

# Projet – Partie 1

Programmation web avancé – IFM26039

## Mise en situation

On veut faire un site web présentant des ou **œuvres culturelles ou traditionnelles**. Cela peut être le site web de présentation d'un musée, un site touristique et culturel, ou encore une application de Collections de Médailles et de Monnaies .

Toute idée similaire sera acceptée. En cas de doute, vous devez vérifier avec votre enseignant.

## Équipe

Le projet se fait en équipe de 2. Certaines équipes de 3 peuvent être acceptées si vous validez avec le professeur.

## Remise

Le projet aura une durée d'environ 4-5 semaines.

À chaque semaine, vous aurez à travailler sur le projet et à montrer les fonctionnalités développées au professeur en classe pour qu'elles soient évaluées et que vous ayez de la rétroaction. Les différents critères seront évalués à chaque semaine et vous serez en mesure de les corriger pour la semaine suivante. Advenant le cas où vous n'avez pas corrigé votre projet en fonction de la rétroaction la semaine après celle-ci, vous serez pénalisés.

La correction se fait à chaque semaine au fur et à mesure que le projet avance. Vous devez donc montrer au professeur ce que vous avez fait à chaque semaine en classe. Vous devez donc vous assurer qu'au moins un des membres de l'équipe est présent et à l'heure en classe à chaque semaine pour présenter votre projet au professeur.

## Design des interfaces

Le projet est dans un but de publication en ligne dans un éventuel portfolio. Je m'attends donc à ce que celui-ci possède une interface graphique ergonomique et professionnelle. Essayez d'être original et de vous démarquer par votre interface graphique.

Votre application doit être responsive, donc bien fonctionner sur un appareil mobile tout en respectant les bonnes normes d'ergonomie d'interface graphique.

## Librairies externes

Si vous voulez utiliser des librairies de code CSS ou Javascript qui ne sont pas montré en classe, consultez le professeur en premier lieu. Certaines librairies pourraient ne pas être autorisées.

L'utilisation d'une librairie de code non autorisée par le professeur pourrait vous donner la note de zéro.

## Fonctionnalités

Votre application Web doit avoir les fonctionnalités suivantes :

- Être créé avec les technologies de React avec Next.js. Les projets utilisant d'autres technologies ne seront pas corrigés.
- Posséder une page d'accueil d'introduction sur votre service ou organisation. Vous pouvez y mettre de l'information générale sur votre organisation fictive. Vous devez obligatoirement avoir au moins une photo et quelques paragraphes de texte. Par exemple, si vous faites un site web pour un groupe de rock en tournée, vous voulez mettre l'information sur le groupe, le type de musique, l'information sur les membres et quelques photos.
- Posséder 2 pages différentes affichant les détails d'un évènement. Chaque page d'évènement doit indiquer son nom, une description, la date et l'heure de l'évènement et contenir quelques photos. Vous pouvez même y mettre des liens ou des téléchargements si vous croyez que cela pourrait être intéressant. Dans le produit final, nous voudrions avoir plus de 2 pages comme celles-ci, mais pour les fins d'évaluation, j'en veux juste 2 pour l'instant.
- Avoir une façon de changer de page, tout en restant sur la même page puisque notre application est une « single page application » (monopage). On veut pouvoir changer entre la page d'accueil et les 2 pages d'évènement. N'utiliser pas de routage client automatisé ici pour l'instant. Utiliser la façon vue en classe.
- Être bien divisé en composants pour favoriser la réutilisation de code.
- Avoir une interface graphique ergonomique et professionnel qui fonctionne aussi bien sur appareil mobile que sur un grand écran.
- Supporter les bonnes pratiques de programmation ainsi que les standard et normes de l'industrie.

Vous pouvez vous créer des pages supplémentaires au besoin, mais les pages indiquées ci-dessus doivent obligatoirement se retrouver dans votre projet.

## Échéancier

Voici un échéancier que vous devez suivre. Vous pouvez prendre de l'avance, mais vous devez montrer au professeur que vous avez complété ou corrigé correctement les éléments nécessaires à chaque semaine.

### Semaine 1

Création du projet, recherche d'information et design

Ce que vous devez faire :

- Créer le projet de base avec Next.js comme vu en classe.
- Mettre le projet dans un repository sur Github, tel que vu en classe.
  - Ajouter tous les membres de votre équipe au repository sur Github.
  - Ajouter le compte de votre professeur comme réviseur du repository sur Github.
- Créer un fichier markdown (.md) nommé « preparation.md » dans un dossier « docs » dans votre projet Next.js.
  - Ce fichier doit contenir une description bien détaillée, mais concise du projet.
  - