialités soient différentes.

Contexte : Niveau

```
invariant SiSpeUeFon1SemPSpe: specialites->size()>0 implies self.ues -> forAll(ue |  
    ue.type=TypeUE::Fondamentale implies  
        self.specialites -> forAll(spe |  
            spe.semestres -> one(sem | sem.ues -> includes(ue))  
        )  
    );
```

11.

- Un semestre ne peut être lié qu'aux unités d'enseignement du niveau auquel il est lié.

Contexte : Niveau

```
invariant SemLieUeMemeNiv:  
    if specialites->size()==0 then  
        self.semestres.ues -> forAll(ue | self.ues -> includes(ue))  
    else  
        self.specialites.semestres.ues -> forAll(ue | self.ues -> includes(ue))  
    endif;
```

- Une unité d'enseignement ne peut être lié qu'aux semestres du niveau auquel il est lié.

Contexte : Niveau

```
invariant UeLieSemMemeNiv:  
    if specialites->size()==0 then  
        self.ues -> forAll( ue |  
            ue.semestres -> forAll(sem | self.semestres -> includes(sem))  
        )  
    else  
        self.ues -> forAll( ue |  
            ue.semestres -> forAll(sem | self.specialites.semestres -> includes(sem))  
        )  
    endif;
```

III. Règles de transformation et Génération de code

1. Transformation Acceleo

Une fois le méta-modèle réalisé, nous avons produite le code Acceleo permettant d'exploiter une instance du méta-modèle de notre langage pour générer un portail pédagogique correspondant à l'instance. Le code Acceleo se compose de quatre template principaux et deux template secondaires.

Les template secondaires sont :

- Le template **generateHautPage(aDepartement : Departement, titre : String, chemin : String)** qui permet de générer le code HTML d'en-tête et du header de toutes les pages du site.
- Le template **generateBasPage(a : String)** qui permet de générer le code de bas de page de toutes les pages site.

Les template principaux sont :

- Le template `generateElement(aDepartement : Departement)` qui permet de générer le fichier `index.html` de notre site qui présentera le département et les formations disponibles au travers de leurs niveaux.
- Le template `generateNiveau(aNiveau : Niveau)` qui permet de générer un fichier html pour chaque niveau qui présentera le niveau, son responsable et les semestres au travers de leurs unités d'enseignement.
- Le template `generateUe(aUe : Ue)` qui permet de générer un fichier html pour chaque UE qui présentera toutes ses informations.
- Le template `generateEnseignant(aDepartement : Departement)` qui permet de générer le fichier `enseignant.html` de notre site qui présentera les enseignants du département, leurs informations et leur responsabilités.

```

generate.mtl x  ContenuPedagogique.ecore  ContenuPedagogique.aird  ContenuPedagogique class diagram  ContenuPedagogique.ecore
1  [comment encoding = UTF-8 /]
2  /**
3   * The documentation of the module generate.
4   */
5  [module generate('http://www.ContenuPedagogique.com')]
6  /**
7   * The documentation of the template generateElement.
8   * @param aDepartement
9   */
10 [template public generateHautPage(aDepartement : Departement, titre : String, chemin : String)]
11 [/template]
12
13 [template public generateBasPage(a : String)]
14 [/template]
15
16 [template public generateElement(aDepartement : Departement)]
17 [/template]
18
19 [template public generateNiveau(aNiveau : Niveau)]
20 [/template]
21
22 [template public generateUe(aUe : Ue)]
23 [/template]
24
25 [template public generateEnseignant(aDepartement : Departement)]
26 [comment @main/]
27 [file (aDepartement.nom+"/enseignant.html", false, 'UTF-8')]
28 [generateHautPage(aDepartement, 'Enseignant du Département', '')/]
29 <div id="navigation">
30   <nav class="nav">
31     <a class="nav-link" href="index.html">Accueil </a>
32     [for (formation : Formation | aDepartement.formations)]
33     <a type="button" data-target="#collapseExample[formation.nom.replaceAll(' ','_')]" aria-expanded="false" aria-controls="#collapseExample[formation.nom.replace
34     [for]
35     <a class="nav-link" href="enseignant.html">Enseignants </a>
36   </nav>
37 </div>

```

2. Processus de génération de code

Pour pouvoir générer le code d'une instance déjà créée, il faut accéder au fichier mtl contenant le code Accileo puis ouvrir les paramètres de configurations du fichier pour configurer et lancer le processus de génération du code.

