Projet APNOTAN

Introduction/Présentation :

Ce document a pour but de décrire le déroulement du prototype de notre projet informatique portant sur un programme la mise en place application Web permettant l’évaluation de films.

Ce rapport est le résultat des recherches qui nous a permis de réaliser les prémices de ce projet.

Il contient l’ensemble des éléments de debut de realisation de notre projet.

Tout d’abord, nous présenterons le projet qui nous a été attribuer .

Nous expliquerons également l’utilité du choix d’un VPN.

Par La suite nous verrons le diagramme des cas d’utilisation

Nous explorons aussi le choix de notre support de réalisation ( vue Js )

Puis pour finir nous dévoilerons une première analyse de notre API REST

Nous espérons que vous prendrez autant de plaisir à lire ce rapport que nous en avons pris à le faire.

Note Sur le Choix d’un VPN :

Sans VPN, les données qui sont envoyé via Internet (surtout via les sites web qui ne respectent pas la norme HTTPS) sont visibles en clair et donc sans protection. Les pirates peuvent visualiser les données en transit et remonter leur piste pour identifier et accéder à l’adresse IP. Cela peut affecter directement l’entreprise si ces données incluent des informations confidentielles .

Indirectement, les utilisateurs du panel peuvent rendre vos serveurs et vos appareils vulnérables pour lancer des cyberattaques.

Choix Technique de Node/Vue :

Vue.js est facilement adaptable et peut être facilement intégré aux grands projets de développement de première ligne sans aucune difficulté. Sans plugins supplémentaires et bibliothèques externes, Vuejs aide à coder des applications robustes. De plus, il offre aux développeurs de bibliothèque de documents énorme, rendant ainsi le processus de développement d’applications Vue.js immensément plus facile.

Il est très flexible car il peut être utilisé dans n’importe quel environnement

Il offre aux développeurs une intégration facile avec des applications déjà existantes. Car il est basé sur le Framework JS.

En raison de son codage simple et facile, il conduit à une meilleure maintenabilité du code qui, en fin de compte, réduit les bugs et finalement nécessite moins d’heure de développement dans la correction de ces derniers , ce qui représente un gain de temps précieux au sein de projets ou le temps est l’une des principale contrainte

Analyse API REST:

Structure du JSON :

Get https://www.notreapplication.fr/liste

data {

Films {

“Nom Film” {

Title : “...”

Release Date : “...”

Auteur : “...”

Note Critique : “...”

Note Critique haute : “...”

Note Critique basse : “...”

etc…

}

“Nom film 2” {...}

}

Notre Api sera basée sera celle de <https://www.themoviedb.org/documentation/api>

Qui permet une utilisation par tous particulier comme professionnel

Et recense un nombre de film conséquent.

Diagramme UML des USE CASES:

FILMOTEC

<<Include>>

Afficher mensuellement une liste des films les mieux notés

Consulter la liste des films et leurs critiques

<<Include>>

S’authentifier

<<Include>>

Gérer les critiques

Membre du panel

<<Include>>

Récupérer les données des films depuis l’API

Récupérer les critiques

**Application APMAGWEB**

**Application APNOTPAN**

Utilisateurs

APMAGWEB