

Projet Montauban - Django

Diagramme de classes (voir png envoyé par mail)

Dictionnaire de données

table/class	Description	Clé	type	taille
Clients				
id_clients	Valeur numérique identifiant de manière unique les clients	pk = (primary key)	IntegerField	AI
nom_clients	Chaîne de caractères pour identifier le nom du client		Charfield	100
prenom_clients	Chaîne de caractères pour identifier le prénom du client		Charfield	100
mail_clients	Chaîne de caractères pour identifier le mail du client		EmailField	100
telephone_clients	Valeur numérique identifiant le numéro de téléphone des clients		IntegerField	10
photo_clients	Chaîne de caractères pour identifier l'url pour la photo du client		ImageField	
date_naissance_clients	Valeur numérique identifiant la date de naissance des clients		DateField	
adresse_clients	Chaîne de caractères pour identifier l'adresse du client		CharField	255
numero_adresse_clients	Valeur numérique identifiant le numéro de porte des clients		IntegerField	
adresse_rue_clients	Chaîne de caractères pour identifier la rue/lieu du client		CharField	255
ville_clients	Chaîne de caractères pour identifier la ville du client		CharField	100
departement_clients	Chaîne de caractères pour identifier le departement du client		CharField	100
code_postal_clients	Valeur numérique identifiant le code postal des clients		IntegerField	5

pays_clients	Chaîne de caractères pour identifier le departement du client		CharField	100
Projets				
id_projets	Valeur numérique identifiant de manière unique les projets	pk = (primary key)	IntegerField	AI
projet_status_projets	Valeur numérique identifiant de manière unique le statut du projet	Fk = (ForeignKey)	IntegerField	AI
projet_clients	Valeur numérique identifiant de manière unique le client associé au projet	Fk = (ForeignKey)	IntegerField	AI
date_debut_projets	Valeur numérique identifiant la date de démarrage d'un projet		DateField	
date_fin_provisoire	Valeur numérique identifiant la date de fin d'un projet		DateField	
description_projets	Chaîne de caractères pour décrire le projet		CharField	255
nom_projets	Chaîne de caractères pour donner un nom au projet		CharField	100
adresse_projets	Chaîne de caractères pour identifier l'adresse du projet		CharField	255
ville_projets	Chaîne de caractères pour identifier la ville du projet		CharField	100
departement_projets	Chaîne de caractères pour identifier le departement du projet		CharField	100
pays_projets	Chaîne de caractères pour identifier le pays ou le projet aura lieu		CharField	100

