
IID1-2023/2024

Rapport du projet

gestion de centre de formation

système d'information

Pr.ROCHD YASSER

réalisé par:

- *Wijdane Taftaf
- *Selma Ezaakouni
- *Salah Eddine Adaoui
- *Nassima Rhannouch

Table des matières:

Problématique :
Dictionnaire de données :
Modèle Conceptuel de données:
Modèle Logique de données:
Schéma relationnel :
Modèle Conceptuel de Traitement :
Règles de gestion:
Vue générale de l'application:
les requêtes utiliser:	
Conclusion :

Problématique :

Nous sommes chargés de développer une application de gestion destinée à un centre de formation. Cette application couvrira divers domaines tels que les cours de langues, les cours de soutien, et les compétences en soft skills.

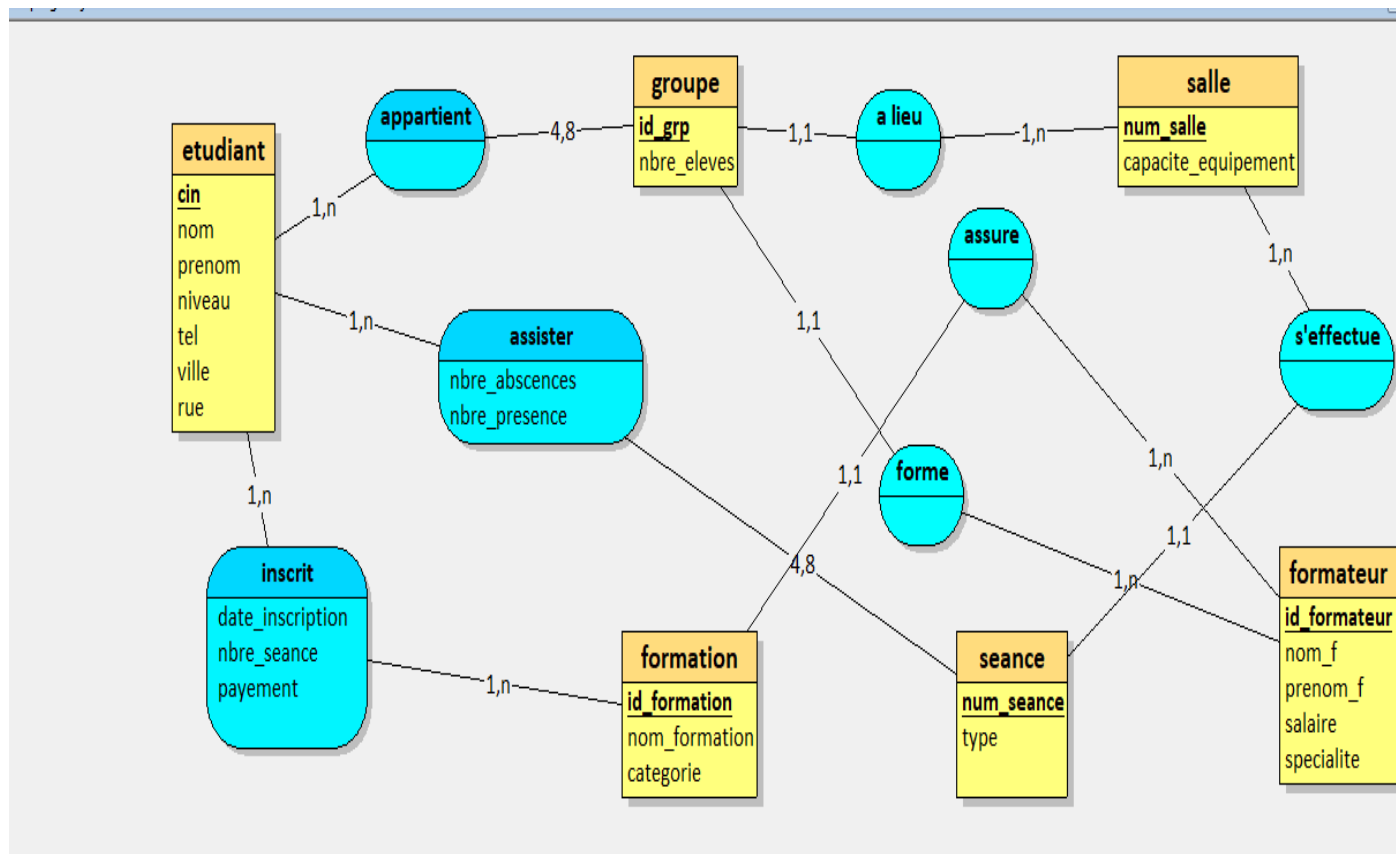
L'objectif principal est de fournir une plateforme complète permettant la gestion efficace des activités au sein du centre de formation. Cette application devra prendre en compte les besoins spécifiques liés aux cours de langues, aux sessions de soutien, ainsi qu'aux formations axées sur les compétences professionnelles.

