**Cahier des charges du travail de diplôme**

**Étudiant :**  Maitre Nicolas

Email : [nicolas.maitre@cpnv.ch](mailto:nicolas.maitre@cpnv.ch)

**Entreprise : WorkStreams**

**Intitulé du sujet : Développement d’une nouvelle interface web et environnement de développement pour la plateforme IntrepidKnowledge.**

**Responsable :**  Fabrice Roublot

Email : fabrice.roublot@workstreams.ch

**Responsable CPNV :**  Nom Prénom

Email :       Tél :

**1er expert externe :**  Alan Pilloud

Email : alan@bureauweb.ch

**2ème expert externe :**  Mentor Reka

Email : reka.mentor@gmail.com

**Dates :** du 3 mai au 18 juin 2021

**Sujet du travail de diplôme**

Cadre du travail :

Le travail se déroulera au sein de WorkStreams, une entreprise spécialisée dans le développement d’outils de formation informatique pour les entreprises. L’entreprise est composée de M. Fabrice Roublot, responsable du développement, Mme. Myriam Meuwly, responsable de la création d'expériences et apprentissages et Pipa 🐶, responsable du bien être dans l’entreprise.

Besoins du client :

Actuellement la plateforme d’e-learning (IntrepidKnowledge) est basée sur un Framework in-house écrit en JavaScript pur ES5 n’utilisant aucune librairie.   
Ce Framework a été écrit sur une durée d’environ 10 ans ce qui veut dire que certaines implémentations ne sont plus efficientes (principalement dû à la compatibilité requise avec IE11).  
L’environnement de développement est actuellement assez simple, ne contenant pas de tests ni de procédures de déploiement continu.

Lors du déploiement, il n’y a pas de solutions de monitoring d’erreurs côté client.

Le but serait de réécrire l’interface de la plateforme de manière plus moderne avec du JavaScript actuel tout en gardant une compatibilité avec IE11.   
La réécriture devrait également permettre de mettre en place un meilleur environnement de développement. Ce travail ne prend pas en compte la réécriture de l’API http.

Solution :

L’idée est d’utiliser ReactJS pour ré implémenter l’interface. Les applications React pouvant être transpilées avec Webpack/Babel, la compatibilité avec IE11 sera majoritairement préservée.   
NextJS et TypeScript seront utilisés afin de bénéficier d’une meilleure structure et expérience de développement.

Des tests unitaires / d’intégration seront intégrés dans le flux de développement.

En production, une solution de monitoring d’erreurs côté client sera intégrée.

**Tâches à réaliser**

Dans le temps imparti, le candidat devra procéder aux tâches suivantes :

- Conception de l’architecture de l’app en NextJS :

- Routage cohérent

- Conception compatible avec le multi-instance (réutilisation des components existants).

- Traductions de l’interface supportées

- Utilisation de TypeScript

- Intégration continue

- Tests unitaires / d’intégration

- Déclenchement automatique par GitLab

- Déploiement automatique sur le serveur de production

- Solution de remontée d’erreurs client

- Création d’une page de l’application. Page « Groupes »

- Affichage de la liste des groupes depuis l’API

- Affichage des groupes et sous-groupes dans le menu de gauche.

- Ajout d’un groupe (Donc système de fenêtre et formulaire)

- Sélection d’un groupe dans la liste (donc système de barre d’action contextuelle)

- Modification d’un groupe

- Suppression d’un groupe

- Wiki pour les développeurs suivants

Tâches à réaliser si le temps le permet :

- Instances

- Création d’une seconde instance contenant un affichage différent des groupes.

- Outil de construction d’instance en ligne de commande avec paramétrage (couleurs, pages disponibles).

- Création de la page « Groupe »

- Affichage des informations

- Réutilisation des données récupérées dans la page « Groupes »

- Affichage des messages postés dans le groupe

- Système de Tabs

- Affichage et modification des sous-groupes.

- Intégration de la page « Actualités » dont le contenu peut être pré-généré car commun à tous les utilisateurs.

- Intégration du calendrier (projet parallèle de diplôme de Diogo)

- Affichage des documents (donc création d’un fil-d’Ariane)

**Prérequis et matériel à disposition**

Pré-requis :

- API Intrepid

- Librairie d’icônes Intrepid et designs d’interface.

- Connaissances en ReactJS

- Connaissances en technologies web (HTML, JS, CSS)

Matériel à disposition :

- PC ou Mac

- VSCode et Postman

- Système de versionning (GitLab)

- Serveur Debian de développement

**Critères d’évaluation du travail**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Lieu et date :** |  | **Signature :** |
| **L’étudiant :** | , le |  |  |
| **Le chef de projet :** | , le |  |  |
| **Responsable école :** | , le |  |  |
| **1er expert externe :** | , le |  |  |
| **2ème expert externe :** | , le |  |  |