Le code obtenu est ranger dans un dossier portant le nom du département de l'instance, et dans ce dossier, on a deux dossiers, un fichier `index.html`, un fichier `enseignant.html` et un fichier `style.css`. Soit, un dossier « ue » ayant les fichiers HTML des UEs et un dossier « niveau » ayant les fichiers HTML des niveaux.

Il faudra ajouter un dossier images ayant deux fichiers `logo.png` et `user.png` qui représente respectivement le logo de l'université du département et une photo par défaut associée à tous les enseignants.

IV. Procédure d'exploitation du projet

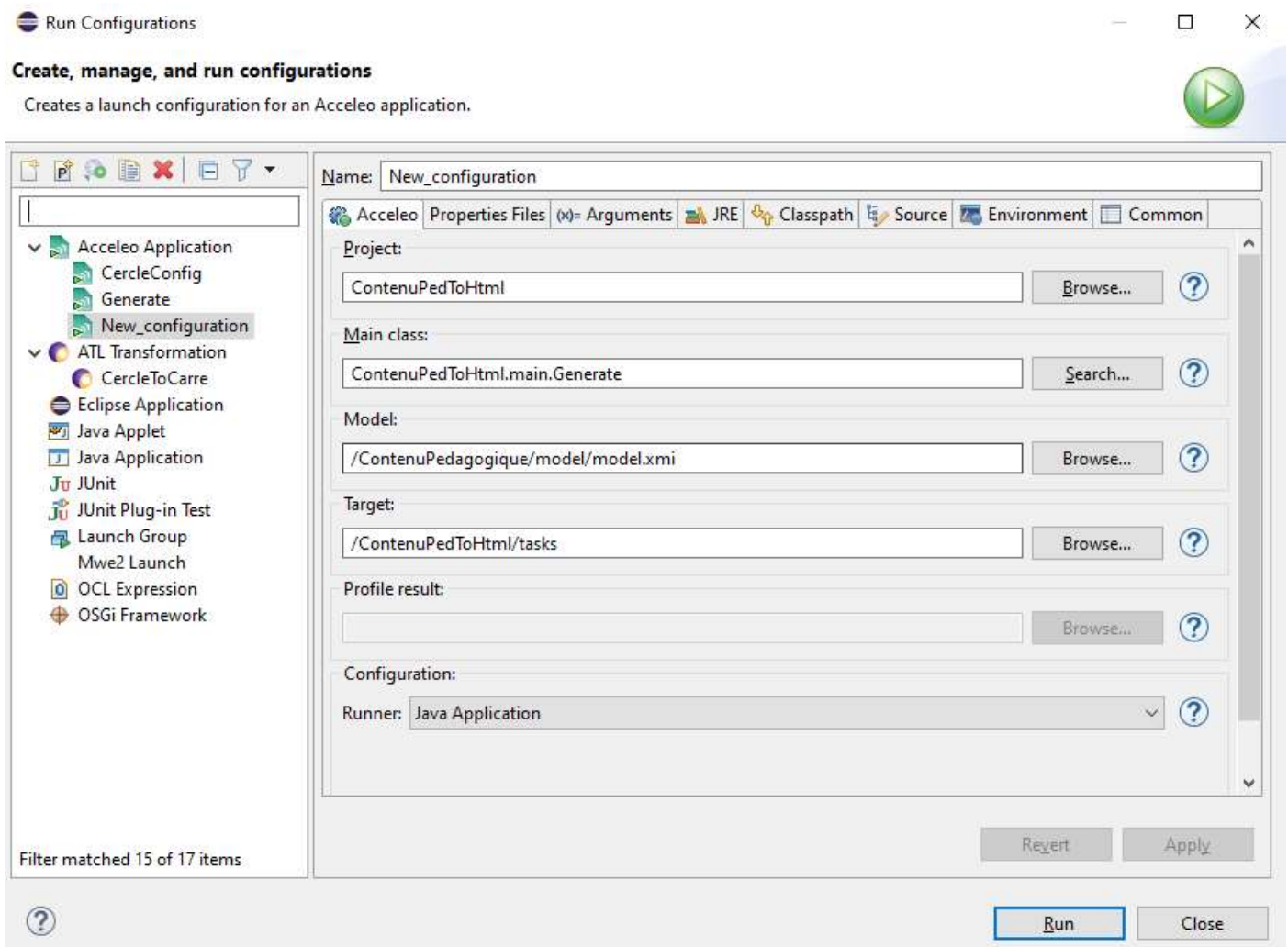
En plus de cette documentation, le projet est livré avec 4 dossiers qu'il faudra importer dans Eclipse. Il s'agit de :

