

BTS SIO 2025

☐ Administration des systèmes et des réseaux (E6 – SISR)

☒ Conception et développement d'applications (E6 – SLAM)

PAGE DE PRÉSENTATION DU DOSSIER

N° d'inscription¹ : |0|2|1|1|1|0|4|0|8|1|8|

NOM : CHARNI

PRÉNOM : Louay

Date de passage ¹ : 01 /04/ 2025	Heure de passage ¹ : 8h00
---	--------------------------------------

CATÉGORIE CANDIDAT ² (UNE CASE À COCHER)	
<input checked="" type="checkbox"/> Scolaire <input type="checkbox"/> Apprenti <input type="checkbox"/> Formation professionnelle continue <input type="checkbox"/> Expérience professionnelle 3 ans	<input type="checkbox"/> Ex-scolaire <input type="checkbox"/> Ex-apprenti <input type="checkbox"/> Ex-formation professionnelle continue

¹ Informations communiquées sur votre convocation envoyée courant mars 2025 sur votre compte **Cyclades**

² Informations communiquées sur votre confirmation d'inscription.

Tampon de L'établissement

SIEC – maison des examens

7 rue Ernest Renan
94749 ARCUEIL CEDEX
Tél : 01 49 12 23 00
www.siec.education.fr



BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2025
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)	
Épreuve E6 - Conception et développement d'applications (option SLAM)	

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 01
Nom, prénom : Chami louay		N° candidat : 02111040818
Épreuve ponctuelle <input type="checkbox"/>	Contrôle en cours de formation <input checked="" type="checkbox"/>	Date : 10/12/2024
Organisation support de la réalisation professionnelle Projet réalisé lors des séances de TD.		
Intitulé de la réalisation professionnelle Site web gestion des Jeux Olympiques		
Période de réalisation : Du 10/10/2024 au 08/11/2024 Lieu : Lycée la Tournelle de La Garenne-Colombes		
<input checked="" type="checkbox"/>	Modalité : <input type="checkbox"/>	Seul(e) <input type="checkbox"/> En équipe <input checked="" type="checkbox"/>
Compétences travaillées <input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données		
Conditions de réalisation⁵ (ressources fournies, résultats attendus) Travail en classe et à la maison à partir d'un support étudiant reprenant un contexte professionnel. L'objectif était de créer un site web dans lequel les utilisateurs pourraient consulter les sports, le calendrier des épreuves et les résultats des JO 2028. Une partie administration était également prévue dans le développement.		
Description des ressources documentaires, matérielles et logicielles utilisées⁶ Ressources documentaires : <input type="checkbox"/> Supports étudiants (cours), Travaux dirigés et travaux pratiques <input type="checkbox"/> Sites de ressources en ligne : Github Ressources matérielles : <input type="checkbox"/> Ordinateur du Lycée <input type="checkbox"/> Connexion internet Ressources logicielles : <input type="checkbox"/> IDE : Visual Studio Code <input type="checkbox"/> Ensemble de solution logicielle : MAMP (serveur Apache et base de données relationnelles MYSQL) <input type="checkbox"/> Outil de modélisation de base de données : Mocodo <input type="checkbox"/> Outil de modélisation UML : Visual Paradigm Langues <input type="checkbox"/> Langages web : HTML5, CSS3, JavaScript et PHP 8 <input type="checkbox"/> Langage base de données : SQL		
Modalités d'accès aux productions⁷ et à leur documentation⁸ Cahier des charges techniques : https://tinyurl.com/458eddu7 Code : https://tinyurl.com/yv5uhc8e Site : jo-app-louay.kesug.com		

⁵ En référence aux conditions de réalisation et ressources nécessaires du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

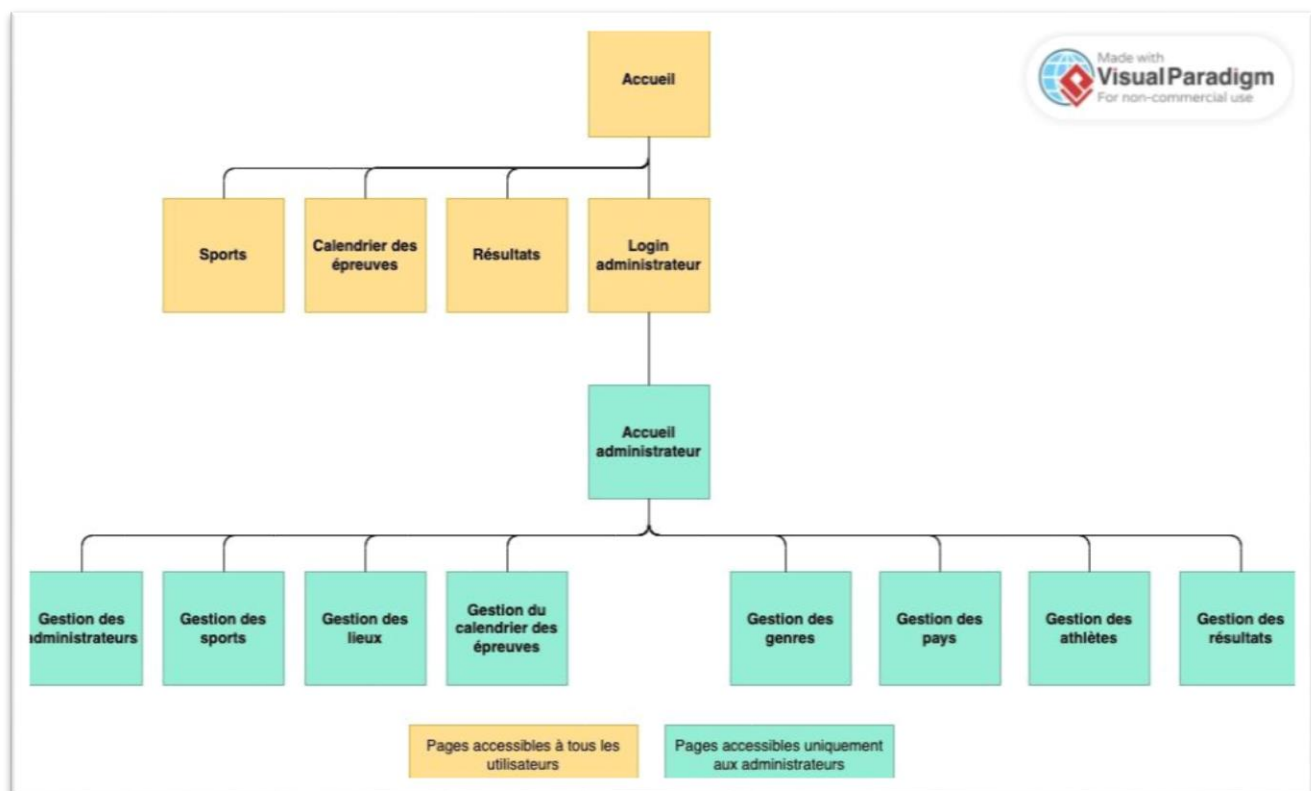
⁶ Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

⁷ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

⁸ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E6 - Conception et développement d'applications (option SLAM)**

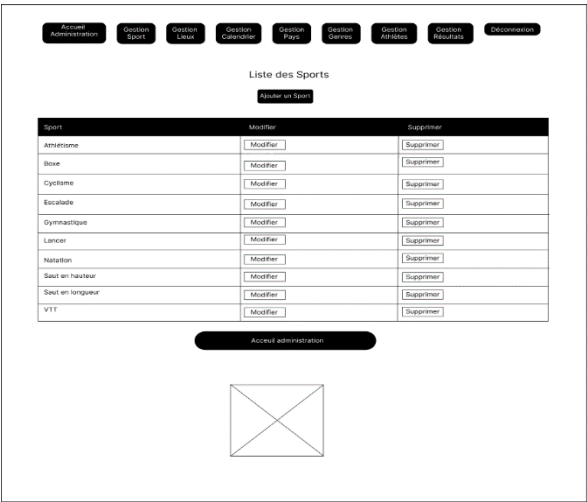
L'objectif était de créer un site web dans lequel les utilisateurs pourraient consulter les sports, le calendrier des épreuves et les résultats des JO 2028. Une partie administration était également prévue dans le développement.

Arborescences

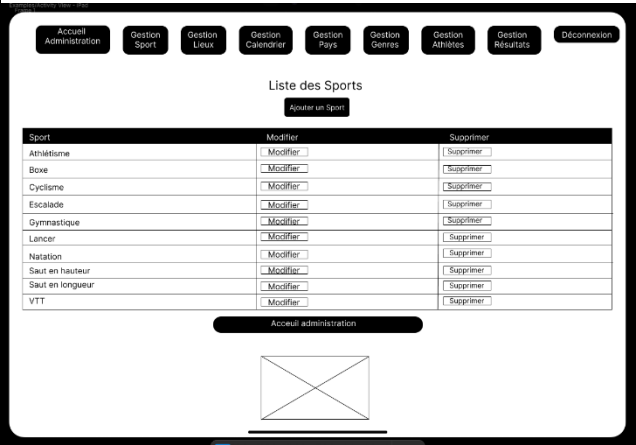
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Wireframes pages de connexion

Ordinateur :



Tablette :



ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Téléphone :

Accueil

Sports

Calendrier des événements

Résultats

Accès administrateur

Connexion

Login :

Mot de passe :