Les fonctionnalités prévues pour l'application incluront, entre autres, la gestion des inscriptions des étudiants, le suivi des présences et des absences, la planification des cours, la gestion des formateurs et le suivi des paiements. Nous visons à concevoir une interface conviviale qui facilite l'expérience tant pour les étudiants que pour le personnel administratif du centre de formation.

Dictionnaire de données:

Attribut	Type	Longueur	Contraintes
cin	AUTO_INCREMENT	0	✓
date	DATE	0	✓
email	CHAR	0	✓
id_formateur	AUTO_INCREMENT	0	✓
nbre_eleves	BIGINT	0	✓
niveau	CHAR	0	✓
nom	CHAR	0	✓
nom_f	CHAR	0	✓
nom_formation	CHAR	0	✓
num_seance	BIGINT	0	✓
prenom	CHAR	0	✓
prenom_f	CHAR	0	✓
rue	BIGINT	0	✓
salaire	BIGINT	0	✓
specialite	CHAR	0	✓
tel	BIGINT	0	✓
type	CHAR	0	✓
ville	CHAR	0	✓

Modèle Conceptuel de données:



Modèle Logique de données:

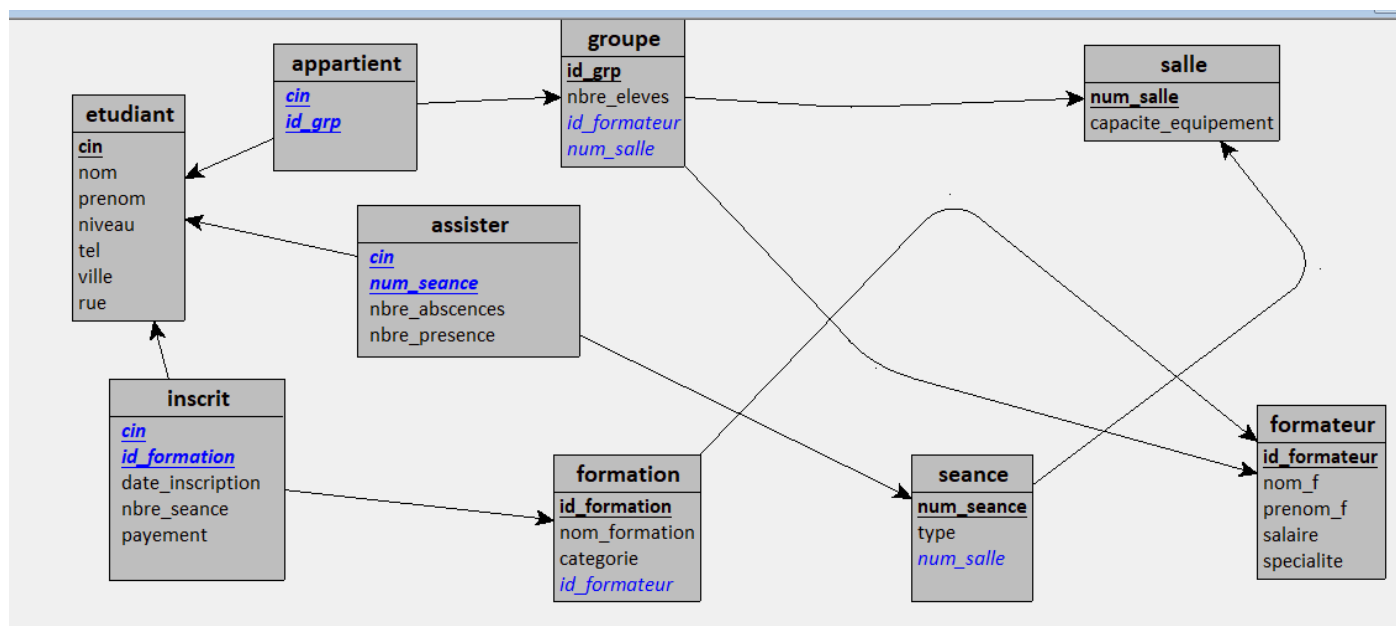
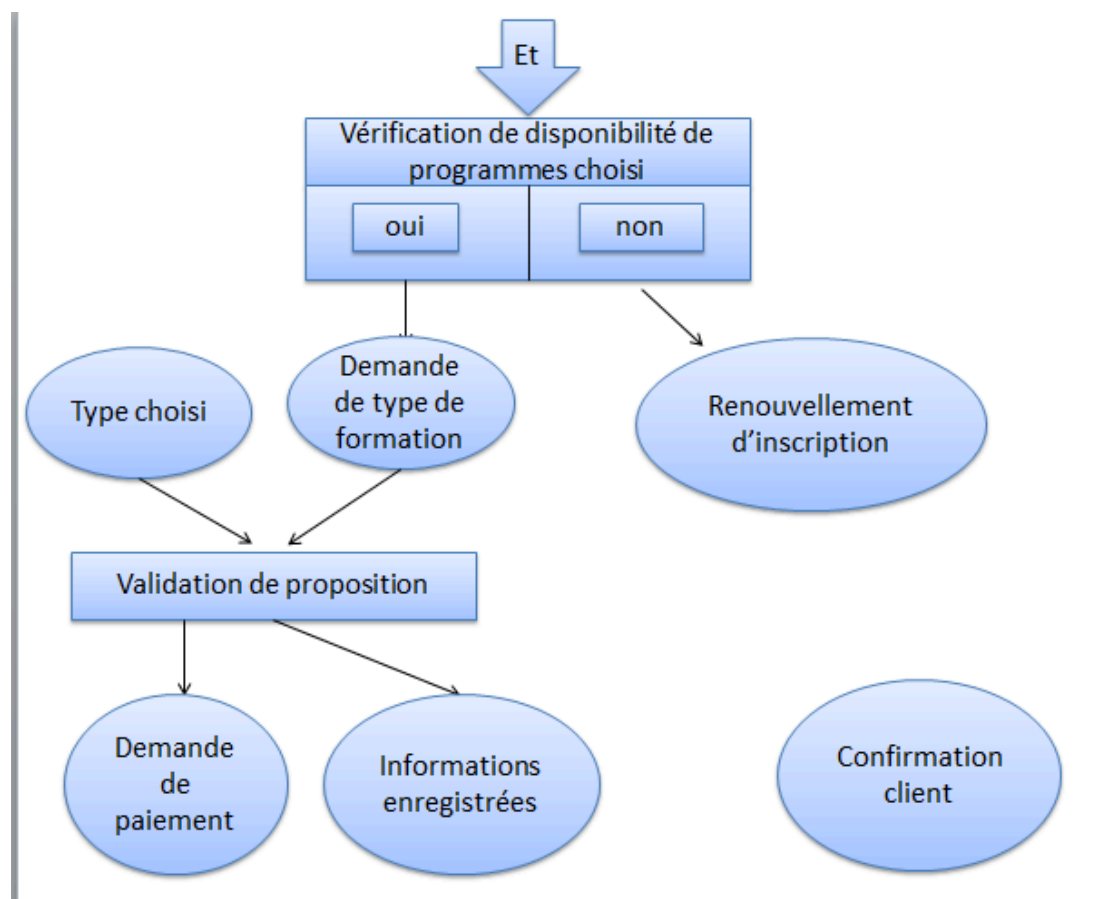
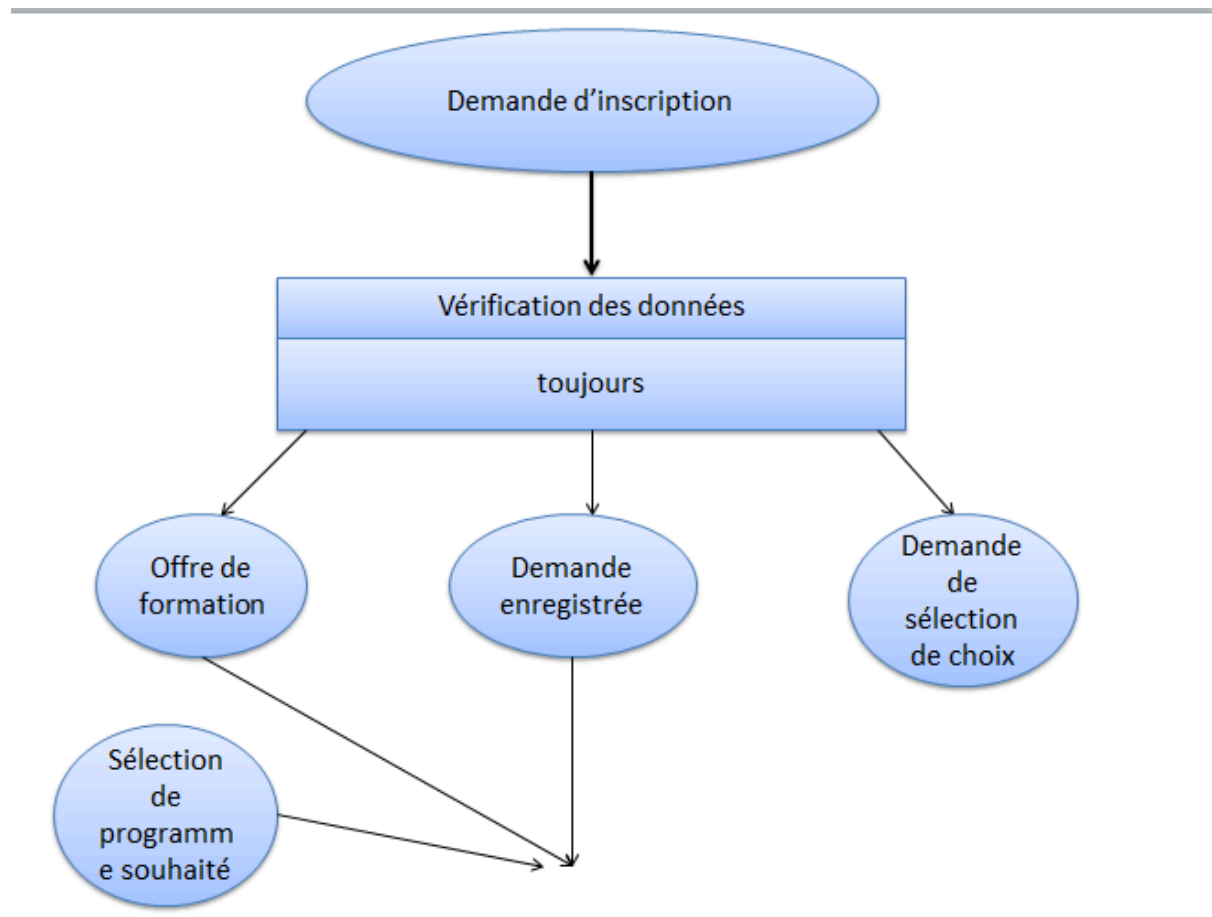
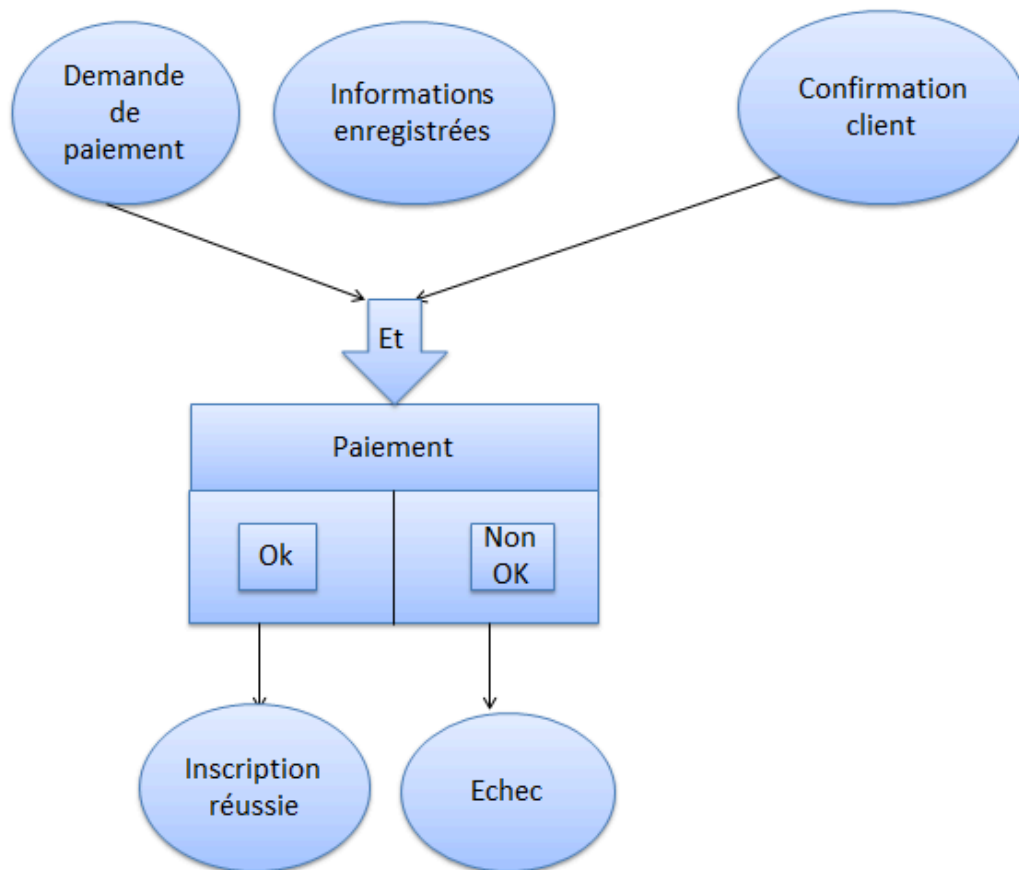