- ContenuPedagogique
- ContenuPedagogique.edit
- ContenuPedagogique.editor
- ContenuPedToHtml

Le projet est livré avec un modèle déjà crée et se trouvant dans le dossier model du Projet ContenuPedagogique. Néanmoins, on peut créer un autre modèle à partir du fichier ContenuPedagogique.ecore dans le dossier metatmodels du projet ContenuPedagogique.

Le code OCL lié au méta-modèle peut être accéder en faisant un clic droit sur le fichier ecore puis en prenant l'option « OCLinEcore Editor » dans le sous menu « Open With ».

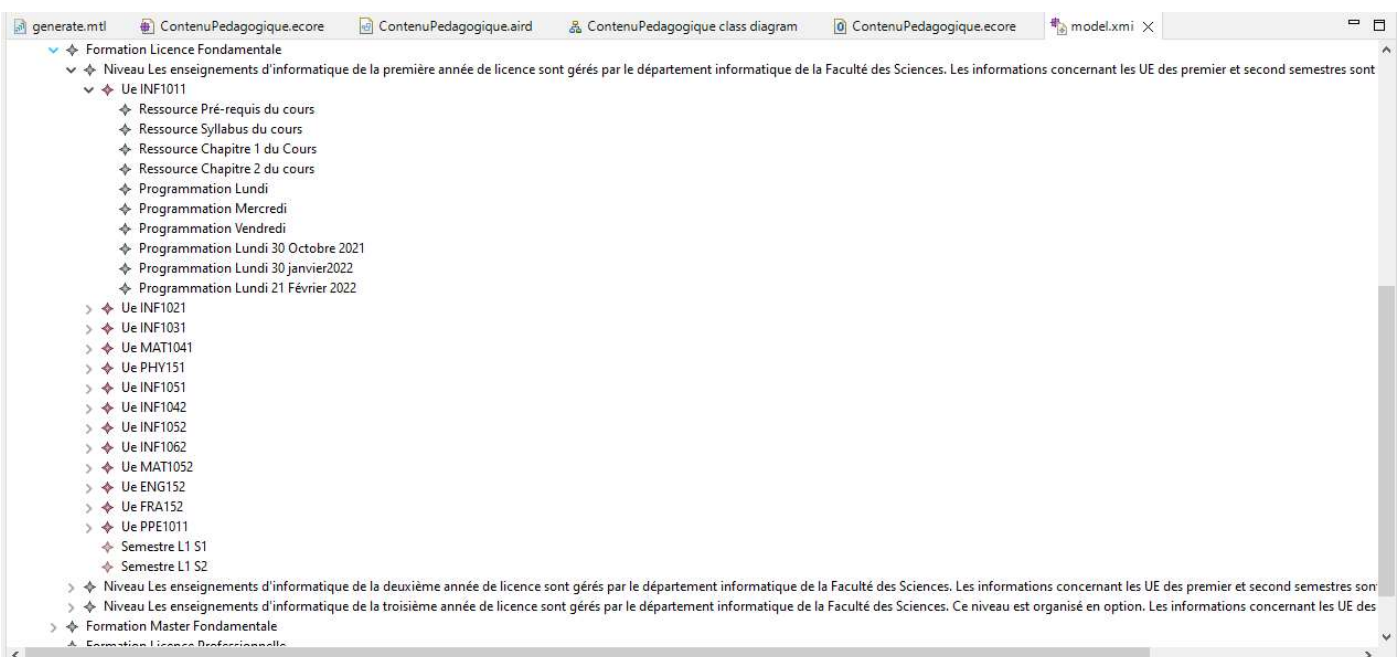
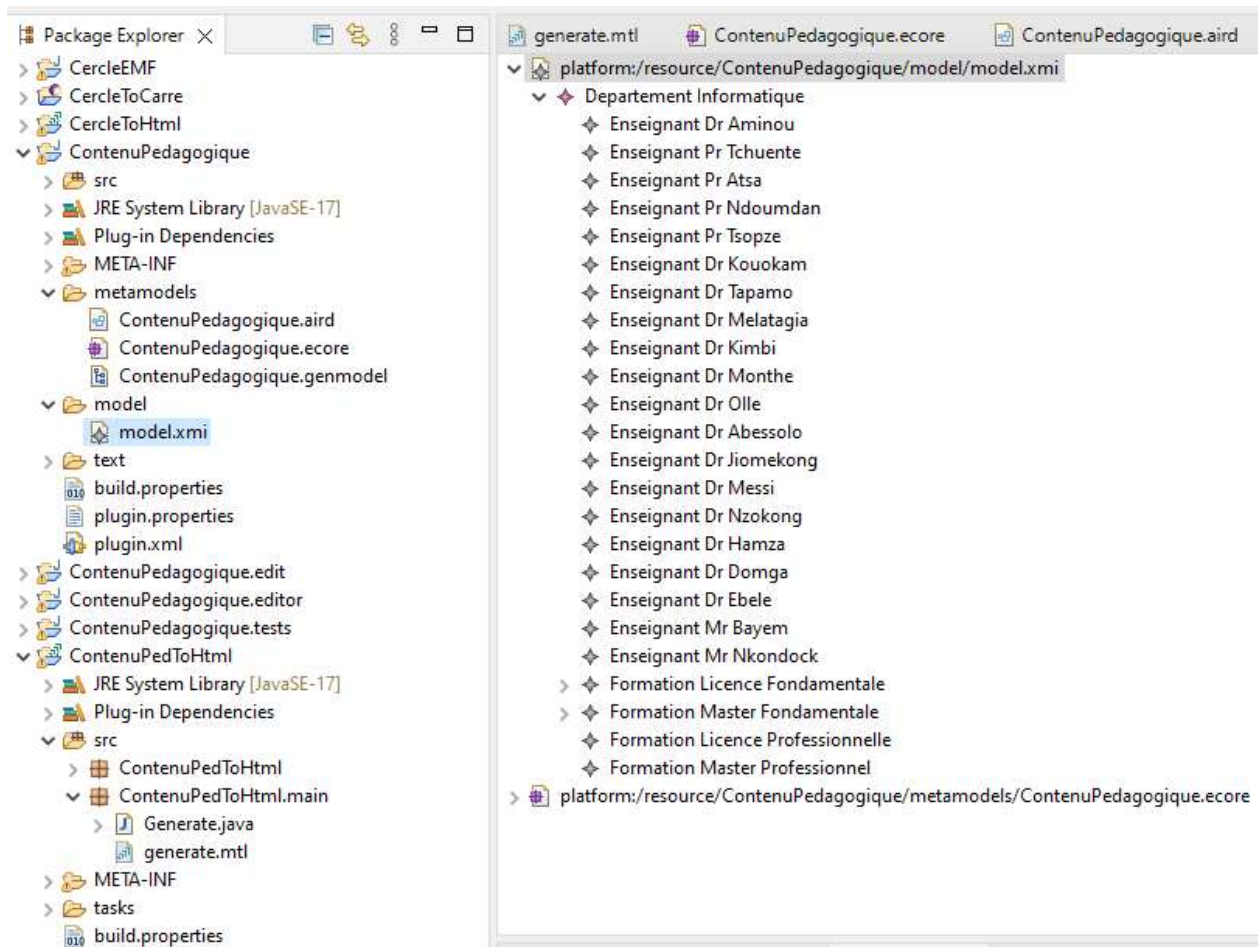
Une fois le modèle crée, on accède au fichier generate.mtl qui se trouve dans le dossier src du Projet ContenuPedToHtml, puis on fait un clic droit dessus et on accède à l'option « Run Configuration » du sous menu « Run as ». De là, on peut associer le fichier .xmi du modèle à utiliser, le dossier de destination du site à générer et la classe main du projet si elle n'est pas définit. Puis faire un clic gauche Apply puis Run.