Se connecter

Ajouter un Sport

Gestion Sports

Gestion lieux

Gestion Calendrier

Gestion Pays

Gestion Genres

Gestion Athlètes

Gestion Résultats

Déconnexion

Liste des Sports

Ajouter un Sport

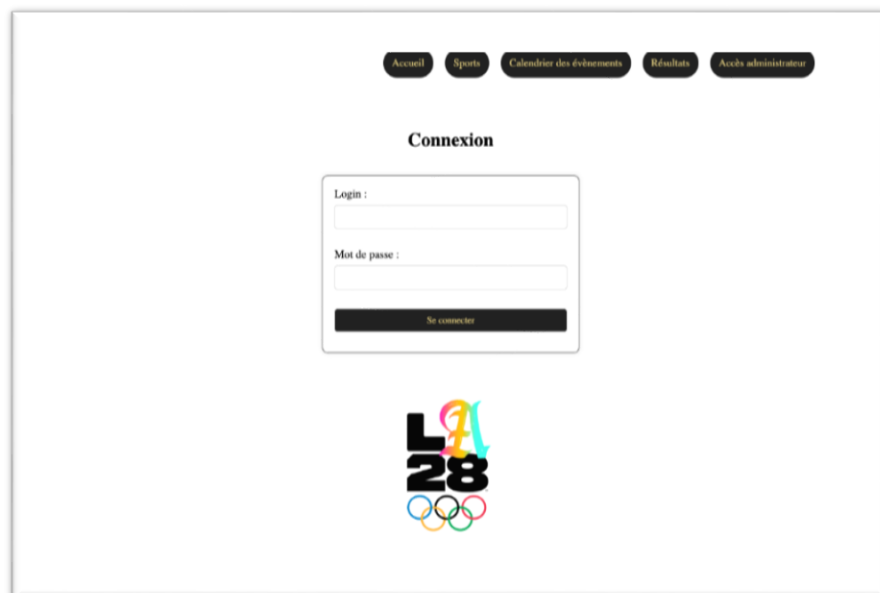
Sport	Modifier	Supprimer
Athlétisme	Modifier	Supprimer
Boxe	Modifier	Supprimer
Cyclisme	Modifier	Supprimer
Escalade	Modifier	Supprimer
Gymnastique	Modifier	Supprimer
Lancer	Modifier	Supprimer
Natation	Modifier	Supprimer
Saut en hauteur	Modifier	Supprimer
Saut en longueur	Modifier	Supprimer
VTT	Modifier	Supprimer

Accueil administration

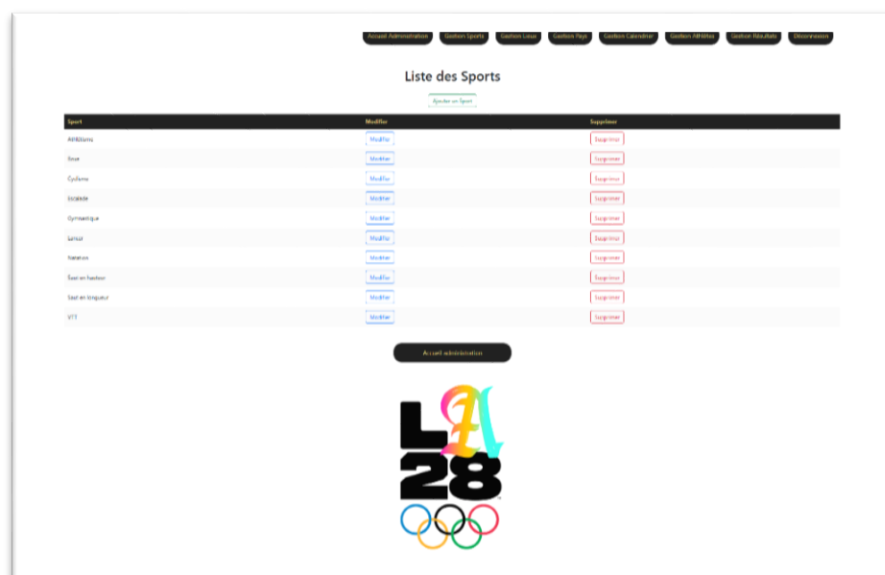
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Maquettes pages de connexion

Ordinateur :



Maquette de la page de connexion. Le header contient cinq boutons : Accueil, Sports, Calendrier des événements, Résultats, et Accès administrateur. Le titre principal est "Connexion". Le formulaire de connexion comprend deux champs : "Login :" et "Mot de passe :", suivis d'un bouton "Se connecter". En bas, se trouve le logo "L1 28" avec les anneaux olympiques.



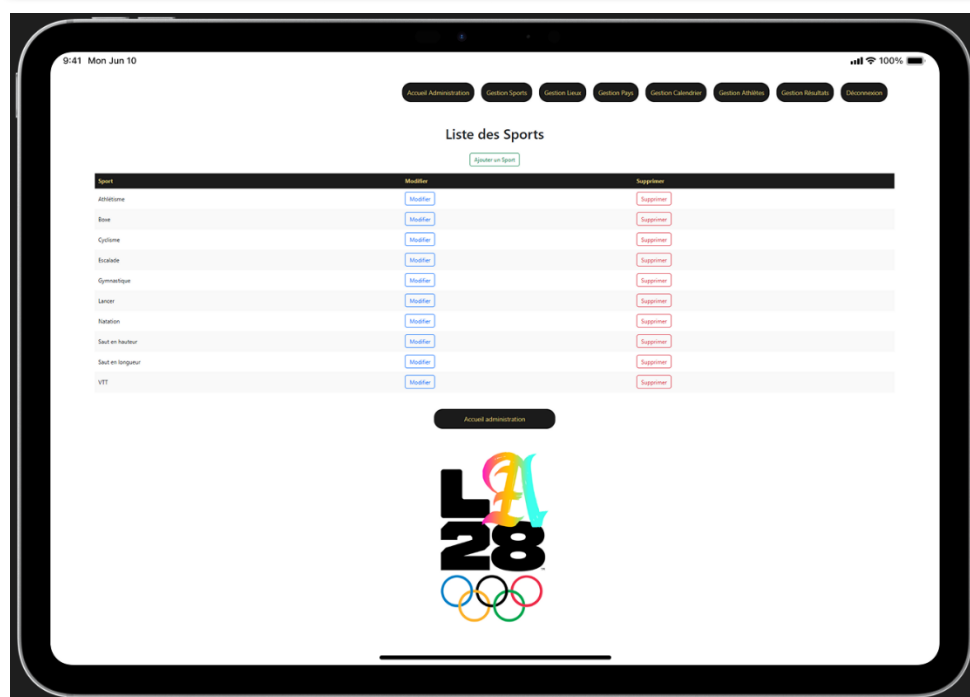
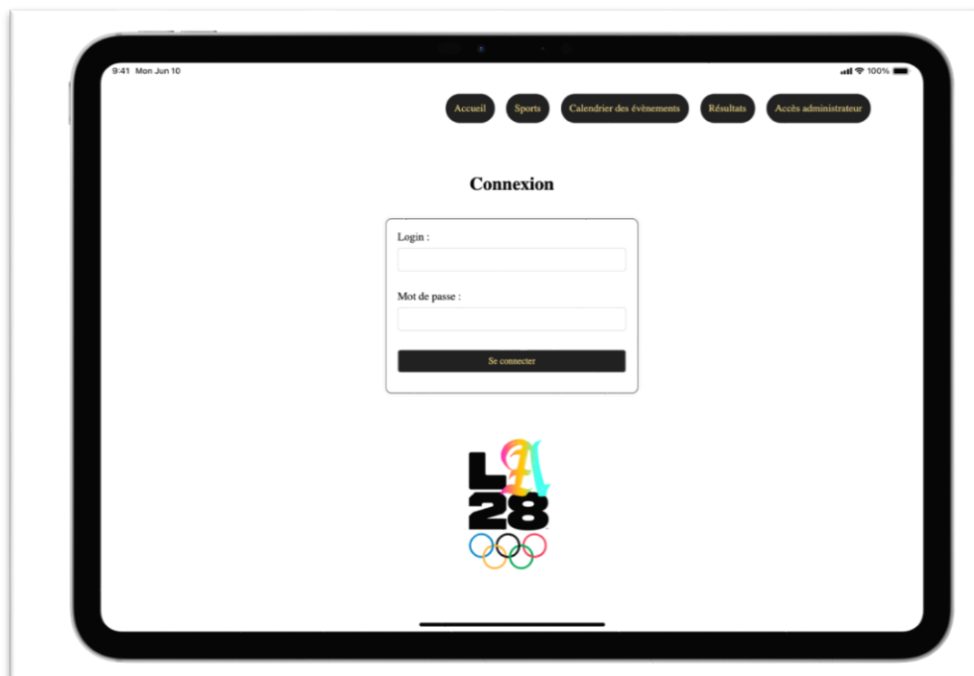
Maquette de la page "Liste des Sports". Le header contient huit boutons : Accueil, Informations, Sports, Calendrier, Résultats, Accès administrateur, Statistiques, et Références. Le titre principal est "Liste des Sports". Une sous-titre "Sélectionner un Sport" est visible. La page présente un tableau à trois colonnes : "Sport", "Modifier", et "Supprimer".

Sport	Modifier	Supprimer
Arts Martiaux	Modifier	Supprimer
Boxe	Modifier	Supprimer
Canoe	Modifier	Supprimer
Cyclisme	Modifier	Supprimer
Escalade	Modifier	Supprimer
Gymnastique	Modifier	Supprimer
Judo	Modifier	Supprimer
Natation	Modifier	Supprimer
Rugby	Modifier	Supprimer
Tennis	Modifier	Supprimer
Tennis de Table	Modifier	Supprimer
Tennis de Synthèse	Modifier	Supprimer
VTT	Modifier	Supprimer

En bas de la page, il y a un bouton "Accès administrateur" et le logo "L1 28" avec les anneaux olympiques.

