

Projet de Groupe en Go :

Service de réservation en ligne

- [Objectif](#)
- [Format du Projet](#)
- [Description détaillée et étapes du projet](#)
 - [Interface Utilisateur en CLI](#)
 - [Menu principal](#)
 - [Interaction avec l'utilisateur](#)
 - [Navigation](#)
 - [Fonctionnalités du menu](#)
 - [Gestion des réservations](#)
 - [Structures de données](#)
 - [Lister les salles disponibles](#)
 - [Créer une réservation](#)
 - [Annuler une réservation](#)
 - [Visualiser les réservations](#)
 - [Base de Données](#)
 - [Structure de la Base de Données](#)
 - [Opérations de Base de Données](#)
 - [Persistance et configuration des identifiants](#)
 - [Export des réservations](#)
 - [Export JSON](#)
 - [Export CSV \(Bonus\)](#)
 - [Interface Web](#)
 - [Mise en place d'un Serveur Web](#)
 - [Création des handlers](#)
 - [Template HTML](#)
 - [Bonus](#)
 - [Fonctionnalités supplémentaires ou créatives au-delà des exigences de base.](#)
 - [Suggestions de conception](#)
 - [Bonnes Pratiques en Go](#)
 - [Modularité et Structure du Code](#)
 - [Librairies Standard Utiles](#)
- [Récapitulatif](#)
 - [Récapitulatif des Étapes Clés](#)
- [Rendu et Travail en Groupe](#)

Objectif

Développer un système de réservation de salles de classe en Go, en commençant avec une interface en ligne de commande pour gérer les réservations, les salles et les utilisateurs. La partie avancée introduira une interface web pour faciliter l'accès et l'interaction avec le système.

Format du Projet

- Groupe de 2 à 3 étudiants.
- Durée : 1 mois.

Description détaillée et étapes du projet

Interface Utilisateur en CLI

Menu principal

- Créez un menu principal en CLI qui offre des options pour chaque fonctionnalité principale du système de réservation.
- **Exemple de menu :**

Bienvenue dans le Service de Réservation en Ligne

1. Lister les salles disponibles
2. Créer une réservation
3. Annuler une réservation
4. Visualiser les réservations
5. Quitter

Choisissez une option :

Interaction avec l'utilisateur

- Implémentez la gestion des entrées utilisateur pour permettre la navigation et l'exécution des actions dans le menu.
- Assurez-vous que le système gère les entrées invalides et guide l'utilisateur vers des choix corrects.
- **Exemple de gestion des entrées :**
- Si l'utilisateur saisit un chiffre non valide, affichez un message d'erreur et demandez de nouveau une entrée.
- Utilisez des boucles et des conditions pour gérer la navigation entre les menus.

Navigation

- Permettez à l'utilisateur de naviguer facilement entre les différentes fonctionnalités. Par exemple, après avoir visualisé ou réservé une salle, l'utilisateur doit pouvoir retourner au menu principal ou effectuer une autre action sans redémarrer le programme.
- **Exemple de navigation :**
- Après l'affichage d'une réservation :

1. Retourner au menu principal
2. Quitter

Choisissez une option :

Fonctionnalités du menu

- **Lister les salles disponibles** : Créez une fonction qui affiche les salles qui ne sont pas réservées pour une plage horaire spécifique.
- **Créer une réservation** : Demandez à l'utilisateur de fournir les détails nécessaires pour une réservation (nom de la salle, date, heure) et vérifiez si la salle est disponible avant de créer la réservation.
- **Annuler une réservation** : Demandez à l'utilisateur de fournir les détails nécessaires pour annuler une réservation (nom de la salle, date, heure) et vérifiez si la réservation existe avant de l'annuler.
- **Visualiser les réservations** : Affichez les réservations existantes pour une salle spécifique, permettez de filtrer pour une date spécifique.

Gestion des réservations

Structures de données

- Définissez une structure `Room` qui représente une salle de classe, incluant des attributs comme l'identifiant de la salle, le nom, et la capacité.
- Créez une structure `Reservation` qui contient les informations d'une réservation, comme l'identifiant de la réservation, l'identifiant de la salle, la date et l'heure.