Règles de gestion

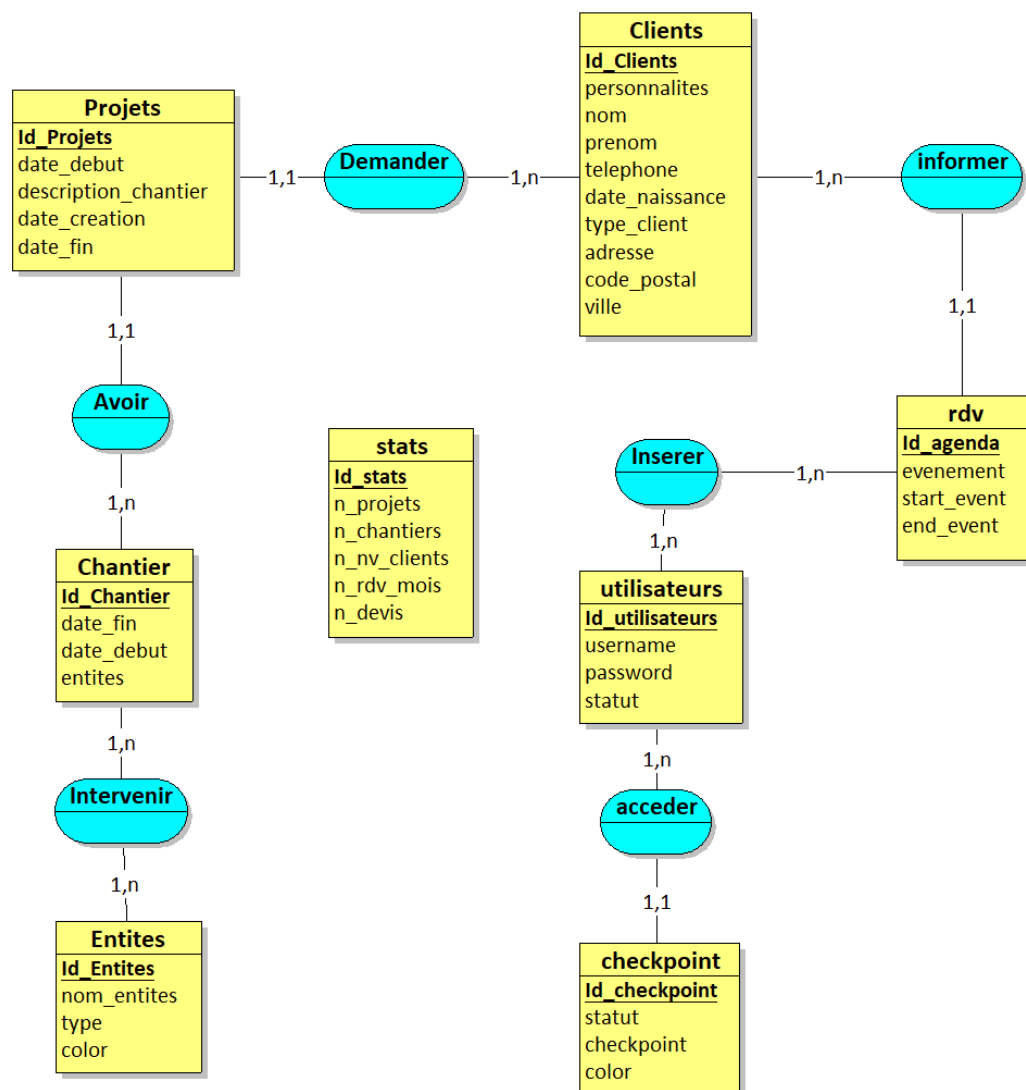
Pour le projet en cours les règles de gestion sont les suivantes:

Les clients peuvent demander plusieurs projets (1 -- n) mais le projet appartient à un seul client (1 -- 1).

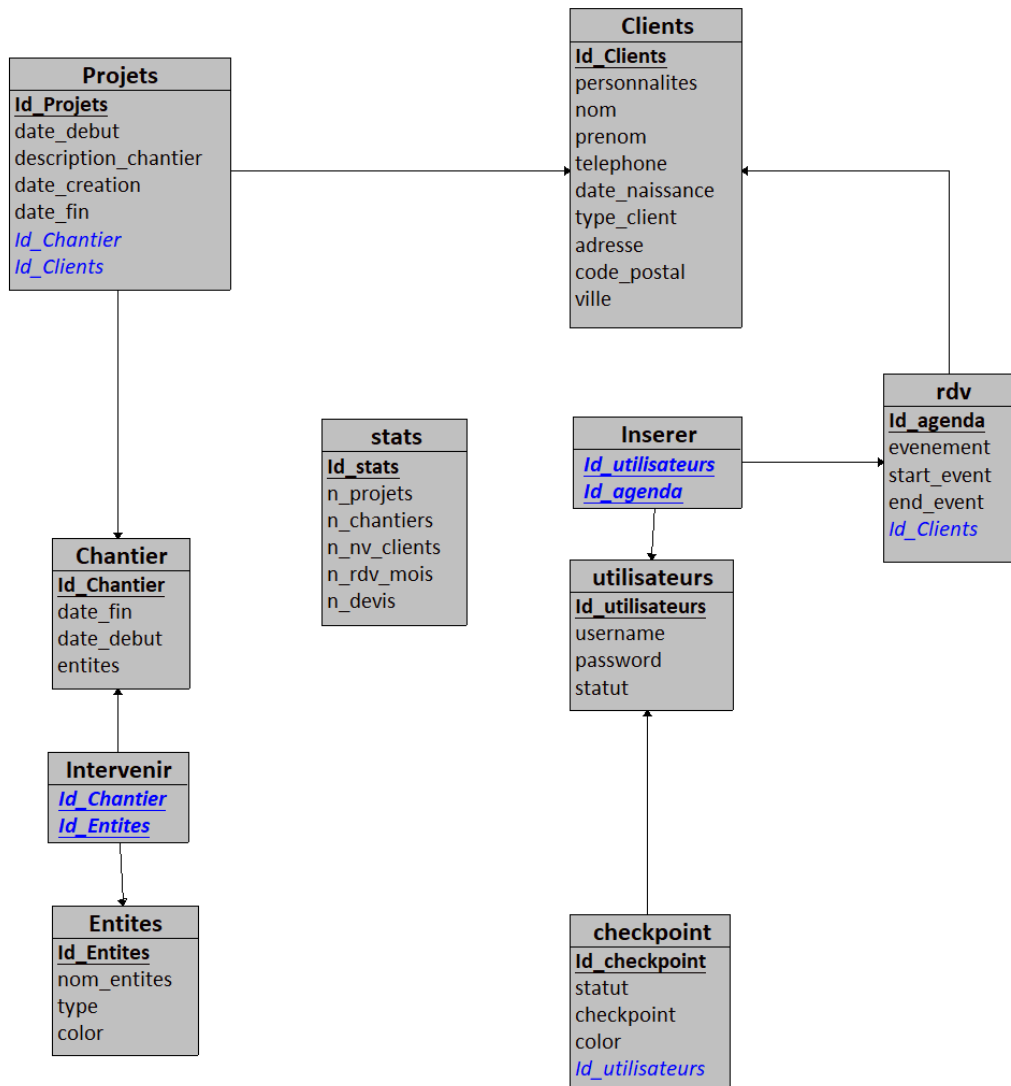
Dans 1 projet peut avoir plusieurs chantiers (1 -- n) mais les chantiers on un seul projet (1 -- 1)