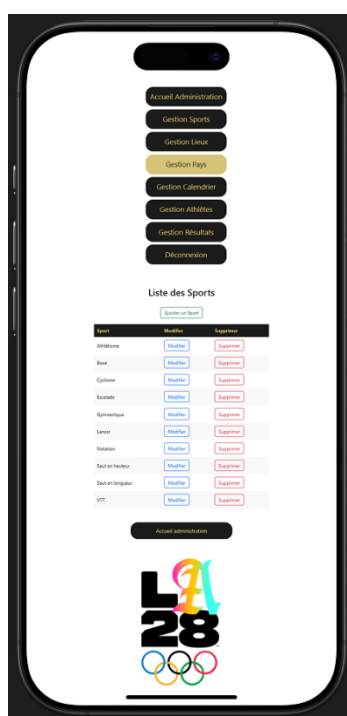
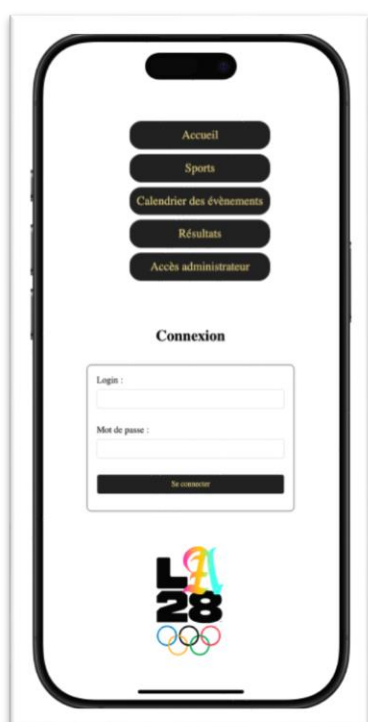
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Tablette :



ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

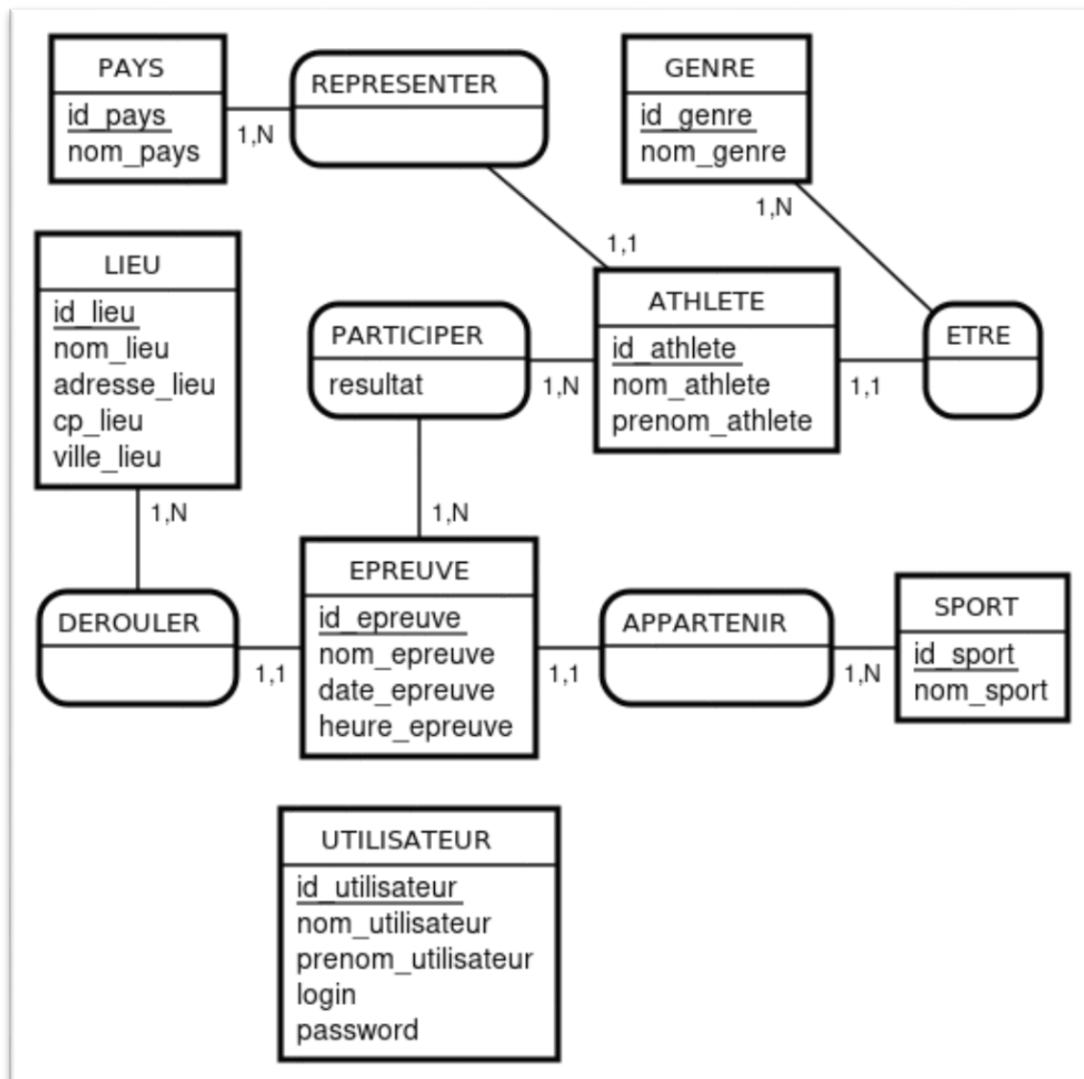
Téléphone :



ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Dictionnaire de données

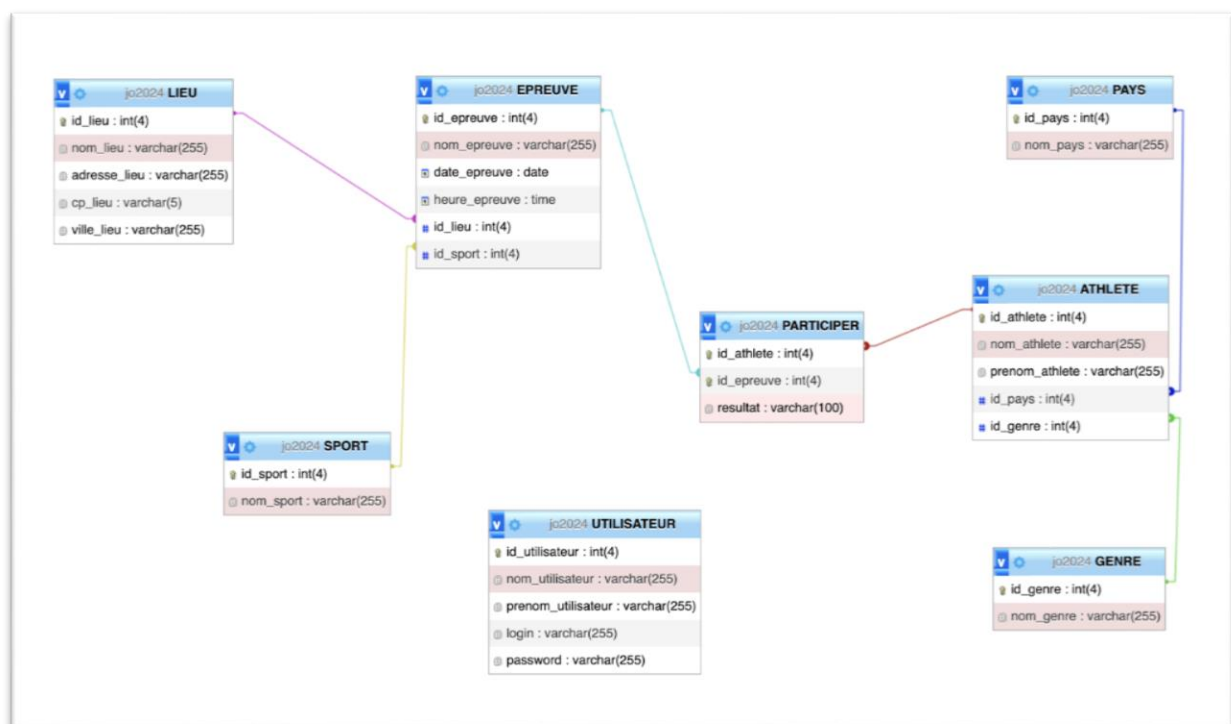
Entité ou association	Libellé de l'attribut	Type
ATHLETE	id_athlete	INT(4)
/	nom_athlete	VARCHAR(255)
/	prenom_athlete	VARCHAR(255)
EPREUVE	id_epreuve	INT(4)
/	nom_epreuve	VARCHAR(255)
/	date_epreuve	DATE
/	heure_epreuve	TIME
GENRE	id_genre	INT(4)
/	nom_genre	VARCHAR(255)
LIEU	id_lieu	INT(4)
/	nom_lieu	VARCHAR(255)
/	adresse_lieu	VARCHAR(255)
/	cp_lieu	VARCHAR(5)
/	ville_lieu	VARCHAR(255)
PARTICIPER	resultat	VARCHAR(100)
PAYS	id_pays	INT(4)
/	nom_pays	VARCHAR(255)
SPORT	id_sport	INT(4)
/	nom_sport	VARCHAR(255)
UTILISATEUR	id_utilisateur	INT(4)
/	nom_utilisateur	VARCHAR(255)
/	prenom_utilisateur	VARCHAR(255)
/	login	VARCHAR(255)
/	password	VARCHAR(255)

Modèles Conceptuel de Données (MCD)

