

## ▀ ▀ ▀ Modèle de copie :

**Créer et administrer une base de données**

**GDWFSCAUBDDEXAIII1A**

**Ceci est un modèle de copie. N’oubliez pas de renseigner vos prénom/nom, ainsi que le nom et le lien vers le projet.**

**Vous pouvez bien sûr agrandir les cadres pour répondre aux questions sur la description du projet si nécessaire.**

**Prénom :**Adèle

**Nom :** Dubois

**ATTENTION ! PENSEZ À RENSEIGNER VOS NOM ET PRÉNOM DANS LE TITRE DE VOS FICHIERS / PROJETS !**

Nom du projet : Évaluation 5

Lien GitHub du projet : ……………

**Description du projet**

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions. Dans cette rubrique, le jury cherche à voir comment vous procédez : comment vous organisez votre travail, comment vous réalisez concrètement la tâche ou l’opération pas à pas.

Utiliser un langage professionnel. Employez-le « je », car vous parlez en votre nom. Vous pouvez écrire au temps présent.

Pour commencer, j’ai planifié ce projet grâce à un tableau Kanban.

Afin de déterminer toutes les fonctionnalités de l’application, j’ai réalisé une analyse du cahier de charges.

Pour cela, j’ai utilisé 3 diagrammes de comportement :

- le diagramme de cas d’utilisations afin de lister les fonctionnalités de l’application :



- le diagramme d’activité pour décomposer les étapes d’une fonctionnalité :



- Diagramme de séquence qui illustre les intéractions entre les différents acteurs :



J’ai ensuite défini l’architecture du code grâce aux spécifications techniques et notamment au diagramme de classe.

Le diagramme de classes m’a permis de représenter les différentes classes, leurs attributs et leurs méthodes ainsi que les relations entre elles.



J’ai ajouté une classe avec les droits des utilisateurs (client, vendeur et administrateur) pour sécuriser l’application.

J’ai ensuite converti le diagramme de classe en modèle relationnel :

CREATE TABLE users (

id INT(11) NOT NULL AUTO\_UNCREMENT PRIMARY KEY,

firstName VARCHAR (50) NOT NULL,

lastName VARCHAR (50) NOT NULL,

birthday DATE NOT NULL,

status VARCHAR (50),

email VARCHAR (50) NOT NULL,

password VARCHAR (50) NOT NULL,

) ;

…

Pour créer la base de données, j’ai utilisé le serveur local XAMP qui m’a permis d’avoir accès à MySQL.

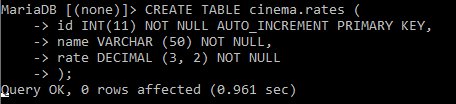
Je me suis ensuite connectée à MySQL :



Et j’ai créé la base de données cinéma :



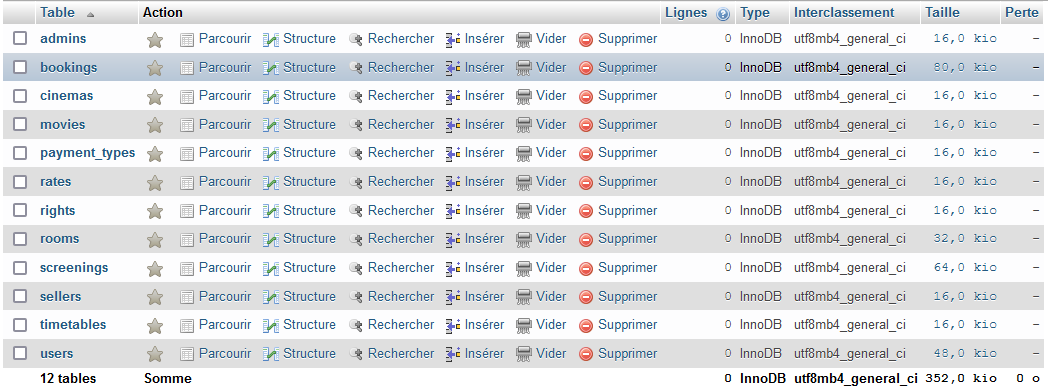
Puis j’ai créé les tables :



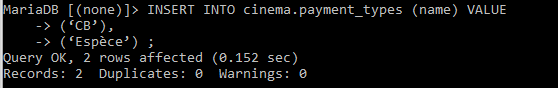
Ainsi que les contraintes :



Pour obtenir les 12 tables :



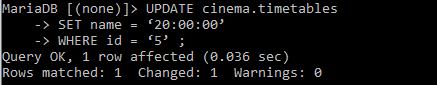
J’ai ensuite alimenté la base de données :



Pour l’identifiant, j’ai préféré utiliser un entier auto-incrémenté plutôt qu’un UUID, pour Universally Unique IDentifiers. En effet, il s’agit d’un petit projet où le risque de conflit est faible. L’inconvénient reste la prédictibilité des identifiants.

Les mots de passe ont été chiffrés pour sécuriser les comptes utilisateurs.

J’ai modifié des données :



Pour finir, j’ai sauvegardé la base de données avec l’utilitaire mysqldump :



Il permet également de restaurer une base de données :



1. Précisez les moyens utilisés. Expliquez tout ce dont vous avez eu besoin pour réaliser vos tâches : langages de programmation, frameworks, outils, logiciels, documentations techniques, etc...

J’ai utilisé le site de Trello qui m’a permis de réaliser un tableau Kanban afin de planifier le projet.

J’ai utilisé le serveur local XAMP afin de pouvoir utiliser le SGBDR MariaDB.

Le terminal CMD de Windows m’a permis d’alimenter la base de données grâce au langage SQL.

Pour la documentation, j’ai utilisé les cours Studi.

1. Contexte. Les noms des organismes, entreprises ou associations, dans lesquels vous avez exercé vos pratiques

NB : Pour le cas des exercices et évaluations demandées sur la plateforme Studi, il s'agit de...Studi.

Ce projet a été réalisé pour Studi.

L’objectif était de modéliser une BDD afin de mettre en place un logiciel de réservation.

Les éléments attendus étaient :

- un diagramme de classe UML ou un Modèle Conceptuel de Données

- un fichier SQL de la retranscription de la base de données dans un SGBD

- un export de la BDD

Le commanditaire était un cinéma.

Les informations que nous possédions sur le commanditaire étaient :

- il y a plusieurs cinémas

- chaque cinéma à plusieurs salles

- chaque salle dispose d’un nombre de place déterminé

- le placement est libre

- un film peut être diffusé dans plusieurs salles aux mêmes horaires

- le tarif diffère en fonction des clients (âge et statut)

- à terme le paiement pourra se faire en ligne

- les séances sont ajoutées par un utilisateur dans chaque cinéma

- des administrateurs seront nécessaires

1. Informations complémentaires (*facultatif*)