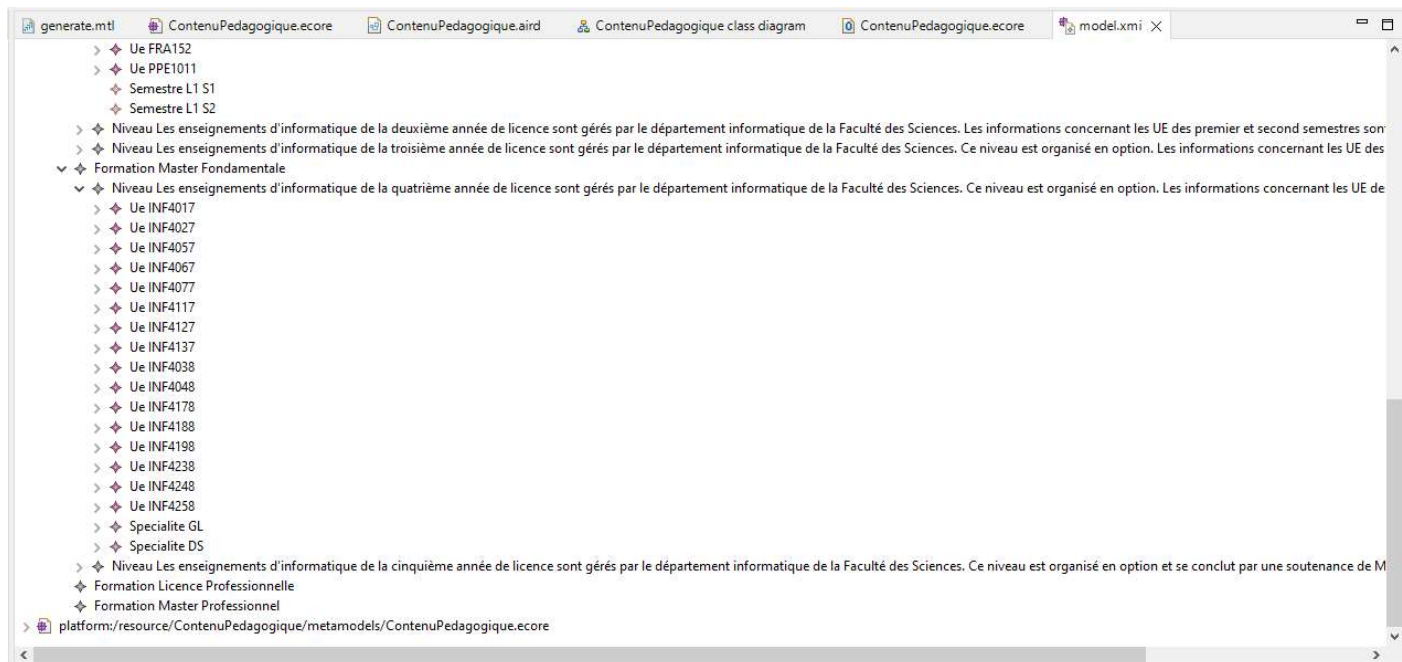


V. Exemple de résultat pour une instance du méta-modèle

1. Modèle

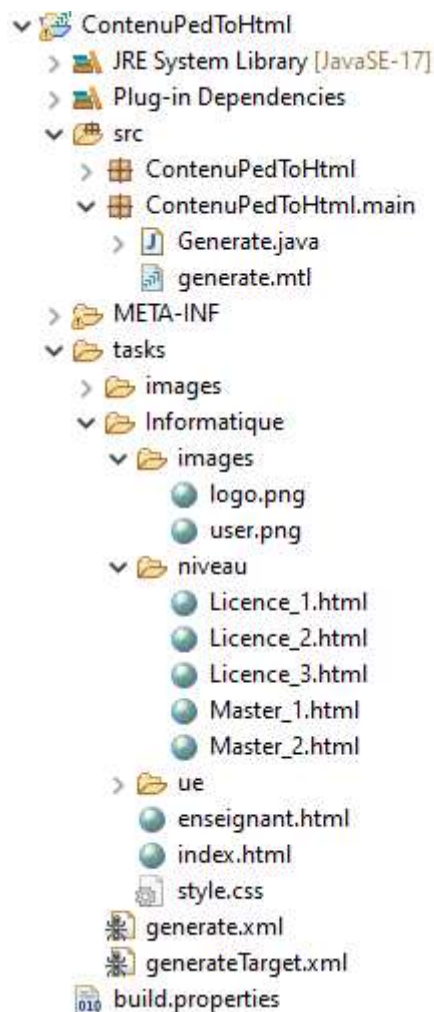
Nous avons créé un modèle à partir de notre méta-modèle tout en lançant le processus de validation pour s'assurer que le modèle était conforme. On a obtenu le modèle suivant :





2. Site Obtenu

Le processus de génération, nous a permis d'obtenir l'arborescence de fichier suivant :



L'ouverture du site nous offre le résultat suivant :

- Page d'accueil : présentation du département, des formations et leurs niveaux par le biais de la nav bar.