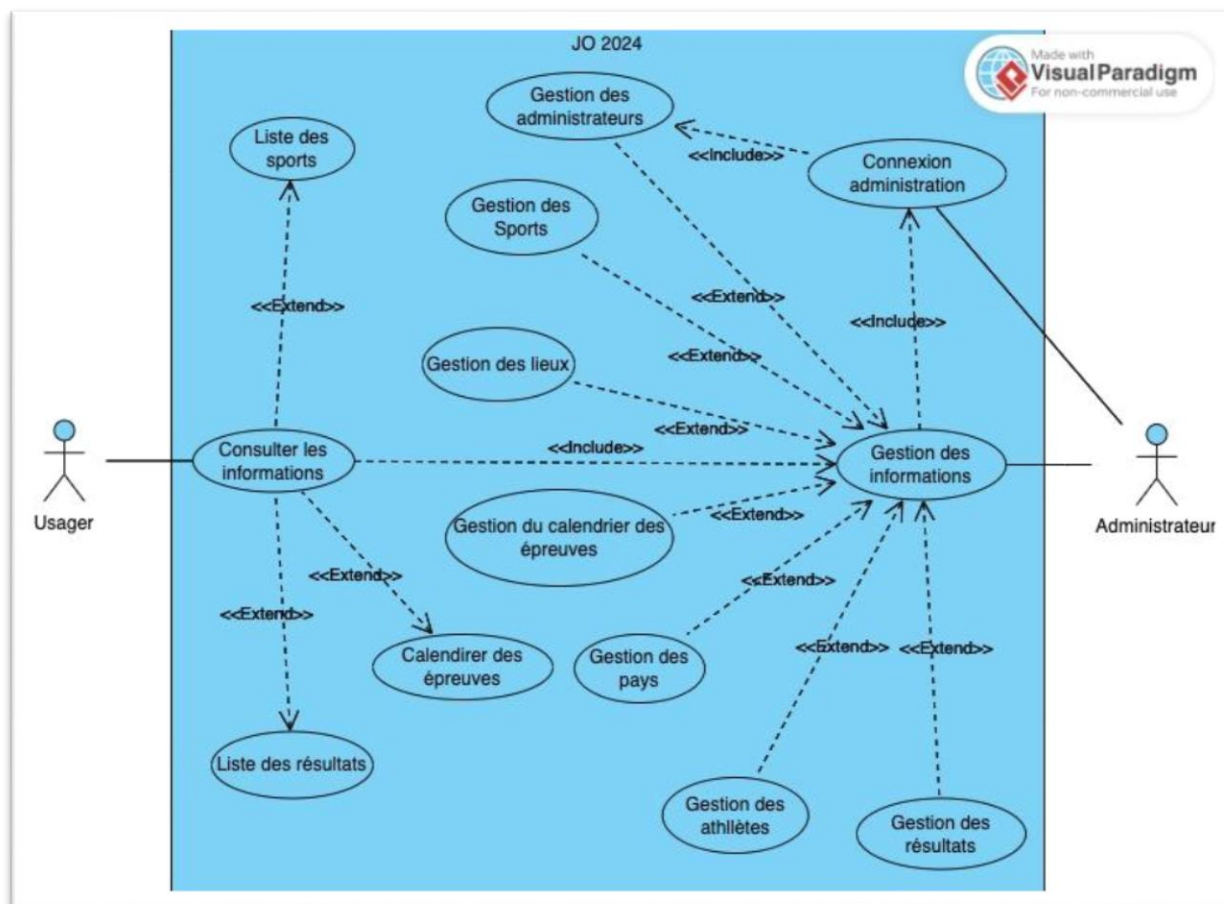
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Modèle Logique de Données (MLD)

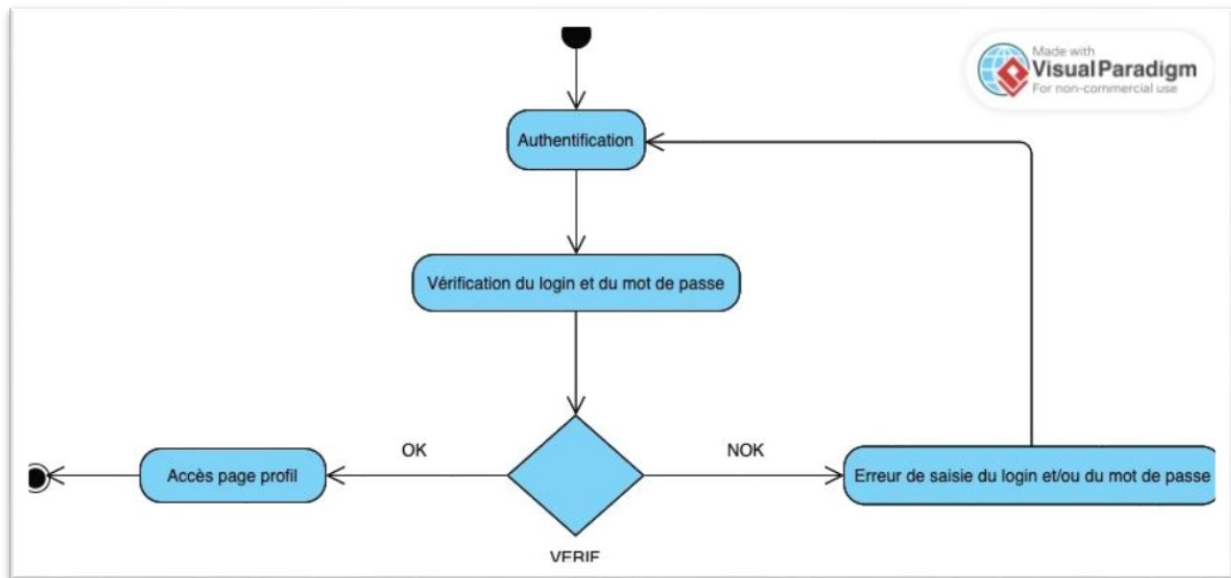
- PAYS(id_pays, nom_pays)
 - Clé primaire : id_pays
- GENRE(id_genre, nom_genre)
 - Clé primaire : id_genre
- LIEU(id_lieu, nom_lieu, adresse_lieu, cp_lieu, ville_lieu)
 - Clé primaire : id_lieu
- PARTICIPER(id_athlete, id_epreuve, resultat)
 - Clé primaire : id_athlete, id_epreuve
 - Clé étrangère : id_athlete en référence à id_athlete de ATHLETE
 - id_epreuve en référence à id_epreuve de EPREUVE
- ATHLETE(id_athlete, nom_athlete, prenom_athlete)
 - Clé primaire : id_athlete
- EPREUVE(id_epreuve, nom_epreuve, date_epreuve, heure_epreuve)
 - Clé primaire : id_epreuve
- SPORT(id_sport, nom_sport)
 - Clé primaire : id_sport
- UTILISATEUR(id_utilisateur, nom_utilisateur, prenom_utilisateur, login, password)
 - Clé primaire : id_utilisateur

Modèle Physique de Données (MPD)

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Diagramme de cas d'utilisation

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Diagramme d'activités (gestion des login)**Exemple de code – Gestion des login (login.php)**

```

<main>
  <?php
  // Affichage d'un message d'erreur en cas de tentative de connexion échouée
  if (isset($_SESSION['error'])) {
    echo '<p style="color: red;">' . htmlspecialchars($_SESSION['error']) . '</p>';
    unset($_SESSION['error']);
  }

  // Activer l'affichage des erreurs pour le développement
  error_reporting(E_ALL);
  ini_set("display_errors", 1);
  ?>

  <h1>Connexion</h1>

  <!-- Formulaire de connexion avec protection CSRF -->
  <form action="../database/auth.php" method="post">
    <input type="hidden" name="csrf_token" value="<?php echo htmlspecialchars($_SESSION['csrf_token']); ?>">

    <label for="login">Login :</label>
    <input type="text" name="login" id="login" required><br><br>

    <label for="password">Mot de passe :</label>
    <input type="password" name="password" id="password" required><br><br>

    <input type="submit" value="Se connecter">
  </form>
</main>

```

Ce code PHP affiche un formulaire de connexion.

Il montre un message d'erreur si la tentative de connexion a échoué.

L'affichage des erreurs PHP est activé pour le développement.

Le formulaire envoie les données en méthode POST vers un fichier `auth.php`.

Il contient un jeton CSRF pour éviter les attaques.

L'utilisateur doit entrer son login et son mot de passe.

Enfin, un bouton permet de valider la connexion.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2025
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)	
Épreuve E6 - Conception et développement d'applications (option SLAM)	

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 2
Nom, prénom : Charni Louay		N° candidat : 02111040818
Épreuve ponctuelle <input checked="" type="checkbox"/>	Contrôle en cours de formation <input type="checkbox"/>	Date : 12/12/2024
Organisation support de la réalisation professionnelle Projet réalisé en séances de TP et à la maison		
Intitulé de la réalisation professionnelle Logicielle de gestion et cartes grises (Java)		
Période de réalisation : 28/11/2024 au 06/03/2025 Lieu : Lycée la Tournelle de La Garenne-Colombes		
Modalité : <input checked="" type="checkbox"/> Seul(e) <input type="checkbox"/> En équipe		
Compétences travaillées <input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données		
Conditions de réalisation⁵ (ressources fournies, résultats attendus) Travail en classe et à la maison à partir d'un support étudiant reprenant un contexte professionnel. L'objectif est de concevoir et développer un logiciel de gestion des cartes grises (ajout, modification et suppression d'information). Le logiciel est développé en Java.		
Description des ressources documentaires, matérielles et logicielles utilisées⁶ Ressources documentaires : <ul style="list-style-type: none"> • Supports étudiants (cours), Travaux dirigés et travaux pratiques • Sites de ressources en ligne : Github Ressources matérielles : <ul style="list-style-type: none"> • Ordinateur fixe, souris, clavier et écran • Connexion internet Ressources logicielles : <ul style="list-style-type: none"> • IDE : Visual Studio Code • Ensemble de solution logicielle : MAMP (serveur Apache et base de données relationnelles MYSQL) • Outil de modélisation de base de données : Mocodo • Outil de modélisation UML : Visual Paradigm • Gestion de projet : Trello • Outil de conception de design : Figma • Plateforme de développement collaborative : GitHub Langages <ul style="list-style-type: none"> • Langages de programmation applicatif : Java • Langage base de données : SQL 		
Modalités d'accès aux productions⁷ et à leur documentation⁸ Cahier des charges techniques : https://tinyurl.com/4wfkvu7f Code : https://tinyurl.com/4t9w757r		

⁵ En référence aux conditions de réalisation et ressources nécessaires du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

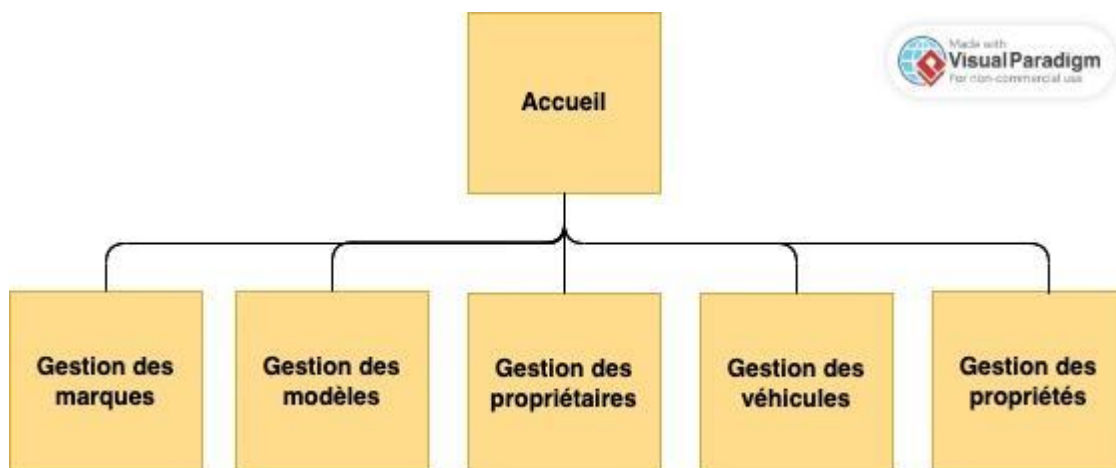
⁶ Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

