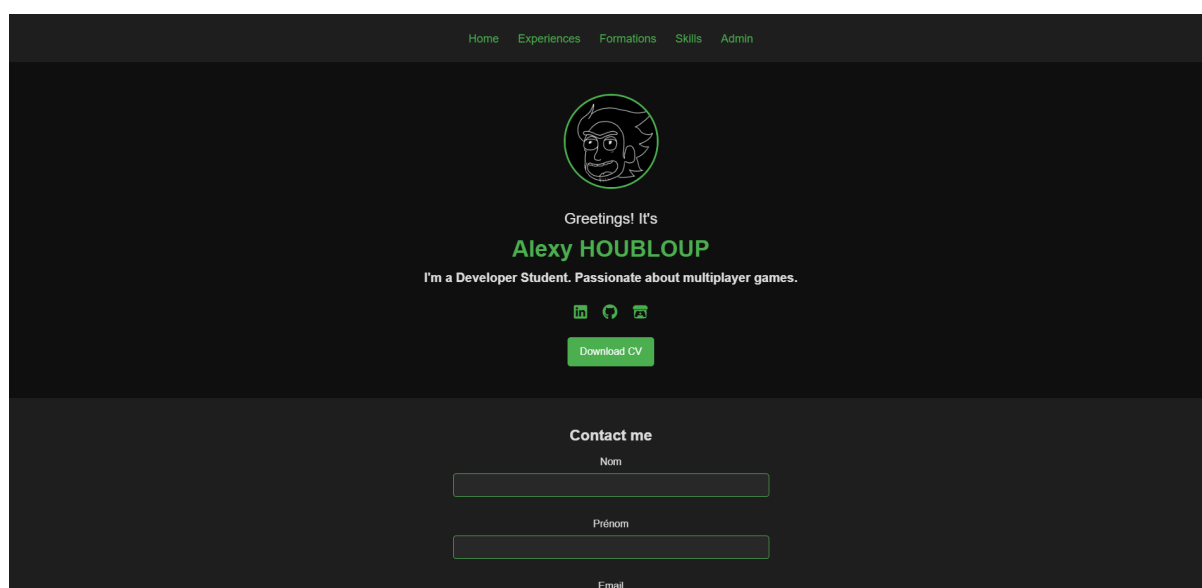


Documentation Technique

Ce document présente la documentation technique de l'application Web développée en Go avec une base de données SQLite et le framework Gin. L'application permet de gérer des données dynamiques telles que les contacts, les formations, les expériences et les compétences via une interface Web



1. Architecture de l'Application

L'application suit une architecture backend utilisant le langage Go et le framework Gin. Elle repose sur une base de données SQLite pour stocker les données. Les templates HTML sont utilisés pour générer les pages dynamiques du frontend.

- Backend: Go avec le framework Gin
- Base de données: SQLite
- Frontend: HTML avec templates

2. Interaction entre Go et HTML

Dans cette application, Go et HTML interagissent principalement à travers l'utilisation de templates. Le langage Go, via le package 'html/template', permet de rendre dynamiquement des pages HTML en injectant des données backend dans des fichiers de template.

Voici le processus de fonctionnement entre Go et HTML:

2.1 . Back-end (Go):

- Go gère les requêtes HTTP et récupère les données nécessaires à partir de la base de données SQLite.
- Les données sont ensuite passées aux templates HTML pour être affichées dynamiquement.

2.2 . Templates HTML:

- Les fichiers HTML sont des templates contenant des espaces réservés pour les données dynamiques.
- Go remplit ces espaces avec les valeurs récupérées depuis la base de données.
- Exemple
 - {{.Title}} ou {{.Description}}

2.3 . Fonctionnalités dynamiques:

- L'injection de données se fait via la méthode ``template.Execute()`` de Go, qui prend en entrée le fichier de template et les données à injecter.
- Cela permet de générer du contenu HTML personnalisé en fonction des informations récupérées par l'application. Ainsi, Go et HTML coopèrent pour générer des pages web dynamiques, où Go s'occupe de la logique backend et des données, tandis que HTML se concentre sur la présentation côté utilisateur.

Choose a category:

Id	Etablissement	Diplome	Date_debut	Date_fin	Description	Actions
1	t	test	test	test	t	

Etablissement

Diplome

Date_debut

Date_fin

Description

Etablissement	Diplome	Date_debut	Date_fin	Description
t	test	test	test	t

3. Documentation de l'API

L'application expose une API simple permettant de récupérer les données par catégorie via une requête GET.

- Endpoint: GET /api/:category
 - Description: Récupérer les données associées à une catégorie (par exemple, 'contact', 'formations').
- Paramètre:
 - category: Nom de la table dans la base de données.
- Réponse:
 - category: Nom de la catégorie
 - columns: Liste des colonnes de la table
 - data: Données contenues dans la table sous forme de tableau