Lister les salles disponibles

- Créez une fonction qui prend en entrée une date et une heure, et retourne une liste des salles disponibles pour cette plage horaire.
Une salle est **disponible** si elle n'a pas de réservation pour la date et l'heure demandées.
- Utilisez des boucles et des conditions pour filtrer les salles réservées.
- **Exemple de liste de salles :**

Salles disponibles pour le 2023-04-01 à 14:00 :

1. Salle 1 (Capacité : 30)
2. Salle 2 (Capacité : 40)
3. Salle 3 (Capacité : 50)

Créer une réservation

La fonction `CreateReservation` sera responsable de créer une nouvelle réservation. Voici les étapes détaillées :

- **Entrée Utilisateur :** Demandez à l'utilisateur de fournir les informations nécessaires pour une réservation, telles que l'ID de la salle, la date et l'heure de début et de fin.
- **Vérification de la Disponibilité :** Avant de créer la réservation, vérifiez si la salle est disponible en utilisant une fonction (ex : `IsRoomAvailable`). Cette fonction parcourt toutes les réservations existantes pour la salle en question et vérifie s'il y a un chevauchement avec la nouvelle réservation demandée.
- **Création de la Réservation :** Si la salle est disponible, créez une nouvelle instance de `Reservation` et attribuez-lui les détails fournis. Ajoutez ensuite cette réservation à la base de données.
- **Retour Utilisateur :** Affichez un message de confirmation à l'utilisateur.

Annuler une réservation

La fonction `CancelReservation` permet à l'utilisateur d'annuler une réservation existante :

- **Identifiant de la Réservation :** Demandez à l'utilisateur de fournir l'identifiant de la réservation à annuler.
- **Recherche de la Réservation :** Recherchez dans la base de données des réservations pour trouver la réservation correspondant à l'identifiant fourni.
- **Annulation :** Si la réservation est trouvée, retirez-la de la base de données. Informez l'utilisateur que la réservation a été annulée avec succès. Si la réservation n'est pas trouvée, informez l'utilisateur.

Visualiser les réservations

La fonction `ListReservations` affiche les réservations :

- **Affichage :** Parcourez la base de données des réservations et affichez les détails pour chaque réservation. Vous pouvez offrir des options pour filtrer les réservations par salle ou par date.
- **Exemple de liste de réservations :**

Réservations pour la Salle 1 :

1. 2023-04-01 14:00 - 2023-04-01 15:00
2. 2023-04-02 10:00 - 2023-04-02 11:00

Base de Données

Structure de la Base de Données

- Utilisez une base de données pour stocker les salles et les réservations.
- Vous devez y stocker les informations sur les salles et les réservations, et implémenter des fonctions pour ajouter, supprimer et récupérer des données de la base de données.

Opérations de Base de Données

- **Ajouter une Salle** : Créez une fonction pour ajouter une nouvelle salle à la base de données. Veillez à ce que son ID soit unique (auto-incrémenté).
 - Cette fonction vous aidera à ajouter de nouvelles salles à la base de données.
- **Ajouter une Réservation** : Créez une fonction pour ajouter une nouvelle réservation à la base de données. Veillez à ce que son ID soit unique (auto-incrémenté).
- **Supprimer une Réservation** : Créez une fonction pour supprimer une réservation de la base de données.
- **Récupérer les Salles** : Créez une fonction pour récupérer toutes les salles de la base de données.
- **Récupérer les Réservations** : Créez une fonction pour récupérer toutes les réservations de la base de données.
- **Récupérer les Réservations par Salle** : Créez une fonction pour récupérer toutes les réservations pour une salle spécifique.
- **Récupérer les Réservations par Date** : Créez une fonction pour récupérer toutes les réservations pour une date spécifique.
- **Vérifier la Disponibilité** : Créez une fonction pour vérifier si une salle est disponible pour une plage horaire spécifique.

Persistence et configuration des identifiants

- **Chargement des Identifiants** : Les identifiants de la base de données doivent être chargés depuis un fichier de configuration externe pour permettre des modifications dynamiques sans recompilation.
- **Fichier d'Initialisation de la Base de Données** : Créez un script ou un fichier d'initialisation pour préparer la base de données à son premier usage.

Export des réservations

Export JSON

- Créez une fonction pour exporter les réservations dans un fichier JSON.
- Assurez-vous que l'export JSON reflète fidèlement la structure et l'état actuel de la base de données.