⁷ Conformément au référentiel du BTS SIO « Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve. ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

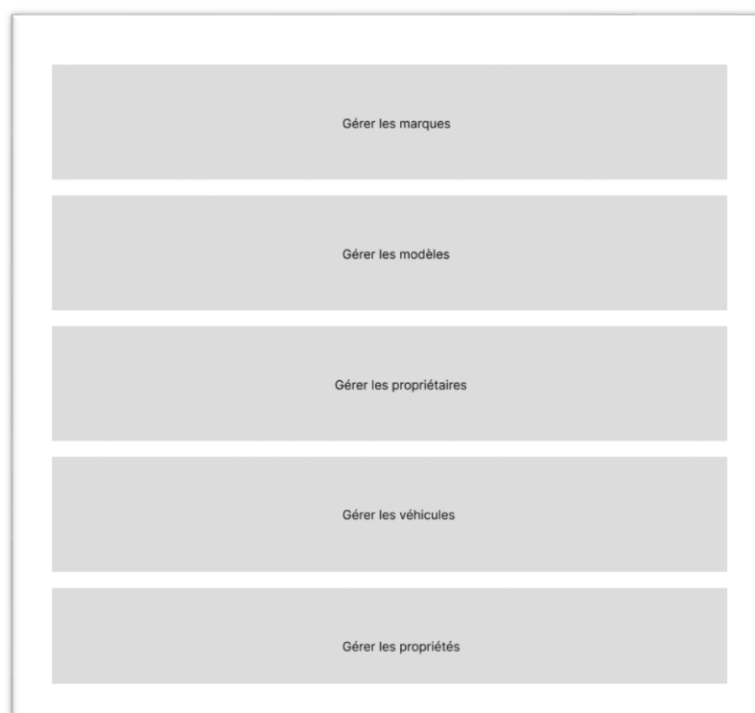
⁸ Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle
(verso, éventuellement pages suivantes)****Épreuve E6 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

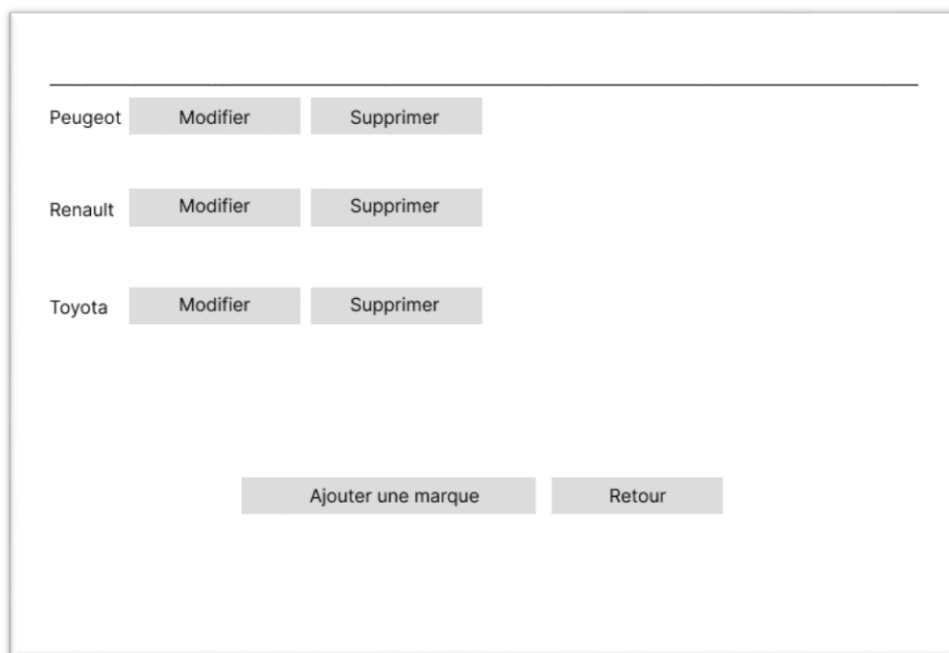
L'objectif est de concevoir et développer un logiciel de gestion des cartes grises (ajout, modification et suppression d'information). Le logiciel est développé en Java.

Arborescence**Wireframes pages d'accueil**

Ordinateur :



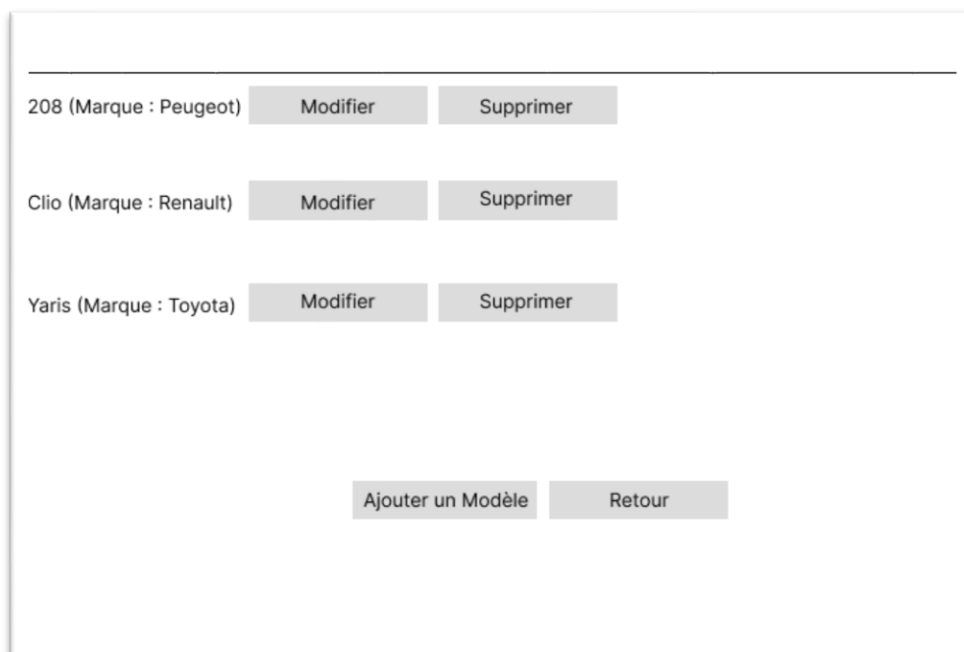
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Wireframes gérer les marques

Wireframe for managing brands. It features a list of brands with 'Modifier' and 'Supprimer' buttons for each. At the bottom, there are buttons for 'Ajouter une marque' and 'Retour'.

Peugeot	Modifier	Supprimer
Renault	Modifier	Supprimer
Toyota	Modifier	Supprimer

Ajouter une marque Retour

Wireframes gérer les modèles

Wireframe for managing models. It features a list of models with their associated brand and 'Modifier' and 'Supprimer' buttons for each. At the bottom, there are buttons for 'Ajouter un Modèle' and 'Retour'.

208 (Marque : Peugeot)	Modifier	Supprimer
Clio (Marque : Renault)	Modifier	Supprimer
Yaris (Marque : Toyota)	Modifier	Supprimer

Ajouter un Modèle Retour

:

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Wireframes gérer les propriétaires

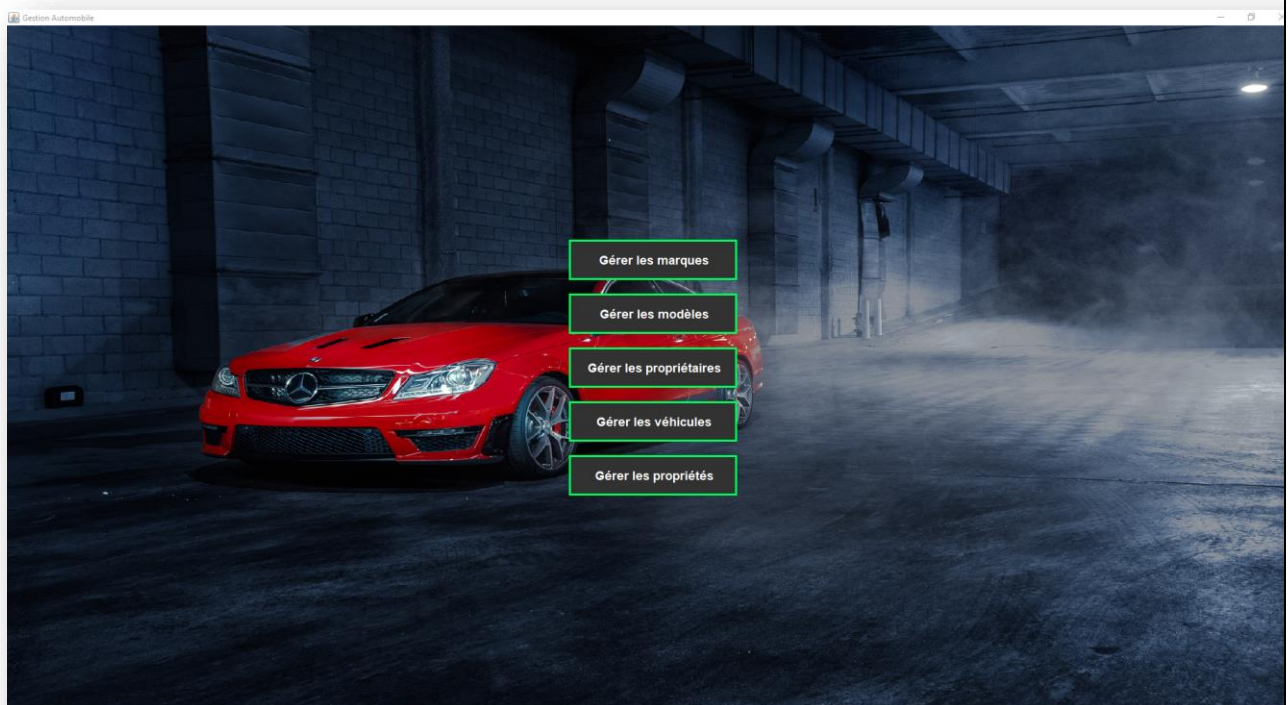
Doe John (123 Rue de paris, 75000 Paris)	Modifier	Supprimer
Doe Jane (456 Avenue des champs, 75008 Paris)	Modifier	Supprimer
Smith Alice (789 Boulevard Haussmann, 75009 Paris)	Modifier	Supprimer