Université de Yaounde I
 Département d'Informatique

Connexion

Département Informatique



[Accueil](#)
[Licence Fondamentale](#)
[Master Fondamentale](#)
[Licence Professionnelle](#)
[Master Professionnel](#)
[Enseignants](#)

Description du Département et son programme

L'informatique est une discipline jeune aux multiples facettes, dont les métiers sont en constante évolution et en interaction avec les sciences connexes à l'informatique. Le parcours Informatique propose à la fois des unités d'enseignement fondamentales (mathématiques, algorithmique, langages de programmation issus des principaux paradigmes, outils formels, automates, cryptographie), ainsi que des unités d'enseignement ouvrant à une professionnalisation plus immédiate (systèmes informatiques, web, bases de données, programmation et conception orientées objet, réseaux, projets de développement). La relative pluridisciplinarité des premières années est une étape indispensable. Elle garantit une formation scientifique large, une ouverture d'esprit et une qualité du socle sur lequel l'étudiant fonde ses connaissances informatiques, selon deux approches : une fondamentale et l'autre plus technique.

Responsable du département

Dr Aminou

Contact(s) du département

Tel : (+237) 222 23 44 96 / (+237) 655 324 213
 PO Box : 812 Yaoundé-Cameroun

Conditions d'accès

Accès en 1ère année licence (L1) : Baccalauréat Scientifique (dans l'ordre de préférence suivant ; C, E, TIC, D avec une bonne note en mathématiques), GCE Advanced Level in two science subjects incluant les mathématiques et excluant la religion.

Accès en 2ème année (L2) : étudiants ayant validé au moins 45 crédits de L1.

Accès en 3ème année (L3) : étudiants ayant validé les 120 crédits de L1 et L2.

Accès en 4ème année (M1) : étudiants ayant une licence fondamentale en informatique.

Accès en 5ème année (M2) : étudiants ayant validé les 60 crédits en M1 avec 12 de moyenne et après une sélection.

Université de Yaounde I



Université de Yaounde I
 Département d'Informatique

Connexion

Département Informatique



[Accueil](#)
[Licence Fondamentale](#)
[Master Fondamentale](#)
[Licence Professionnelle](#)
[Master Professionnel](#)
[Enseignants](#)

Description du Département et son programme

L'informatique est une discipline jeune aux multiples facettes, dont les métiers sont en constante évolution et en interaction avec les sciences connexes à l'informatique. Le parcours Informatique propose à la fois des unités d'enseignement fondamentales (mathématiques, algorithmique, langages de programmation issus des principaux paradigmes, outils formels, automates, cryptographie), ainsi que des unités d'enseignement ouvrant à une professionnalisation plus immédiate (systèmes informatiques, web, bases de données, programmation et conception orientées objet, réseaux, projets de développement). La relative pluridisciplinarité des premières années est une étape indispensable. Elle garantit une formation scientifique large, une ouverture d'esprit et une qualité du socle sur lequel l'étudiant fonde ses connaissances informatiques, selon deux approches : une fondamentale et l'autre plus technique.

Responsable du département

Dr Aminou

Contact(s) du département

Tel : (+237) 222 23 44 96 / (+237) 655 324 213
 PO Box : 812 Yaoundé-Cameroun

Conditions d'accès

Accès en 1ère année licence (L1) : Baccalauréat Scientifique (dans l'ordre de préférence suivant ; C, E, TIC, D avec une bonne note en mathématiques), GCE Advanced Level in two science subjects incluant les mathématiques et excluant la religion.

Accès en 2ème année (L2) : étudiants ayant validé au moins 45 crédits de L1.


Accès en 3ème année (L3) : étudiants ayant validé les 120 crédits de L1 et L2.

Accès en 4ème année (M1) : étudiants ayant une licence fondamentale en informatique.

Accès en 5ème année (M2) : étudiants ayant validé les 60 crédits en M1 avec 12 de moyenne et après une sélection.


