**Une image contenant graphisme, Graphique, créativité

Description générée automatiquement** **Une image contenant texte, Police, logo, Graphique

Description générée automatiquement**

**UNIVERSITÉ NANGUI ABROGOUA CENTRE DE FORMATION CONTINUE**

|  |  |
| --- | --- |
|  |  |

**MASTER 1**

**STATITISTIQUE ET INFORMATIQUE DÉCISIONNELLE**

**SID 2 MASTER 1, 2023-2024**

***PROJETS A RENDRE :***

* Projet Web
* Projet de protection de données
* Projet XML

|  |  |
| --- | --- |
| **Auditeur :** YEO Domèhin David | **Professeur :** Dr Zézé Sylvain |
|  |  |

Table des matières

[I. PROJET WEB: MON APPLICATION EN DJANGO 3](#_Toc184126810)

[a) projet de gestion d’une bibliothèque 3](#_Toc184126811)

[b) Rédaction du projet Conception et Mise en Œuvre d'une Base de Données avec MySQL 3](#_Toc184126812)

[ Contexte 3](#_Toc184126813)

[ Étape 1 : Analyse des Besoins 3](#_Toc184126814)

[ Étape 2 : Conception de la base de donnée 4](#_Toc184126815)

[II. 2- PROJET PROTECTION DE DONNEES BIBLIOTHEQUE : 9](#_Toc184126816)

[III. 3-PROJET XML: FAITES LA DTD ET LE XML-SCHEMA DE LA BASE DE DONNEES DE VOTRE APPLICATION WEB 11](#_Toc184126817)

[1) Structure de base de données : 11](#_Toc184126818)

[2) DTD (Document Type Définition) : 11](#_Toc184126819)

[3) XML Schéma (XSD) : 13](#_Toc184126820)

# PROJET WEB: MON APPLICATION EN DJANGO

## projet de gestion d’une bibliothèque

le projet consiste à développer une application permettant de gérer efficacement les livres, les emprunts et les retours, tout en assurant le suivi des abonnés.

## Rédaction du projet Conception et Mise en Œuvre d'une Base de Données avec MySQL

### Contexte

Nous sommes chargé de concevoir et de mettre en œuvre une base de données pour une petite bibliothèque en ligne. La bibliothèque doit stocker des informations sur les livres, les auteurs, les membres, et les emprunts.

### Étape 1 : Analyse des Besoins

* + Exigences Fonctionnelles

Gestion des livres : Ajouter, modifier, supprimer et consulter des informations sur les livres.

Gestion des auteurs : Ajouter, modifier, supprimer et consulter des informations sur les auteurs.

Gestion des membres : Ajouter, modifier, supprimer et consulter des informations sur les membres.

Gestion des emprunts : Permettre aux membres d'emprunter des livres et enregistrer les emprunts.

* + Exigences Non-Fonctionnelles

Performance : La base de données doit être performante pour des opérations fréquentes de lecture et écriture.

Sécurité : Les données doivent être protégées contre les accès non autorisés.

Fiabilité : La base de données doit garantir l'intégrité des données.

### Étape 2 : Conception de la base de donnée

**-- Création de la base de données**

CREATE DATABASE IF NOT EXISTS LivreAppDB;

USE LivreAppDB;

**-- Création de la table des utilisateurs**

CREATE TABLE Utilisateurs (

id INT AUTO\_INCREMENT PRIMARY KEY,

nom VARCHAR(50) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

role ENUM('admin', 'utilisateur') DEFAULT 'utilisateur',

date\_creation TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

**-- Création de la table des livres**

CREATE TABLE Livres (

id INT AUTO\_INCREMENT PRIMARY KEY,

titre VARCHAR(255) NOT NULL,

auteur VARCHAR(100) NOT NULL,

date\_publication DATE,

utilisateur\_id INT,

date\_ajout TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (utilisateur\_id) REFERENCES Utilisateurs(id) ON DELETE SET NULL

);