Ajouter un Propriétaire

Retour

Maquettes pages d'accueil

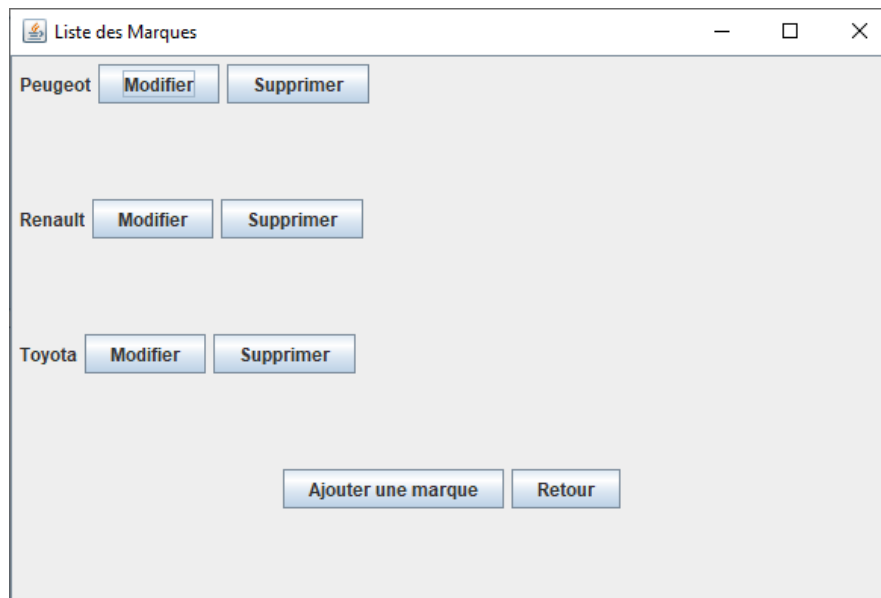
Ordinateur :



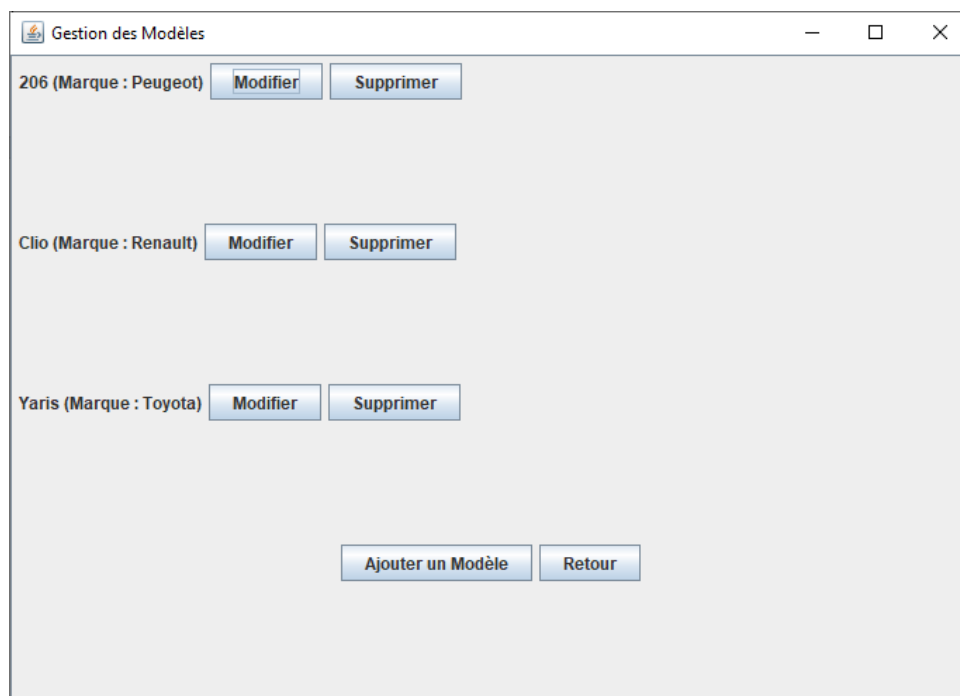
ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Maquettes pages Gérer les marques

Ordinateur :

**Maquettes pages Gérer les modèles**

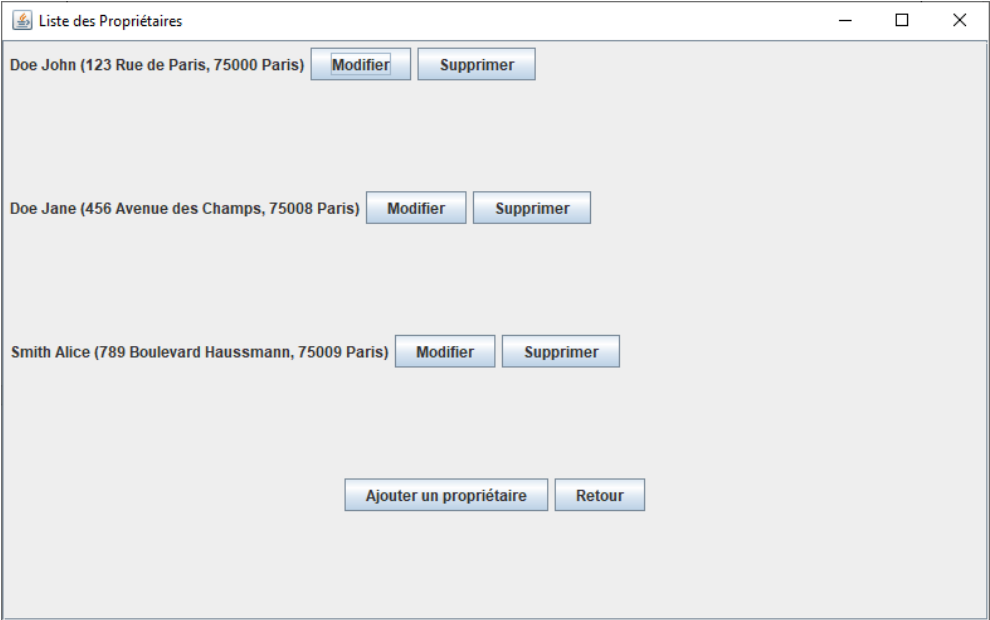
Ordinateur :



ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Maquettes pages Gérer les propriétaires

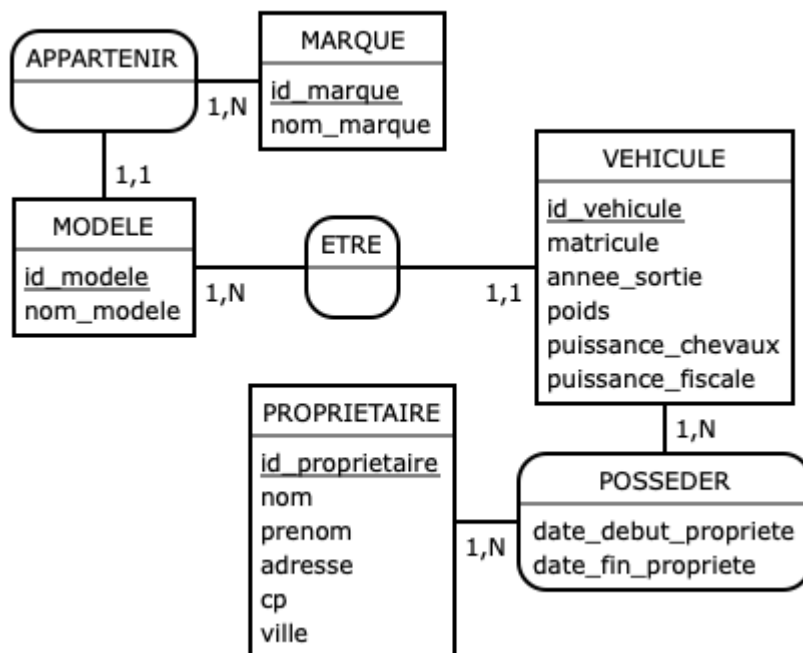
Ordinateur :



Dictionnaire de données

Entité ou association	Libellé de l'attribut	Type
MARQUE	id_marque	INT(4)
	nom_marque	VARCHAR(255)
MODELE	id_modele	INT(4)
	nom_modele	VARCHAR(255)
VEHICULE	id_marque	INT(4)
	id_vehicule	INT(4)
	matricule	VARCHAR(50)
	annee_sortie	VARCHAR(255)
	poids	VARCHAR(255)
	puissance_chevaux	VARCHAR(255)
	puissance_fiscale	VARCHAR(255)
POSSEDER	id_modele	INT(4)
	id_proprietaire	INT(4)
	id_vehicule	INT(4)
	date_debut_propriete	DATE
PROPRIETAIRE	date_fin_propriete	DATE
	id_proprietaire	INT(4)
	nom	VARCHAR(255)
	prenom	VARCHAR(255)
	adresse	VARCHAR(255)
	cp	VARCHAR(10)
	ville	VARCHAR(255)

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Modèles Conceptuel de Données (MCD)**Modèle Logique de Données (MLD)**

MARQUE (id_marque, nom_marque)

Clé primaire : id_marque

MODELE (id_modelle, nom_modelle, id_marque)