[Licence 1](#)
[Licence 2](#)
[Licence 3](#)

- Page de présentation des enseignants




Université de Yaounde I
 Département d'Informatique


Connexion




[Accueil](#)
[Licence Fondamentale](#)
[Master Fondamentale](#)
[Licence Professionnelle](#)
[Master Professionnel](#)
[Enseignants](#)



Dr Aminou (Chef)
 Chargé de Cours
 browndon@gmail.com
 Bureau : R201
 Responsable de 2 UE(s) :
 INF4258 - INF5119




Pr Tchunte
 Professeur
 browndon@gmail.com
 Bureau : R201
 Responsable de 4 UE(s) :
 INF1011 - INF2054 -
 FRA3045 - INF5009



Pr Atsa
 Professeur
 browndon@gmail.com
 Bureau : R201
 Responsable de 3 UE(s) :
 ENG152 - INF4027 -
 INF5029


Université de Yaounde I

- Page du niveau Licence 1



Université de Yaounde I
 Département d'Informatique

Connexion



[Accueil](#)
[Licence Fondamentale](#)
[Master Fondamentale](#)
[Licence Professionnelle](#)
[Master Professionnel](#)
[Enseignants](#)

Licence Fondamentale : Licence 1

[L1 S1](#)
[L1 S2](#)

Description du niveau

Les enseignements d'informatique de la première année de licence sont gérés par le département informatique de la Faculté des Sciences. Les informations concernant les UE des premier et second semestres sont accessibles via les liens L1 S1 et L1 S2. Ces enseignements concernent les étudiants de plusieurs portails et formations.

Responsable du niveau
 Dr Kouokam

Université de Yaounde I



Département Informatique



[Accueil](#) [Licence Fondamentale](#) [Master Fondamentale](#) [Licence Professionnelle](#) [Master Professionnel](#) [Enseignants](#)

Licence Fondamentale : Licence 1

[L1 S1](#) [L1 S2](#)

[INF1011](#) [INF1021](#) [INF1031](#) [MAT1041](#) [PHY151](#) [INF1051](#)

Description du niveau

Les enseignements d'informatique de la première année de licence sont gérés par le département informatique de la Faculté des Sciences. Les informations concernant les UE des premier et second semestres sont accessibles via les liens L1 S1 et L1 S2. Ces enseignements concernent les étudiants de plusieurs portails et formations.

Responsable du niveau

Dr Kouokam

- Page de description d'une UE du niveau Licence 1



Département Informatique



[Accueil](#) [Licence Fondamentale](#) [Master Fondamentale](#) [Licence Professionnelle](#) [Master Professionnel](#) [Enseignants](#)

Licence Fondamentale : Licence 1

[L1 S1](#) [L1 S2](#)

Semestre : L1 S1

INF1011 : Introduction à l'algorithmique

[Description](#) [Type](#) [Responsable](#) [Contenu](#) [Calendrier](#) [Ressources](#)

Objectifs de l'unité d'enseignement

Dans ce cours, une introduction à l'écriture des algorithmes et la description des données sont faites. Il présente une initiation à la résolution de problème simple au moyen d'algorithmes, en particulier les problèmes de recherche.

Nombre de Crédits de l'unité d'enseignement

6 crédits

Département Informatique



[Accueil](#) [Licence Fondamentale](#) [Master Fondamentale](#) [Licence Professionnelle](#) [Master Professionnel](#) [Enseignants](#)

Licence Fondamentale : Licence 1

[L1 S1](#) [L1 S2](#)

Semestre : L1 S1

INF1011 : Introduction à l'algorithmique

[Description](#) [Type](#) [Responsable](#) [Contenu](#) [Calendrier](#) [Ressources](#)

Cours Magistral

Jour : **Lundi**
Heure: **07h - 09h55**
Salle : **Amphi 315**

Travaux Pratiques

Jour : **Mercredi**
Heure: **10h05 - 12h55**
Salle : **Amphi 315**

Travaux Dirigés

Jour : **Vendredi**
Heure: **12h00**
Salle : **NB3**

Contrôle Continu

Jour : **Lundi 30 Octobre 2021**
Heure: **07h**
Salle : **Amphi 315**

Examen

Jour : **Lundi 30 janvier 2022**
Heure: **07h**
Salle : **Amphi 315**

Rattrapage

Jour : **Lundi 21 Février 2022**
Heure: **07h**
Salle : **Amphi 315**

Université de Yaoundé I



Université de Yaoundé I
Département d'Informatique

[Connexion](#)

