

Au moment de cliquer sur "Je confirme mon rendez-vous" le site reviendra automatiquement à la page d'accueil. Vous recevrez un mail de confirmation immédiatement.

Rendez-vous*

Coupe Homme 15€

<

December 2023

>

MO	TU	WE	TH	FR	SA	SU
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

12/21/2023

10:00

10:30

11:00

11:30

12:00

12:30

13:00

13:30

14:00

14:30

15:00

15:30

16:00

16:30

17:00

17:30

18:00

18:30

Nom et Prénom :

Numéro de téléphone :

Email :

☐ J'accepte les conditions générales.*

JE CONFIRME MON RENDEZ-VOUS

Documentation stage

Table des matières

1. Introduction

1.1 Objectifs du projet

1.2 Technologies utilisées

2. Mise en place de l'environnement de développement

2.1 Installation de Node.js et npm

2.2 Initialisation du Projet Vue.js

2.3 Configuration de Firebase

3. Conception de la base de données Firebase

3.1 Structuration de la base de données

3.1 Règles de sécurité Firebase

4. Développement de l'application Vue.js

4.1 Création des Composants Vue

4.2 Gestion de l'Authentification avec Firebase Authentication

4.3 Implémentation de l'Interface Utilisateur pour la Prise de Rendez-vous

1. Introduction

1.1 Objectif du projet

Le projet vise à créer un site internet permettant aux clients de prendre des rendez-vous de manière efficace pour un salon de coiffure, améliorant ainsi la gestion des rendez-vous et offrant une expérience conviviale.

2.1 Technologies utilisées

Le site sera développé en utilisant les technologies suivantes :

- HTML
- CSS
- JavaScript
- Vue.js
- Firebase (pour la gestion de la base de données et l'authentification)

2. Mise en place de l'environnement de Développement

L'étape suivante consiste à mettre en place l'environnement de développement pour le projet. Cela inclut l'installation de Node.js et npm, l'initialisation du projet Vue.js, ainsi que la configuration de Firebase.

2.1 Installation de Node.js et npm

Avant de débiter le développement, j'ai assuré l'installation de Node.js et npm sur ma machine. Node.js est un environnement d'exécution JavaScript côté serveur, tandis que npm est le gestionnaire de paquets Node.js. J'ai suivi les instructions sur le site officiel de Node.js pour télécharger et installer ces outils.

2.2 Initialisation du projet Vue.js

L'installation de Vue CLI a été nécessaire pour utiliser des commandes Vue spécifiques. Lors de la création du projet, j'ai suivi les instructions interactives pour personnaliser la configuration de mon projet.

2.3 Configuration de Firebase

Pour intégrer Firebase à mon projet Vue.js, j'ai suivi les étapes suivantes :

- Création d'un projet Firebase : J'ai créé un nouveau projet sur la console Firebase.
- Configuration du projet Vue.js avec Firebase : Après avoir installé le module Firebase dans mon projet Vue.js à l'aide de la commande `npm install firebase`, j'ai configuré Firebase dans le fichier principal du projet. J'y ai ajouté les informations de configuration de mon projet Firebase.

Cela m'a permis de mettre en place Firebase dans mon projet Vue.js, prêt à être utilisé pour la gestion de la base de données et de l'authentification.

3. Conception de la base de données Firebase

Maintenant que l'environnement de développement est configuré, nous allons passer à la conception de la base de données Firebase. Cette étape consiste à définir la structure de la base de données et à mettre en place des règles de sécurité pour assurer un accès approprié aux données.

3.1 Structuration de la base de données

La structuration de la base de données Firebase est cruciale pour garantir une organisation efficace des données. J'ai identifié les entités principales de mon application, telles que les utilisateurs, les rendez-vous, et d'autres informations pertinentes.

Pour chaque entité, j'ai défini les champs nécessaires et établi les relations appropriées entre elles. Par exemple, la table des rendez-vous peut être liée à la table des utilisateurs pour attribuer chaque rendez-vous à un utilisateur spécifique.

3.2 Règles de sécurité Firebase

Afin de protéger mes données, j'ai mis en place des règles de sécurité Firebase. Ces règles définissent qui peut accéder à quelle partie de la base de données et dans quelles conditions. Cela permet d'assurer un accès sécurisé et de prévenir tout accès non autorisé.

Les règles de sécurité Firebase sont définies dans le fichier de configuration Firebase et spécifient les conditions sous lesquelles les données peuvent être lues ou écrites. Par exemple, je peux restreindre l'accès à certaines données uniquement aux utilisateurs authentifiés, assurant ainsi la confidentialité des informations sensibles.

En structurant judicieusement ma base de données et en définissant des règles de sécurité appropriées, je m'assure que mon application fonctionne de manière fiable et sécurisée, offrant une expérience utilisateur optimale tout en préservant l'intégrité des données.

2.

4. Développement de l'Application Vue.js

Nous entrons maintenant dans la phase de développement de l'application Vue.js. Cette étape comprend la création des composants Vue nécessaires, la gestion de l'authentification avec Firebase, et l'implémentation de l'interface utilisateur pour la prise de rendez-vous.

4.1 Création des Composants Vue

Pour structurer mon application, j'ai créé des composants Vue distincts pour chaque fonctionnalité. Par exemple, j'ai créé des composants pour l'affichage des rendez-vous, la gestion du formulaire de prise de rendez-vous, l'authentification, etc.

Cette approche modulaire facilite la maintenance, la réutilisation, et permet une gestion plus claire des différentes parties de l'application. Chaque composant est responsable d'une fonction spécifique, favorisant la lisibilité et la scalabilité du code.

4.2 Gestion de l'Authentification avec Firebase Authentication

L'authentification des utilisateurs est une partie cruciale de mon application. J'ai intégré Firebase Authentication pour permettre aux utilisateurs de s'inscrire, de se connecter, et de gérer leur compte de manière sécurisée.

En utilisant les fonctionnalités fournies par Firebase Authentication, j'ai pu simplifier le processus d'authentification tout en garantissant la sécurité des données sensibles. Cela inclut la gestion des sessions utilisateur, la récupération de mot de passe, et d'autres fonctionnalités liées à l'authentification.

4.3 Implémentation de l'Interface Utilisateur pour la Prise de Rendez-vous

L'interface utilisateur pour la prise de rendez-vous est un élément central de mon application. J'ai conçu une interface conviviale permettant aux utilisateurs de sélectionner des créneaux horaires, de fournir des informations nécessaires, et de confirmer leurs rendez-vous.

En utilisant Vue.js, j'ai implémenté des fonctionnalités dynamiques pour rendre l'expérience utilisateur fluide et interactive. Les composants Vue créés précédemment ont été intégrés dans l'interface pour assurer une structure modulaire et une gestion efficace des états.

Avec ces composants en place, l'utilisateur peut naviguer facilement à travers l'application, créer un compte, se connecter, et planifier ses rendez-vous de manière intuitive.