Clé primaire : id_modelle

Clé étrangère : id_marque en référence à id_marque de MARQUE

VEHICULE(id_vehicule, matricule, annee_sortie, poids, puissance_chevaux, puissance_fiscale, id_modelle)

Clé primaire : id_vehicule

Clé étrangère : id_modelle en référence à id_modelle de MODELE

POSSEDER(id_proprietaire, id_vehicule, date_debut_propriete, date_fin_propriete)

Clé primaire : id_proprietaire, id_vehicule

Clé étrangère : id_proprietaire en référence à id_proprietaire de PROPRIETAIRE

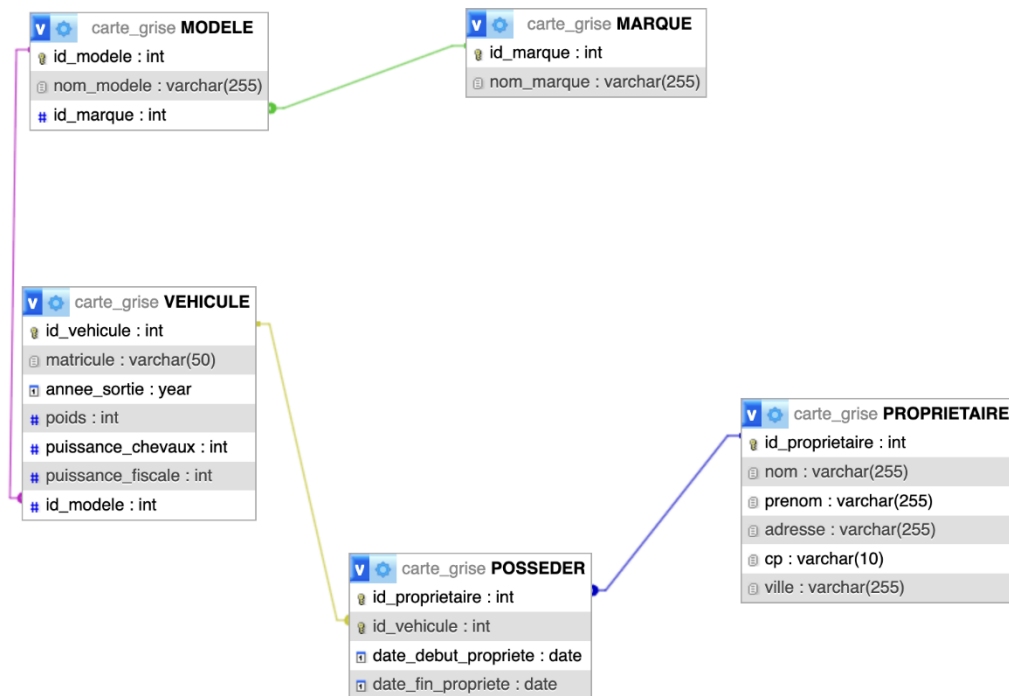
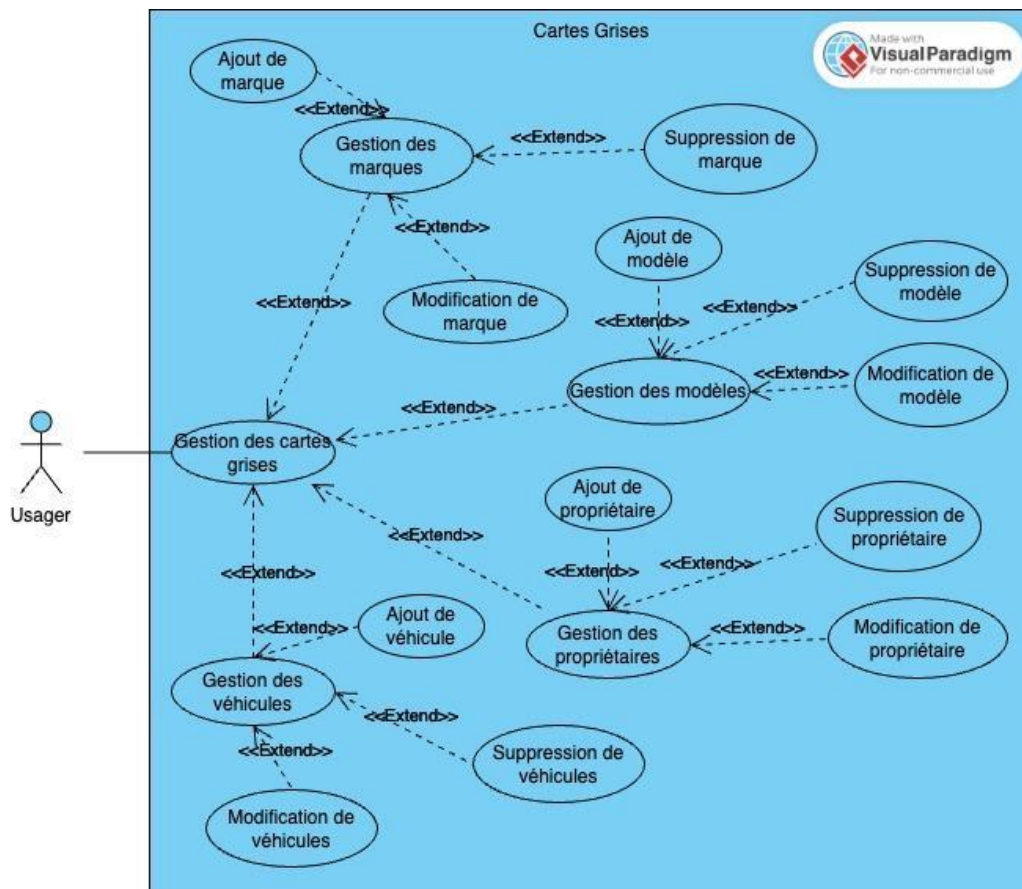
id_vehicule en référence à id_vehicule de VEHICULE

PROPRIETAIRE(id_proprietaire, nom, prenom, adresse, cp, ville)

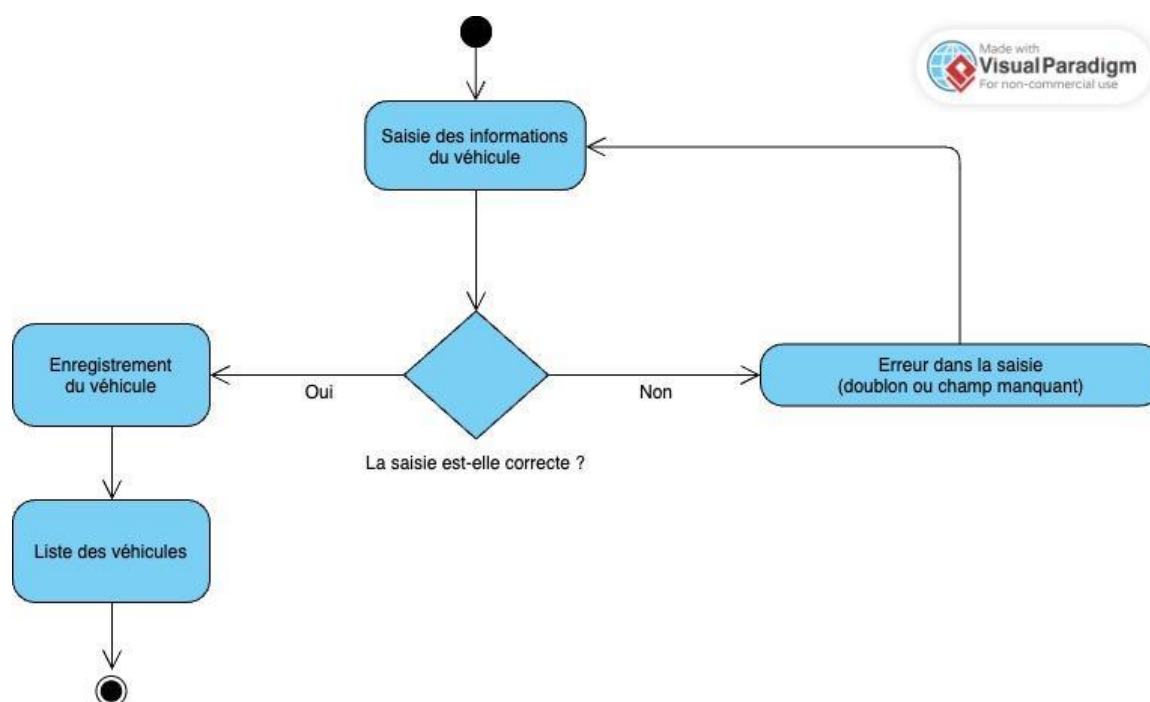
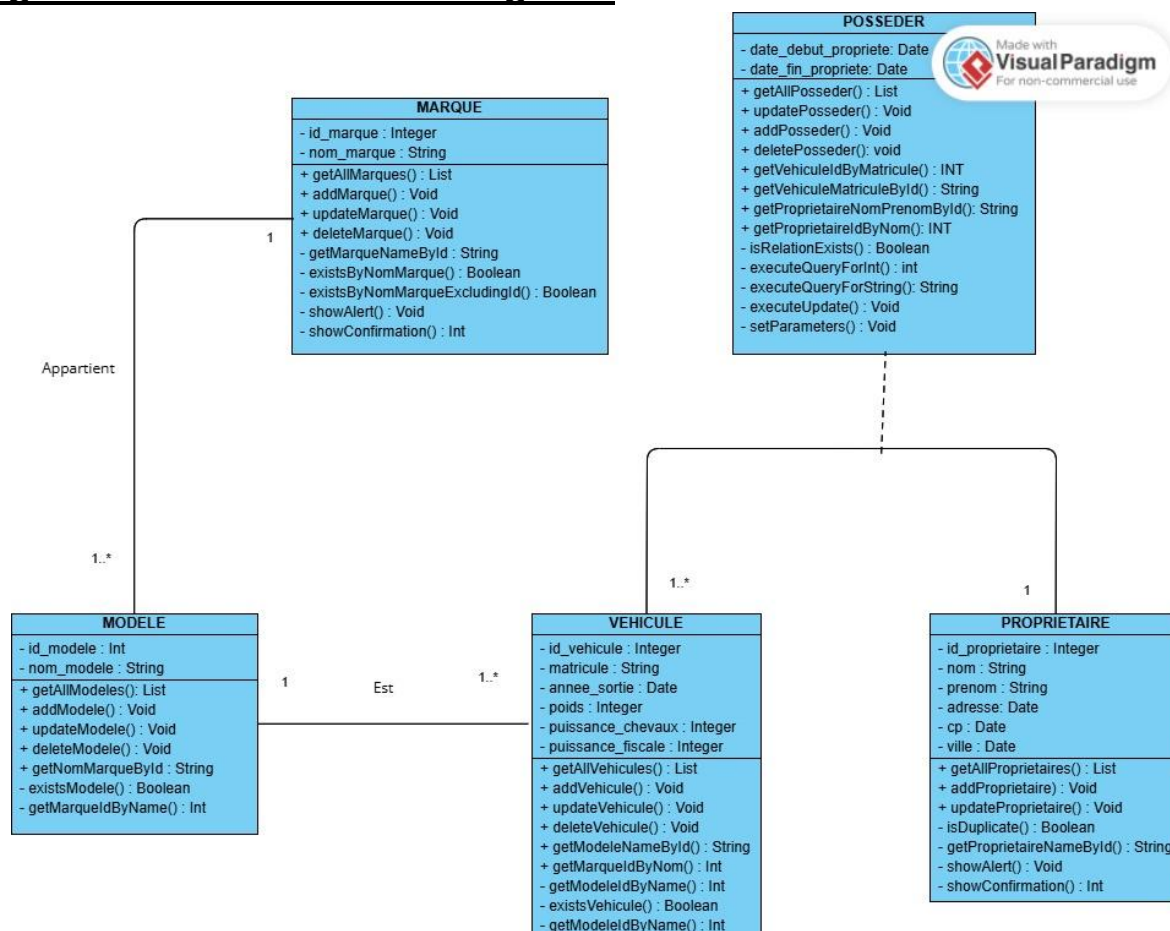
Clé primaire : id_proprietaire