Département Informatique



[Accueil](#) [Licence Fondamentale](#) [Master Fondamentale](#) [Licence Professionnelle](#) [Master Professionnel](#) [Enseignants](#)

Licence Fondamentale : Licence 1

[L1 S1](#) [L1 S2](#)

Semestre : L1 S1

INF1011 : Introduction à l'algorithmique

[Description](#) [Type](#) [Responsable](#) [Contenu](#) [Calendrier](#) [Ressources](#)

Fichiers de l'unité d'enseignement

[Pré-requis du cours](#)

[Syllabus du cours](#)

[Chapitre 1 du Cours](#)

[Chapitre 2 du cours](#)

Université de Yaoundé I

- Page du niveau Master 1



Université de Yaounde I
Département d'Informatique

[Connexion](#)

Département
Informatique



[Accueil](#) [Licence Fondamentale](#) [Master Fondamentale](#) [Licence Professionnelle](#) [Master Professionnel](#) [Enseignants](#)

Master Fondamentale : Master 1

[M1 S1 GL](#) [M1 S2 GL](#) [M1 S1 DS](#) [M1 S2 DS](#)

Description du niveau

Les enseignements d'informatique de la quatrième année de licence sont gérés par le département informatique de la Faculté des Sciences. Ce niveau est organisé en option. Les informations concernant les UE des premier et second semestres sont accessibles via les liens M1 Option S1 et M1 Option S2. Ces enseignements concernent les étudiants de plusieurs portails et formations.

Responsable du niveau

Dr Monthe

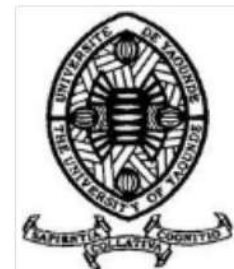
Université de Yaounde I



Université de Yaounde I
Département d'Informatique

[Connexion](#)

Département
Informatique



[Accueil](#) [Licence Fondamentale](#) [Master Fondamentale](#) [Licence Professionnelle](#) [Master Professionnel](#) [Enseignants](#)

Master Fondamentale : Master 1

[M1 S1 GL](#) [M1 S2 GL](#) [M1 S1 DS](#) [M1 S2 DS](#)

[INF4017](#) [INF4027](#) [INF4117](#) [INF4127](#) [INF4137](#)

Description du niveau


Les enseignements d'informatique de la quatrième année de licence sont gérés par le département informatique de la Faculté des Sciences. Ce niveau est organisé en option. Les informations concernant les UE des premier et second semestres sont accessibles via les liens M1 Option S1 et M1 Option S2. Ces enseignements concernent les étudiants de plusieurs portails et formations.

Responsable du niveau

Dr Monthe



Université de Yaounde I

- Page de description d'une UE fondamentale du niveau Master 1



Université de Yaounde I
 Département d'Informatique

Connexion

[Accueil](#)
[Licence Fondamentale](#)
[Master Fondamentale](#)
[Licence Professionnelle](#)
[Master Professionnel](#)
[Enseignants](#)

Master Fondamentale : Master 1

[M1 S1 GL](#)
[M1 S2 GL](#)
[M1 S1 DS](#)
[M1 S2 DS](#)

Semestre : M1 S1

INF4017 : Complexité et algorithmique avancée

[Description](#)
[Type](#)
[Responsable](#)
[Contenu](#)
[Calendrier](#)
[Ressources](#)

Objectifs de l'unité d'enseignement

Aucun objectif associé

Nombre de Crédits de l'unité d'enseignement

0 crédits



Université de Yaounde I

- Page de description d'une UE spécialisé du niveau Master 1



Université de Yaounde I
 Département d'Informatique

Connexion

[Accueil](#)
[Licence Fondamentale](#)
[Master Fondamentale](#)
[Licence Professionnelle](#)
[Master Professionnel](#)
[Enseignants](#)

Master Fondamentale : Master 1

[M1 S1 GL](#)
[M1 S2 GL](#)
[M1 S1 DS](#)
[M1 S2 DS](#)

Spécialité : Génie Logiciel et Système d'Informations

Semestre : M1 S1 GL

INF4057 : Architectures Logicielles

[Description](#)
[Type](#)
[Responsable](#)
[Contenu](#)
[Calendrier](#)
[Ressources](#)

Objectifs de l'unité d'enseignement

Aucun objectif associé

Nombre de Crédits de l'unité d'enseignement

0 crédits

Université de Yaounde I