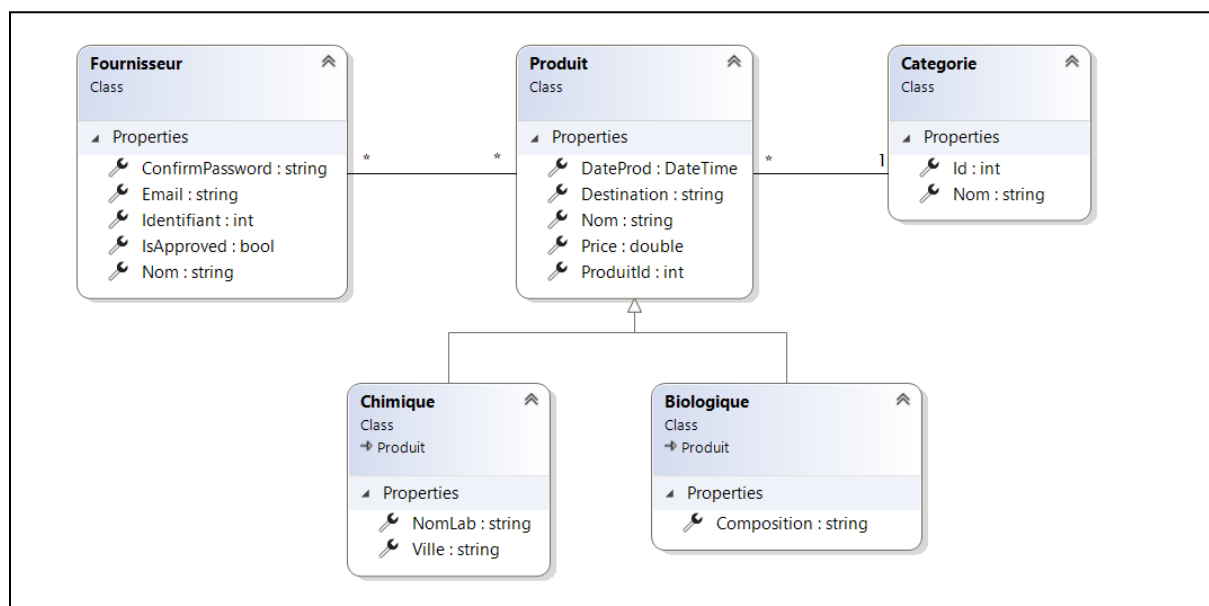
 <p>esprit Se former autrement HONORIS UNITED UNIVERSITIES</p>	<h2 style="margin: 0;">EXAMEN</h2> <p>Semestre : <input type="checkbox"/> 1 <input checked="" type="checkbox"/> 2</p> <p>Session : <input checked="" type="checkbox"/> Principale <input type="checkbox"/> Rattrapage</p>		
<p>Module : Architecture des systèmes d'information I (.Net)</p> <p>Enseignante : Abir Lahbecha</p> <p>Classes : 3ALINFO1-3ALINFO4</p> <p>Documents autorisés : <input checked="" type="checkbox"/> OUI <input type="checkbox"/> NON Nombre de pages : 4 pages</p> <p>Date : 15/04/2023 Heure : 08h30 Durée : 1h30m</p>			
<p>ETUDIANT(e)</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 60%; border: none;"> N° Carte : Nom et Prénom : </td> <td style="width: 40%; border: none;"> Classe : Salle : </td> </tr> </table>		N° Carte : Nom et Prénom :	Classe : Salle :
N° Carte : Nom et Prénom :	Classe : Salle :		

NB : -Toute carte Wifi active est une tentative de fraude
-Respectez l'architecture vue en cours et les patrons de conception étudiés.

Enoncé du problème

On désire implémenter une application Web de gestion des Produits, à l'aide du Framework .Net6 en suivant le diagramme de classes ci-dessous:



Partie I: Entity-Framework (10 pts)

1. Implémenter le diagramme de classes ci-dessus. (3 pts)

2. En utilisant les annotations:

dans la classe **Fournisseur**:

- a. La propriété **Identifiant** doit être une clé primaire.(0.5 pt).
- b. La propriété **Nom** doit avoir une longueur minimale de 3 caractères et maximale de 12 caractères (0.5 pt).

dans la classe **Produit**:

- c. La propriété **DateProd** doit être une date valide avec affichage d'un message d'erreur dans le cas échéant (0.5 pt).
- d. Définir une propriété **CategorieFk** qui doit être la clé étrangère qui correspond à la relation avec la classe **Categorie** (0.5 pt).

3. En utilisant **FluentAPI**:

- a. Configurer la relation Many To Many entre **Fournisseur** et **Produit** en précisant **Facture** comme étant le nom de la table associative.(1 pt)
- b. Configurer la colonne qui représente le Discriminator de la hiérarchie d'héritage par défaut (TPH) entre **Produit**, **Chimique** et **Biologique** afin qu'elle soit nommée **TypeProduit** de type **char** et qui prend la valeur **C** si le type du **Produit** est **Chimique**, la valeur **B** si le type de **Produit** est **Biologique** et la valeur **P** sinon. (1.5 pt)

4. Implémenter une pré-convention qui mappe toutes les propriétés de type **String** dans des colonnes de longueur maximale 50 caractères. (1pt)

5. Générer la base de données nommée sous la forme **NomPrenomProduit**. (1.5pts)

Partie II: Services (4 pts)

Dans les classes de services spécifiques, créer:

6. une méthode qui permet de retourner les fournisseurs d'une catégorie passée en paramètre.(1pt)

7. Une méthode qui permet de retourner les cinq premiers produits chimiques qui ont un prix supérieur au prix donné. (1.5pts)

8. Une méthode qui permet de retourner la moyenne des prix des produits biologiques d'une catégorie donnée. (1.5 pts)

Partie III: ASP MVC (6 pts)

9. Alimenter la BDD par **deux banques** comme le montre la figure suivante: (0.5 pt)

	Id	Nom
	1	alimentaire
	2	Cosmetique

10. Réaliser une vue qui permet de créer un **Produit** comme le montre la figure suivante, tout en choisissant la **Categorie** à partir d'une liste déroulante. (2.5 pts)

Create

Produit

Destination

Nom

DateProd

Price

CategorieFK

alimentaire

Create

[Back to List](#)

11. Rediriger vers une vue **Index** qui liste l'ensemble des produits.(1.5 pt)

Index						
Create New						
ProduitId	Destination	Nom	DateProd	Price	CategorieFK	
1	Italie	Produit1	13/04/2023	15	alimentaire	Edit Details Delete

12. Ajouter un bloc de recherche à la vue **Index**, qui permet de filtrer les produits par **Nom**.
(1.5pt)

Recherche

Saisir un nom :

ProduitId	Destination	Nom	DateProd	Price	CategorieFK	
1	Italie	Produit1	13/04/2023	15	alimentaire	Edit Details Delete

Bon travail 😊