

“Evil Kanban” - A Vue.js Kanban project

Thomas Dellacase & Maxime Hadj

I - Comparatif de Vue.js, Svelte et Angular

Introduction

Les frameworks et bibliothèques JavaScript jouent un rôle crucial dans le développement web moderne, simplifiant les tâches complexes et améliorant l'efficacité de la création d'interfaces utilisateur interactives et dynamiques. Dans ce document, nous présentons trois frameworks front end populaires : Vue.js, Svelte et Angular. Chacun de ces frameworks a ses forces et faiblesses, répondant à différents besoins et préférences des développeurs.

Vue.js

Vue.js est un framework JavaScript conçu pour la construction d'interfaces utilisateur. Il est conçu pour être adopté de manière progressive, permettant aux développeurs d'intégrer Vue.js facilement dans des projets existants.

Avantages

Courbe d'apprentissage facile : Vue.js a une courbe d'apprentissage douce, le rendant accessible aux débutants. Sa syntaxe est simple et assez facile d'accès.

Flexibilité : Vue.js est flexible et peut être utilisé aussi bien pour des applications monopages que pour des interfaces web complexes.

Polyvalence : Vue.js peut être intégré à d'autres bibliothèques ou projets existants sans complications majeures. Il permet aux développeurs d'adopter progressivement les fonctionnalités de Vue.js.

Inconvénients

Écosystème plus restreint : Bien que Vue.js ait un écosystème en croissance, l'activité au sein de la communauté web étant moins importante que sur des frameworks plus établis, il peut être plus difficile d'accéder à l'information recherchée.

Préoccupations quant à la maturité : Certaines entreprises peuvent exprimer des inquiétudes quant à la maturité de Vue.js par rapport à des frameworks plus établis tels qu'Angular.

Svelte

Svelte est une approche radicalement nouvelle pour la création d'interfaces utilisateur. Il déplace le travail traditionnellement effectué par le navigateur dans une étape de compilation, générant ainsi un code optimisé et efficace.

Avantages

Performance : Svelte génère un code JavaScript hautement optimisé lors de l'étape de compilation. Il en résulte des bundles plus petits et des performances d'exécution améliorées.

Aucun surcoût à l'exécution : Contrairement à Vue.js et React, Svelte déplace le travail du framework vers le temps de compilation, éliminant ainsi tout surcoût à l'exécution pour les utilisateurs finaux.

Simplicité : La syntaxe de Svelte est minimaliste et ses concepts sont faciles à saisir. Il réduit le code superflu et simplifie le développement.

Inconvénients

Nouveau paradigme : Les développeurs habitués aux frameworks traditionnels pourraient trouver l'approche de Svelte peu familière, ce qui peut entraîner une courbe d'apprentissage plus difficile.

Communauté peu développée : Svelte étant un framework plus récent, a une communauté plus petite par rapport à Vue.js ou Angular. Cela peut affecter la disponibilité des frameworks et des ressources.

Angular

Angular est un framework front-end complet développé et maintenu par Google. Il offre une solution complète pour la création d'applications web.

Avantages

Framework complet : Angular est un framework complet avec un ensemble de fonctionnalités puissantes prêtes à l'emploi, notamment l'injection de dépendances, la navigation et la gestion de states.

Intégration de TypeScript : Angular est construit avec TypeScript, ce qui améliore la productivité des développeurs grâce au typage statique et aux fonctionnalités ECMAScript modernes.

Support entreprise : Angular est bien adapté pour les applications à grande échelle et bénéficie d'un solide soutien de la part de Google. Il offre une architecture structurée adaptée au développement à l'échelle de l'entreprise.

Inconvénients

Courbe d'apprentissage abrupte : Angular a une courbe d'apprentissage plus élevée par rapport à Vue.js et Svelte. Ses fonctionnalités et concepts étendus peuvent être intimidants pour les débutants.

Code très verbeux : Les applications Angular peuvent contenir une quantité considérable de code redondant, ce qui peut ralentir le développement et augmenter la complexité de la base de code.

Taille du framework : Les applications Angular ont tendance à avoir des tailles de bundle plus importantes par rapport à Vue.js et Svelte, ce qui peut affecter les temps de chargement globaux.

II - Choix de la technologie

Ce document détaille les raisons du choix de la technologie Vue.js comme framework pour le développement de notre application Kanban. Cette décision est cruciale pour la réussite de notre projet, et je pense qu'il est important que tout le monde comprenne la logique qui est à l'origine de choix..

Evil Kanban est une application d'organisation de tâches. Afin de réaliser ce projet, nous avons choisi d'utiliser le framework Vue.js. En effet, les applications de "Todo list" étant généralement peu gourmandes en ressources, Vue.js nous a semblé être la solution la plus adaptée au développement de ce projet. En effet, la facilité de prise en main ainsi que la flexibilité du framework sont les raisons majeures qui constituent ce choix, de plus, l'architecture basée sur les composants nous est familière. Vue.js est donc selon nous un bon compromis entre efficacité et flexibilité, et le bon choix afin de mener à bien ce projet.