Modèle Physique de Données (MPD)

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Diagramme de cas d'utilisationDiagramme d'activités (ajout des vehicules)

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Diagramme des classes cartes grisesExemple de code – Gestion des propriétés (PossederController.java)

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

1. Code 1 : PossederView.java

Explication du code :

Ce code correspond à une méthode `addPossederAction` qui permet d'ajouter une nouvelle relation "**Posséder**" entre un véhicule et un propriétaire dans une application Java Swing.

Fonctionnement :

1. Création d'un formulaire :

- 4 champs sont demandés à l'utilisateur via une boîte de dialogue :
- Matricule du véhicule
- Nom du propriétaire
- Date de début de propriété
- Date de fin de propriété (optionnelle)

2. Récupération et vérification des données :

- Les identifiants du véhicule et du propriétaire sont récupérés via un contrôleur.
- Si l'un d'eux n'est pas trouvé, un message d'erreur s'affiche.

3. Ajout de la relation :

- Les dates sont converties au format `Date`.
- La relation est ajoutée en appelant la méthode `controller.addPosseder()`.

4. Actualisation de la vue :

- La méthode `refreshView()` est appelée pour mettre à jour l'affichage.

En résumé, ce code ouvre un formulaire pour saisir les informations et ajoute la relation véhicule-propriétaire si les données sont valides.

Lien : Ce code fait le lien entre l'interface graphique et la base de données, en passant par le contrôleur.

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

*** Action pour ajouter une nouvelle relation Posséder.**

```
*/
private void addPossederAction(ActionEvent e) {
    JTextField matriculeField = new JTextField(10);
    JTextField proprietaireNomField = new JTextField(10);
    JTextField dateDebutField = new JTextField(10);
    JTextField dateFinField = new JTextField(10);

    Object[] message = {
        "Matricule Véhicule:", matriculeField,
        "Nom du Propriétaire:", proprietaireNomField,
        "Date Début Propriété (YYYY-MM-DD):", dateDebutField,
        "Date Fin Propriété (YYYY-MM-DD) (optionnel):", dateFinField
    };

    int option = JOptionPane.showConfirmDialog(this, message, "Ajouter une
Relation", JOptionPane.OK_CANCEL_OPTION);
    if (option == JOptionPane.OK_OPTION) {
        try {
            int idVehicule =
controller.getVehiculeIdByMatricule(matriculeField.getText());
            int idProprietaire =
controller.getProprietaireIdByNom(proprietaireNomField.getText());

            if (idVehicule == -1 || idProprietaire == -1) {
                JOptionPane.showMessageDialog(this, "Matricule ou
propriétaire non trouvé.", "Erreur", JOptionPane.ERROR_MESSAGE);
                return;
            }

            SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
            java.util.Date dateDebut =
sdf.parse(dateDebutField.getText());
            java.util.Date dateFin = dateFinField.getText().isEmpty() ?
null : sdf.parse(dateFinField.getText());

            controller.addPosseder(idProprietaire,
matriculeField.getText(), dateDebut, dateFin);
            refreshView();
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, "Erreur dans les données
fournies", "Erreur", JOptionPane.ERROR_MESSAGE);
        }
    }
}
```


ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Explication du code :

Le contrôleur :

- Recherche l'ID du véhicule grâce à `getVehiculeIdByMatricule()`.
- Vérifie si la relation propriétaire-véhicule existe déjà avec `isRelationExists()`.
- Si ce n'est pas le cas, il exécute une requête SQL d'insertion dans la table **POSSEDER** avec les données reçues.
- L'insertion est réalisée via `executeUpdate()`.

Lien : Ce code fait le pont entre la vue et la base de données. Il vérifie les doublons, envoie les données vers la BDD et s'appuie sur le modèle `Posseder` pour organiser les données.

```
/**
 * Ajoute une nouvelle relation de possession entre un propriétaire
 et un véhicule.
 * @param idProprietaire Identifiant du propriétaire
 * @param matriculeVehicule Matricule du véhicule
 * @param dateDebut Date de début de possession
 * @param dateFin Date de fin de possession (peut être null)
 */
public void addPosseder(int idProprietaire, String
matriculeVehicule, java.util.Date dateDebut, java.util.Date dateFin) {
    int idVehicule = getVehiculeIdByMatricule(matriculeVehicule);

    // Vérifie si le véhicule existe et si la relation n'existe pas
    déjà
    if (idVehicule == -1 || isRelationExists(idProprietaire,
idVehicule)) {
        JOptionPane.showMessageDialog(null, "Relation existante ou
véhicule introuvable.", "Erreur", JOptionPane.ERROR_MESSAGE);
        return;
    }

    String query = "INSERT INTO POSSEDER (id_proprietaire,
id_vehicule, date_debut_propriete, date_fin_propriete) VALUES (?, ?, ?,
?)";

    executeUpdate(query, new Object[]{
        idProprietaire,
        idVehicule,
        new java.sql.Date(dateDebut.getTime()),
        dateFin != null ? new java.sql.Date(dateFin.getTime()) :
null
    });
}
```

ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle

Fichier : `Posseder.java` (Modèle)

Ce fichier représente **la classe qui mappe la table "POSSEDER" de la base de données.**

Explication du code :

La classe **Posseder** représente une relation entre un propriétaire et un véhicule avec :

- Des attributs :
 - `id_proprietaire`
 - `id_vehicule`
 - `date_debut_propriete`
 - `date_fin_propriete` (optionnelle)
- Des méthodes clés :
 - Un constructeur pour créer un objet `Posseder`.
 - Des getters et setters pour accéder et modifier les données.
 - `toString()` pour afficher la relation sous forme de texte.
 - `estProprietaireActuel()` pour savoir si la relation est toujours valide (pas de date de fin).

Lien :

La classe est utilisée par le contrôleur pour manipuler et stocker les données récupérées depuis la base de données. Elle sert de modèle pour représenter chaque relation propriétaire-véhicule.

```
/**
 * Classe représentant la table "POSSEDER" de la base de données.
 * Les noms des attributs correspondent aux colonnes de la table.
 */
public class Posseder {
    private int id_proprietaire; // Identifiant unique du propriétaire (clé étrangère)
    private int id_vehicule; // Identifiant du véhicule associé (clé étrangère)
    private Date date_debut_propriete; // Date de début de la propriété du véhicule
    private Date date_fin_propriete; // Date de fin de la propriété du véhicule (peut être null si la propriété est
    toujours active)

    /**
     * Constructeur principal permettant d'initialiser un objet Posseder.
     * @param id_proprietaire Identifiant du propriétaire
     * @param id_vehicule Identifiant du véhicule
     * @param date_debut_propriete Date de début de la propriété
     * @param date_fin_propriete Date de fin de la propriété (peut être null)
     */
    public Posseder(int id_proprietaire, int id_vehicule, Date date_debut_propriete, Date date_fin_propriete) {
        this.id_proprietaire = id_proprietaire;
        this.id_vehicule = id_vehicule;
        this.date_debut_propriete = date_debut_propriete;
        this.date_fin_propriete = date_fin_propriete;
    }
}
```

ANNEXE 9-2 : Modèle de fiche de contrôle de conformité pour l'épreuve

☐ Épreuve E6 - Administration des systèmes et des réseaux (option SISR)☒ Épreuve E6 - Conception et développement d'applications (option SLAM)

CONTRÔLE DE CONFORMITÉ

Nom et prénom : CHARNI Louay

N° candidat : 02111040818

Conformément à l'arrêté du 22 juillet 2008 (B0 n° 32 du 28 août 2008) fixant définition et conditions de délivrance de certaines spécialités de brevet de technicien supérieur dont l'*annexe I* définissant le contrôle de conformité du dossier support d'épreuve, une commission de contrôle a été chargée d'apprécier la conformité des dossiers des candidats.

Après vérification, votre candidature ne peut être retenue pour le(s) motif(s) ci-dessous :

☐ absence de dossier ;☐ dépôt du dossier au-delà de la date fixée par les autorités académiques.

Vous ne pourrez pas être interrogé(e), la note « non valide » (NV) vous sera attribuée pour l'épreuve, et le diplôme ne pourra vous être délivré.

Date du contrôle :

Visa :