**Bigscreen**

**Cahier des charges:**

Nom du projet: Bigscreen

Date : 12/09/2022 au 10/10/2022

Noms: FUSTIER Julien et MARCEAU Cyril

**Analyse Client:**

Bigscreen est une entreprise travaillant dans la VR qui est un secteur en forte expansion et à besoin d’avoir un outil qui leur permettrait de mieux cibler leur clientèle d’où la réalisation d’une application de sondage leur permettant de connaître leur attente.

**Choix technologiques:**

* Back-end: Concernant le back-end nous avons développé une API rest avec laravel avec l’ORM eloquent.
* Front-end: Pour le front nous avons fait le choix d’utiliser [next.js](https://nextjs.org/) car ce framework front permet d’intégrer plus facilement les différentes routes de l’application, des middlewares ainsi que des gains de performance et de sécurité comparé à une application react.js ou l'on dois tout développer à la main.

**Github:**

<https://github.com/cyrilmarceau/Bigscreen>

**Déploiement:**

Changer les variables d’environnements:

* (DB\_USERNAME et DB\_PASSWORD)

Installation des dépendances:

* $ cd api && composer install
* $ cd admin && npm i

Lancer les migrations:

* php artisan migrate:fresh –seed

Lancer le serveur de développement

* php artisan serve

Identifiant de connexion:

* Email: [admin@admin.fr](mailto:admin@admin.fr)
* Mot de passe: password

**Méthode de travail:**

Nous utiliserons la méthodologie agile car elle permet de s’adapter à la charge de travail en délivrant un travail fonctionnel en continue et permet de mieux diviser le travail étape par étape.

Pour gérer le versionning nous utilisons la méthodologie git flow. Nous partons d’une branche main qui contient le code finalisé. A partir de celle-ci créons une branche par tâche de travail. Lorsque nous avons terminé la fonctionnalité du code nous faisons un merge dans la branche main. Ceci nous permet de travailler de façon efficace et structurée.

Nous utiliserons la convention de commit (détail [ici](https://www.conventionalcommits.org/en/v1.0.0/)) qui permet d’avoir une vue globale et simple des différents commits réalisés durant ce projet.

**Outils utilisés:**

Concernant les différentes fonctionnalités du projet nous avons utilisés [jira](https://www.atlassian.com/fr/software/jira) qui permet d’avoir un aperçus des différents tickets à traiter durant le projet:

[**https://cyrilmarceau.atlassian.net/jira/software/projects/BIG/boards/2**](https://cyrilmarceau.atlassian.net/jira/software/projects/BIG/boards/2)

Pour le versionning nous utilisons [github](https://github.com/cyrilmarceau/Bigscreen).

Pour la mise en forme du schéma de la base de données nous utiliserons DB diagram. Un outil spécialement adapté pour les développeurs qui permet de relier le code avec la représentation visuelle des tables.

**Evaluation du temps de travail**

| **Entrée** | **Sortie (jours)** |
| --- | --- |
| API - Migrations | 0.5 |
| API - Seeders + factories | 0.5 |
| API - Récupérer la liste des sondages | 0.1 |
| API - Récupérer 1 sondage | 0.1 |
| API - Récupérer la liste des questions | 0.1 |
| API - Authentification | 0.2 |

**Liste fonctionnelle**

| **FRONT** | **API** |
| --- | --- |
|  | API - Migrations |
|  | API - Seeders + factories |
|  | API - Récupérer la liste des sondages |
|  | API - Récupérer 1 sondage |
|  | API - Récupérer la liste des questions |
|  | API - Authentification |
|  |  |

**Recettage**

| **Fonctionnalité** | **Status** |
| --- | --- |
| API - Migrations | Fonctionnelle |
| API - Seeders + factories | Fonctionnelle |
| API - Récupérer la liste des sondages | Fonctionnelle |
| API - Récupérer 1 sondage | Fonctionnelle |
| API - Récupérer la liste des questions | Fonctionnelle |
| API - Authentification | Fonctionnelle |
|  |  |
|  |  |

**Documentation:**

**Front:**

**Back:**