Export CSV (Bonus)

- En tant que fonctionnalité bonus, ajoutez une option pour exporter les données en format CSV, utile pour l'analyse de données ou la migration.

Interface Web

Vous pouvez choisir d'implémenter une interface web pour faciliter l'accès et l'interaction avec le système de réservation.

Mise en place d'un Serveur Web

- Initialisez votre serveur web en Go en utilisant `http.ListenAndServe`.
- Définissez des routes pour gérer les requêtes HTTP pour les différentes fonctionnalités du système de réservation.

Création des handlers

- Pour chaque routes définies, créez des handlers pour gérer les requêtes HTTP.
- Les handlers devraient appeler les fonctions appropriées pour gérer les opérations de réservation.
 - Si vous avez bien codé les fonctions de réservation de la CLI, vous pouvez les réutiliser pour gérer les requêtes HTTP.

Template HTML

- Créez des templates HTML pour afficher les différentes pages de l'interface web.
- Utilisez le système de templating de Go pour intégrer dynamiquement les données dans vos pages HTML.

Bonus

Fonctionnalités supplémentaires ou créatives au-delà des exigences de base.

Suggestions de conception

Bonnes Pratiques en Go

- **Clarté et Simplicité** : Écrivez un code clair et lisible. Go privilégie la simplicité ; évitez les constructions compliquées inutiles.
- **Nommage des Variables et Fonctions** : Utilisez des noms descriptifs et cohérents pour les variables, fonctions et autres identifiants.
- **Commentaires et Documentation** : Documentez votre code avec des commentaires pertinents. Expliquez le 'pourquoi' derrière les choix de conception complexes.

Modularité et Structure du Code

- **Découpage en Packages** : Organisez votre code en packages logiques. Par exemple, un package pour la gestion de la base de données, un autre pour l'interface utilisateur, etc.
- **Fonctions Réutilisables** : Créez des fonctions qui accomplissent des tâches spécifiques et peuvent être réutilisées dans différents contextes du projet.

Librairies Standard Utiles

- **fmt pour l'Affichage et la Saisie** : Utilisez la librairie `fmt` pour afficher des informations à l'utilisateur et lire les saisies.
- **Gestion du Temps avec time** : La librairie `time` est essentielle pour manipuler les dates et les heures, notamment pour la création et le rappel des événements.
- **Manipulation de Fichiers avec os et io** : Pour la lecture et l'écriture des fichiers de configuration, de sauvegarde JSON, et d'export CSV.
- **Encodage JSON avec encoding/json** : Utilisez `encoding/json` pour convertir les données de la base de données en JSON et vice-versa.
- **Serveur Web avec net/http** : La librairie `net/http` est utilisée pour créer un serveur web en Go.

Récapitulatif

Vous avez désormais un aperçu complet du projet de création d'un système de réservation en ligne. Ce projet vous offre l'opportunité de mettre en pratique les compétences acquises en programmation Go, tout en relevant des défis réalistes liés à la gestion de données, à l'interface utilisateur, et à la conception de logiciels.

Récapitulatif des Étapes Clés

1. **Interface Utilisateur en CLI** : Créez un menu interactif pour gérer les réservations, les salles et les utilisateurs.
2. **Gestion des Réservations** : Implémentez les fonctions pour créer, annuler et visualiser les réservations.
3. **Base de Données** : Utilisez une base de données pour stocker les salles et les réservations, et implémentez des fonctions pour ajouter, supprimer et récupérer des données.
4. **Export des Réservations** : Ajoutez des fonctionnalités pour exporter les réservations en JSON et CSV.
5. **Interface Web** : Implémentez une interface web pour faciliter l'accès et l'interaction avec le système de réservation.

Rendu et Travail en Groupe

- Code source complet et commenté.
- Fichier **README** avec instructions, description des fonctionnalités et répartition des tâches.
- Documentation technique décrivant la structure du code, les choix de conception, et la logique.

Assurez-vous que le code est bien organisé, documenté, et testé, avant la soumission finale.

Préparez une présentation détaillée du projet pour la soutenance, en mettant en avant les fonctionnalités implémentées, la collaboration au sein du groupe et les contributions individuelles.