Schéma relationnel:

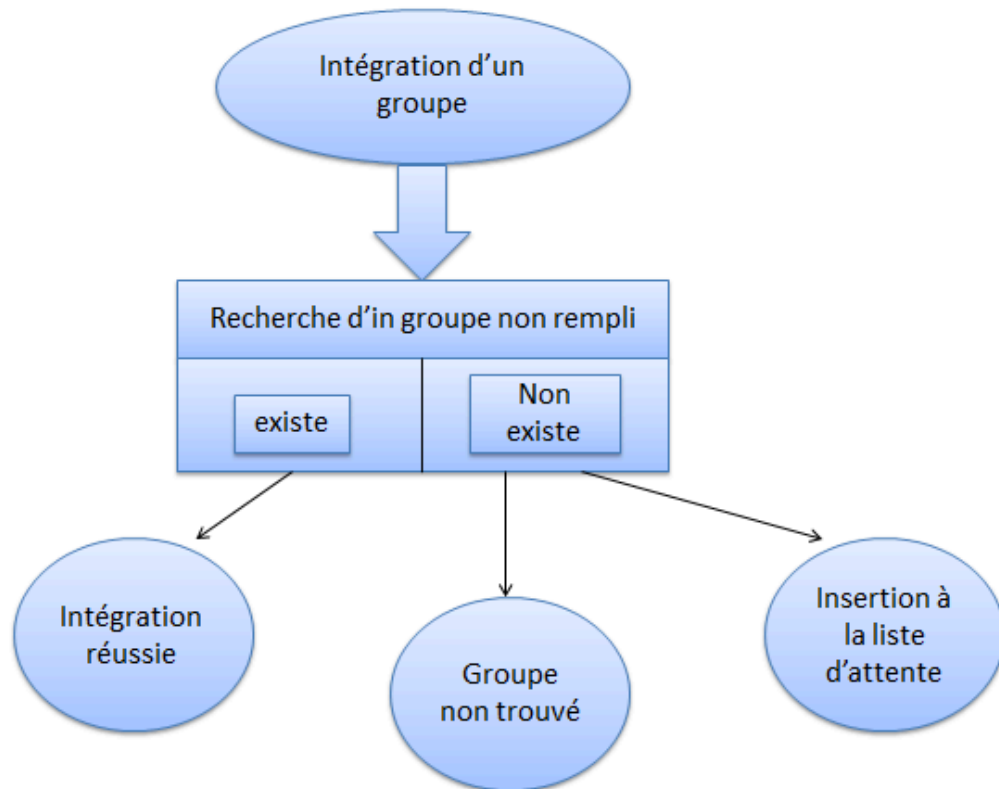
```
etudiant = (cin VARCHAR(8), nom CHAR(25), prenom VARCHAR(25), niveau VARCHAR(25), tel INT, ville VARCHAR(50), rue VARCHAR(25));
salle = (num_salle INT, capacite_equipement INT);
formateur = (id_formateur VARCHAR(25), nom_f VARCHAR(25), prenom_f VARCHAR(25), salaire INT, specialite VARCHAR(25));
formation = (id_formation VARCHAR(25), nom_formation CHAR(25), categorie CHAR(25), #id_formateur);
seance = (num_seance VARCHAR(25), type CHAR(25), #num_salle);
groupe = (id_grp VARCHAR(25), nbre_eleves INT, #id_formateur, #num_salle);
appartient = (#cin, #id_grp);
assister = (#cin, #num_seance, nbre_absences INT, nbre_presence INT);
inscrit = (#cin, #id_formation, date_inscription DATE, nbre_seance INT, paiement INT);
```