Une entité peut intervenir sur plusieurs chantiers (1 -- n) et dans le chantier peut avoir plusieurs entités au même temps (1 -- n)

MCD



MLD



Les différentes phases de programmation du projet en Django

Avant tout, pourquoi j'ai choisi Django pour développer l'application web pour Montauban?

J'ai choisi Django parce que c'est un framework connu et reconnu avec multiples avantages. Django intervient pour la création d'applications web, et sa conception est caractéristique de ce genre de framework : il s'articule complètement autour du pattern logiciel **MVC** (modèle/vue/contrôleur).

Django prend clairement le parti d'encourager très fortement le programmeur à écrire le moins de code possible, le plus réutilisable possible. Ces objectifs se traduisent concrètement par :

- des helpers et autres outils pratiques pour ne pas se répéter (**DRY** : Don't Repeat Yourself) ;

- des possibilités de modularisation et de couplage faible du code étendu.

Le framework a d'autres points forts telles que:

- Un moteur de template très puissant implémentant un concept d'héritage de templates, à la syntaxe simple.
- Un ORM (Object Relational Mapper) défiant toute concurrence en matière de simplicité et présentant des performances très honorables
- Une gestion du routage d'URL (contrôleur frontal des applications) élégante, fondée sur les patterns d'expressions régulières nommés de Python

Les différentes phases de la création d'un projet en Django sont:

1. Création d'un dossier pour le projet
2. Création d'un environnement virtuel avec la commande "pipenv shell" (pipenv est installé une première fois avec la commande 'pip install pipenv').
3. Création d'un projet avec la commande "pipenv startproject <nom_du_projet>"
4. Accéder au projet avec la commande "cd <nom_du_projet>"
5. A l'intérieur du projet on trouvera le fichier manage.py et settings
6. Création d'une "app" avec la commande "python manage.py startapp <nom_app>"
7. Modifier dans les settings l'accès à la base de données (dans mon cas pour le projet Montauban j'ai choisi la BDD MySQL)
8. Créer l'accès admin du projet avec la commande "python manage.py createsuperuser"
9. Créer les différentes classes en "models.py" (qui correspondent aux tables de notre BDD)
10. Créer sur les "views.py" les fonctions pour rendre les pages html.
11. Ajouter sur "urls.py" les chemins d'accès de Django dans les "urlpatterns"
12. Créer nos tables en BDD avec la commande "python manage.py makemigrations" et après la commande "python manage.py migrate"

Et ça c'est la explication simple de l'utilisation de Django pour créer une application web.