**-- Création de la table des commentaires**

CREATE TABLE Commentaires (

id INT AUTO\_INCREMENT PRIMARY KEY,

contenu TEXT NOT NULL,

livre\_id INT,

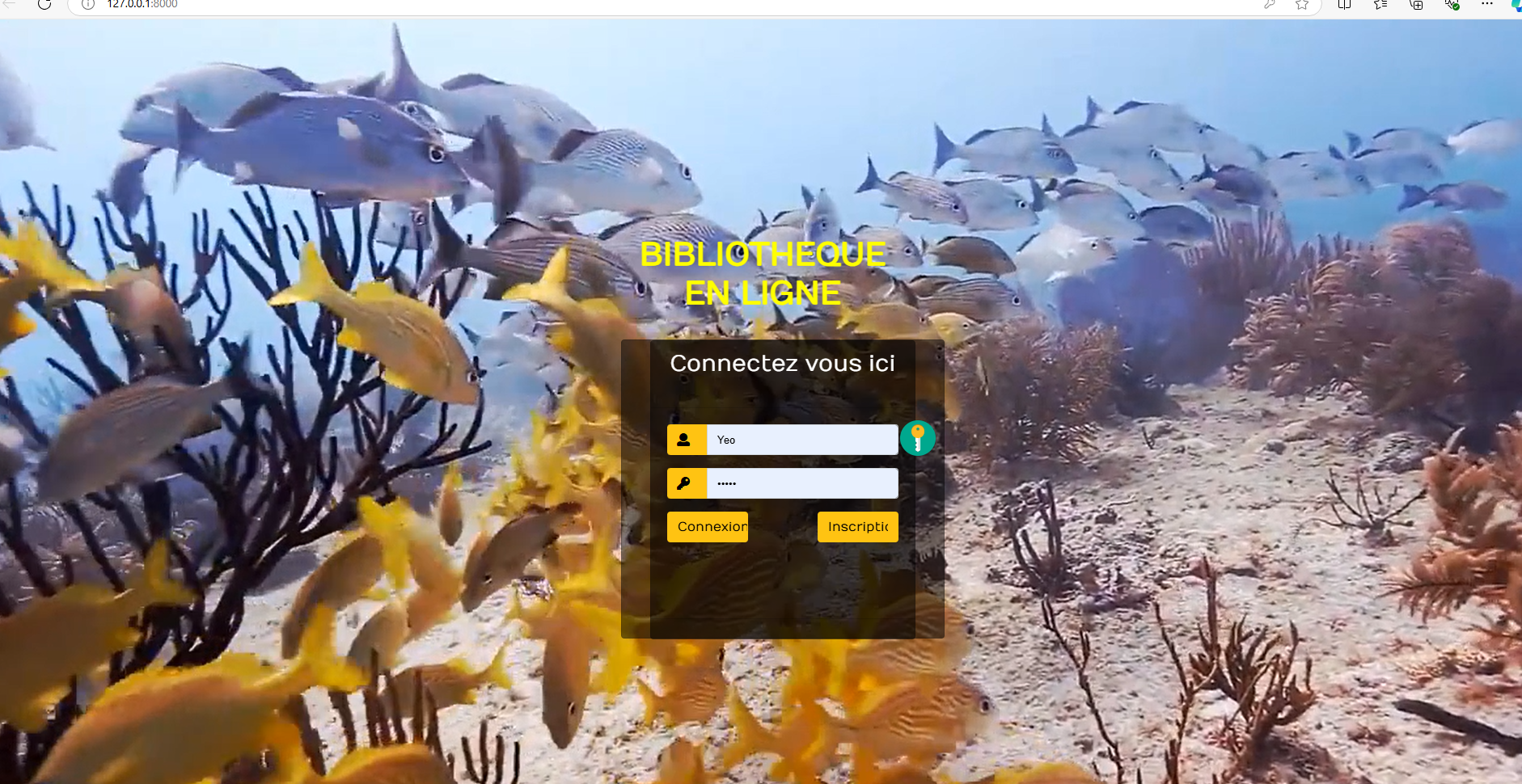
utilisateur\_id INT,

date\_commentaire TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (livre\_id) REFERENCES Livres(id) ON DELETE CASCADE,

FOREIGN KEY (utilisateur\_id) REFERENCES Utilisateurs(id) ON DELETE CASCADE

);