Modèle Conceptuel de Traitement :





2ème MCT:



Règles de Gestion:

- 1) pour les cours de soutien, le paiement est pour la séance avec un prix de 150 DHS
- 2) pour une séance 'test' , le client doit payer 200 DHS
- 3) les étudiants ayant choisi cours de soutien du collège au bac , se focaliseront sur les matières scientifiques. à l'exception du bac 2 et 1er bac.(philo et ang pour bac 2 et ara,ei,hg,fr pour bac1)
- 4) Le maximum de nombre de places dans une classe est 8, le minimum est 4
- 5) 2 Types de séances : en ligne et présentiel (avec même prix)

- 6) pour les langues , soft skills. le client doit s'inscrire obligatoirement à 8 séances pour 1200 DHS
- 7) Pour une absence, elle ne sera pas compté. mais à partir de la 2ème , elle sera prise en considération
- 8) les clients inscrits sont insérés automatiquement dans une liste d'attente en cas d'absence de groupes libres
- 9) les clients sont regroupés en fonction de leurs niveaux et programme choisis
- 10) pour chaque séance, une liste d'absence est effectué
- 11) le remboursement n'est pas accepté
- 12) l'emploi du temps sera le suivant:

-dimanche: cours de soutien

- samedi: langue

-mercredi: soft skills

-lundi: à distance

Vue générale de l'application:

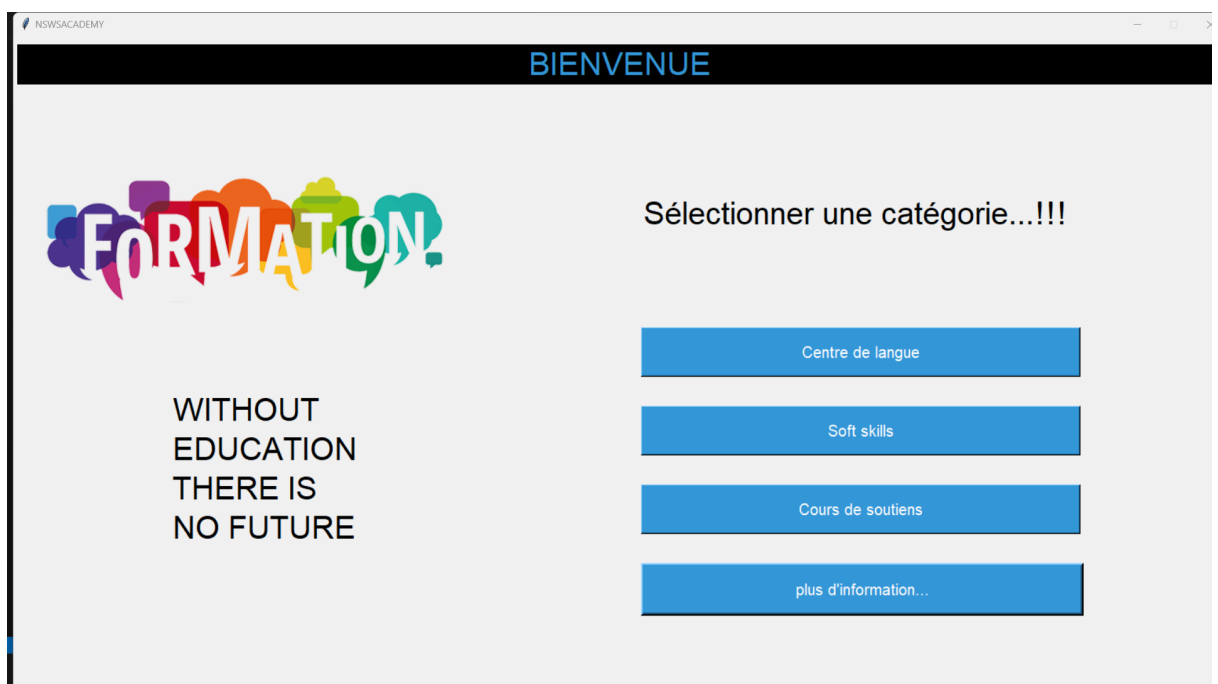
Gestion des Utilisateurs :

- Enregistrement et gestion des étudiants, des formateurs.



Gestion des Cours et Formations :

- Création et planification de cours pour différentes spécialités (langues, soutien scolaire, soft skills, etc.)



Gestion des Groupes :

- Création et gestion des groupes d'étudiants en fonction des cours et des niveaux.
- Affectation de formateurs à des groupes spécifiques.



Gestion des Séances et Présences :

- Programmation et suivi des séances de cours.
- Enregistrement des présences et gestion des absences.

Gestion des Paiements :

- Suivi des paiements des étudiants

les requêtes utiliser :

les étudiants de soft skills:

```
select e.nom,e.prenom,f.categorie as choix from etudiant e,inscrit i,formation f where e.CIN=i.CIN and i.id_formation=f.id_formation and f.categorie='soft skills'
```

les étudiants de langue:

```
select e.nom,e.prenom,f.categorie as choix from etudiant e,inscrit i,formation f where e.CIN=i.CIN and i.id_formation=f.id_formation and f.categorie='langue'
```

les étudiants de soutien scolaire:

```
select e.nom,e.prenom,f.categorie as choix from etudiant e,inscrit i,formation f where e.CIN=i.CIN and i.id_formation=f.id_formation and f.categorie='soutien scolaire'
```

les listes de formations de langue:

```
select nom_formation,categorie from formation where categorie='langue'
```

les listes de formation de soutien scolaire:

```
select nom_formation,categorie from formation where categorie='soutien scolaire'
```

les listes de formation de soft skills:

```
select nom_formation,categorie from formation where categorie='soft skills'
```

les listes de groupes de langue:

```
select distinct a.id_grp,f.categorie from appartient a,inscrit i,formation f where a.cin=i.cin and f.id_formation=i.id_formation and f.categorie='langue'
```

les listes de groupes de soutien scolaire:

```
select distinct a.id_grp,f.categorie from appartient a,inscrit i,formation f where a.cin=i.cin and f.id_formateur=i.id_formateur and f.categorie='soutien scolaire'
```

les listes de groupes de soft skills:

```
select distinct a.id_grp,f.categorie from appartient a,inscrit i,formation f where a.cin=i.cin and f.id_formateur=i.id_formateur and f.categorie='soft skills'
```

