

M1IF 13 – Programmation Web avancée et mobile Contrôle continu terminal

Durée : 1 heure 00 – Documents, ordinateurs, calculatrices et téléphones portables interdits

Identification

Anonymisez votre copie, et remplissez les parties « Diplôme », « Épreuve » et « Date ».

Ensuite, vous écrirez directement vos réponses sur cet énoncé.

Numéro de copie :

Cours (16 points)

1. Quel est le principal problème que l'on peut avoir pour débugger une application Web générée avec WebPack et comment s'en prémunir ? Pourquoi peut-on avoir le même problème avec Web Assembly ? (2 pts.)
Le code est minifié et il est donc impossible de trouver, en regardant les dev tools, quels sont les fichiers / lignes des sources où se trouvent les problèmes... À moins de mapper le code source sur le code minifié. De la même façon, les modules Web Assembly sont compilés et chargés côté client en bytecode.
2. Listez et expliquez à quoi servent les 3 grandes fonctions du Watcher de VueJS. (2 pts.)
Écouter les notifications de changements du modèle ; pousser ces changements aux directives concernées ; gérer les dépendances entre éléments du modèle (le changement d'une variable peut déclencher celui d'une autre).
3. Expliquer dans quel sens se fait le one-way data binding, et pourquoi il peut être intéressant de faire du one-way data-binding plutôt que du two-way data-binding ? (2 pts.)
Sens : données vers interface ; permet de garder la trace des changements appliqués sur les données, meilleure gestion de l'asynchrone et de la concurrence. Pas de synchronisation « magique ».
4. Donnez un exemple de code malicieux que pourrait renvoyer le serveur suite à une requête cross-domaine en JSONP et expliquez le processus qui fait que ce code serait forcément exécuté côté client. (2 pt.)
N'importe quel code JS (par exemple window.close();) peut être exécuté, puisque ce code se retrouverait dans un élément script de la page, et donc interprété par le navigateur.
5. Dans l'approche « Mobile First », on recommande l'utilisation des media queries de manière « ascendante ». Donnez un exemple pour illustrer ce point. (2 pts.)
*p.details {visibility : hidden;}
@media screen and (min-width: 500px) {
p.details {visibility : visible;}}*
6. Expliquer techniquement pourquoi l'utilisation d'event listeners et de callbacks ne permet pas réellement de faire du multitâche. (2 pts.)
L'event listener rajoute un message à la file demandant l'exécution du callback. Son exécution commencera lorsque l'event loop aura exécuté toutes les tâches insérées dans la file précédemment et se fera avant toute tâche insérée ultérieurement.
7. Expliquez ce qu'apportent les service workers aux Progressive Web Apps et à quel besoin ils répondent en comparaison aux applications natives. (2 pt.)
Il permet de contrôler le cache de l'application (app shell et données), ce qui permet de redémarrer l'application sans la recharger ou en offline, comme lorsqu'elle est installée en local.
8. Écrivez la commande NPM qui permet de rajouter la version 3.1.0 (ou une version mineure supérieure) d'eslint au bon endroit dans le fichier ./package.json de votre projet Node. (2 pt.)
npm install eslint@^3.1.0 --save-dev

9. Quelle est la différence entre le store Vuex et LocalStorage (1 pt.)

L'un (LocalStorage) est persistant et pas l'autre.

Outils (4 points)

10. Durant les cours et les exposés, un assez grand nombre d'outils ont été abordés. Remplissez le tableau ci-dessous (la première ligne est un exemple) :

Nom	Type d'outil	Langage principal	Spécificité technique 1	Spécificité technique 2
Vue	Framework	JS	MVVM	Gestion avancée des états avec Vuex
Angular	Framework	TypeScript	IoC	Material design
Phaser	Game framework	JS ou TypeScript	WebGL, Canvas	Moteur physique, sprites...
React	Bibliothèque	JS	Programmation réactive	Declarative UIs
Lighthouse	Browser plugin / NPM module	JS	Analyse de qualité / performance	Outil de reporting intégré
PlayCanvas	Game engine	JS	WebGL	Publication sur le store PlayCanvas / outils de collaboration (payants)
Node	Plateforme d'exécution	JS	Event loop	Structure de modules arborescente (CommonJS)
Modernizr	Bibliothèque	JS	Détection des capacités d'un device / navigateur	Utilisation de classes de style CSS
ESLint	Linter / outil d'aide au développement	JS	Fonctionne à l'aide de règles (configurable)	Module NPM / CLI