Figure 1Interface de Connexion

Une image contenant capture d’écran, texte, logiciel, Logiciel multimédia

Description générée automatiquementFigure 2Interface du Tableau de bord

Figure 3 Interface d'Ajout de livreUne image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Une image contenant texte, logiciel, Icône d’ordinateur, Page web

Description générée automatiquement

Figure 4 Livres Ajoutés

# 2- PROJET PROTECTION DE DONNEES BIBLIOTHEQUE :

**-- Création d'un trigger pour mettre à jour la table des utilisateurs lors de l'insertion d'un livre**

DELIMITER //

CREATE TRIGGER before\_livre\_insert

BEFORE INSERT ON Livres

FOR EACH ROW

BEGIN

DECLARE msg VARCHAR(255);

IF NEW.utilisateur\_id IS NULL THEN

SET msg = 'Erreur : Un livre doit être associé à un utilisateur.';

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = msg;

END IF;

END;

//

DELIMITER ;

-- Trigger pour historiser la suppression d'un livre

CREATE TABLE Historique\_Suppressions (

id INT AUTO\_INCREMENT PRIMARY KEY,

titre VARCHAR(255),

auteur VARCHAR(100),

date\_suppression TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

DELIMITER //

CREATE TRIGGER after\_livre\_delete

AFTER DELETE ON Livres

FOR EACH ROW

BEGIN

INSERT INTO Historique\_Suppressions (titre, auteur)

VALUES (OLD.titre, OLD.auteur);

END;

//

DELIMITER ;

Une image contenant texte, capture d’écran, logiciel, nombre

Description générée automatiquement

Figure 5Base de donnée Bibliothèque

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

Figure 6 Trigger Bibliothèque

# 3-PROJET XML: FAITES LA DTD ET LE XML-SCHEMA DE LA BASE DE DONNEES DE VOTRE APPLICATION WEB

# Structure de base de données :

* **Livres** : titre, auteur, genre, année de publication.
* **Utilisateurs** : nom, prénom, email, rôle (admin, utilisateur).
* **Emprunts** : utilisateur, livre, date d'emprunt, date de retour.

# DTD (Document Type Définition) :

<!DOCTYPE Bibliotheque [

<!ELEMENT Bibliotheque (Livres, Utilisateurs, Emprunts)>

<!ELEMENT Livres (Livre+)>

<!ELEMENT Livre (Titre, Auteur, Genre, Annee)>

<!ELEMENT Titre (#PCDATA)>

<!ELEMENT Auteur (#PCDATA)>

<!ELEMENT Genre (#PCDATA)>

<!ELEMENT Annee (#PCDATA)>

<!ELEMENT Utilisateurs (Utilisateur+)>

<!ELEMENT Utilisateur (Nom, Prenom, Email, Role)>

<!ELEMENT Nom (#PCDATA)>

<!ELEMENT Prenom (#PCDATA)>

<!ELEMENT Email (#PCDATA)>

<!ELEMENT Role (#PCDATA)>

<!ELEMENT Emprunts (Emprunt+)>

<!ELEMENT Emprunt (Utilisateur, Livre, DateEmprunt, DateRetour)>

<!ELEMENT DateEmprunt (#PCDATA)>

<!ELEMENT DateRetour (#PCDATA)>

]>

# XML Schéma (XSD) :

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="Bibliotheque">

<xs:complexType>

<xs:sequence>

<xs:element name="Livres">

<xs:complexType>

<xs:sequence>

<xs:element name="Livre" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="Titre" type="xs:string"/>

<xs:element name="Auteur" type="xs:string"/>

<xs:element name="Genre" type="xs:string"/>

<xs:element name="Annee" type="xs:int"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="Utilisateurs">

<xs:complexType>

<xs:sequence>

<xs:element name="Utilisateur" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="Nom" type="xs:string"/>

<xs:element name="Prenom" type="xs:string"/>

<xs:element name="Email" type="xs:string"/>

<xs:element name="Role" type="xs:string"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="Emprunts">

<xs:complexType>

<xs:sequence>

<xs:element name="Emprunt" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="Utilisateur" type="xs:string"/>

<xs:element name="Livre" type="xs:string"/>

<xs:element name="DateEmprunt" type="xs:date"/>

<xs:element name="DateRetour" type="xs:date"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>