la liste des étudiants en attend dans 'langue':

```
SELECT e.nom,e.prenom FROM etudiant e,inscrit i,formation f WHERE NOT EXISTS ( SELECT 1 FROM appartient WHERE appartient.CIN=e.CIN) and e.CIN=i.CIN and i.id_formation=f.id_formation and f.categorie='langue'
```

la liste des étudiants en attend dans 'soft skills':

```
SELECT e.nom,e.prenom FROM etudiant e,inscrit i,formation f WHERE NOT EXISTS ( SELECT 1 FROM appartient WHERE appartient.CIN=e.CIN) and e.CIN=i.CIN and i.id_formation=f.id_formation and f.categorie='soft skils'
```

la liste des étudiants en attend dans 'soutien scolaire:

```
SELECT e.nom,e.prenom FROM etudiant e,inscrit i,formation f WHERE NOT EXISTS ( SELECT 1 FROM appartient WHERE appartient.CIN=e.CIN) and e.CIN=i.CIN and i.id_formation=f.id_formation and f.categorie='soutien scolaire'
```

la liste des formateurs de soutien scolaire:

```
select * from formateur where specialite in ('math','pc','svt','anglais','français','hg','ei','philo','arabe')
```

la liste des formateurs de langue:

```
select * from formateur where specialite in ('anglais_L','français_L','chinois')
```

la listes des formateurs de soft skills:

```
select * from formateur where specialite in ('e_commerce','dev_web','dv_personnel','trading')
```

la liste des groupes avec ses catégories:

```
select distinct a.id_grp,f.categorie from appartient a,inscrit i,formation f where a.cin=i.cin and f.id_formation=i.id_formation and f.categorie='nom de formation'
```

la liste des étudiants avec leurs séances restantes:

```
SELECT inscrit.CIN AS CIN_etudiant,inscrit.id_formation,inscrit.nbr_seance,assister.nbr_presence,CASE WHEN COALESCE(assister.nbr_absc, 0) = 0 THEN inscrit.nbr_seance - COALESCE(assister.nbr_presence, 0) - COALESCE(assister.nbr_absc, 0) ELSE inscrit.nbr_seance - COALESCE(assister.nbr_presence, 0) - COALESCE(assister.nbr_absc, 0) + 1
```

```
END AS nombre_restant_de_seances FROM inscrit,assister where  
inscrit.CIN=assister.CIN
```

la liste des groupes pleines:

```
SELECT id_grp,nbre_eleves FROM groupe WHERE nbre_eleves = 8
```

la liste de jour et de type de séances:

```
select distinct jour,type from seance
```

la liste des étudiants avec leurs groupes:

```
SELECT e.cin,e.nom,e.prenom,e.niveau,a.id_grp from etudiant  
e,appartient a where e.CIN=a.CIN
```

la liste des étudiants qui ont terminé leurs séances:

```
SELECT inscrit.CIN AS cin_etudiant, inscrit.id_formation,  
inscrit.nbr_seance, assister.nbr_presence FROM inscrit JOIN assister ON  
inscrit.CIN = assister.CIN WHERE (inscrit.nbr_seance -  
COALESCE(assister.nbr_presence, 0) - COALESCE(assister.nbr_absc, 0) +  
1) = 0
```

Conclusion:

En conclusion de ce rapport sur le projet de gestion d'un centre de formation, nous avons exploré en détail les différentes facettes de la conception et de la mise en œuvre d'une base de données pour un tel établissement. La création d'une base de données efficace et bien structurée est essentielle pour la gestion fluide des étudiants, des cours, des formateurs et des divers aspects administratifs du centre.

L'utilisation de SQL (Structured Query Language) a été au cœur de ce projet, nous permettant de définir et de manipuler les différentes tables de la base de données. Les requêtes SQL ont été cruciales pour extraire des informations pertinentes, mettre

à jour des données, et assurer une intégrité et une cohérence globales dans le système.

Ainsi que L'utilisation de Tkinter a enrichi l'expérience utilisateur en offrant une interface graphique conviviale pour interagir avec la base de données. Cette intégration a permis de créer des fenêtres, des boutons et des champs interactifs, offrant ainsi une manière intuitive et visuelle de gérer les informations du centre de formation.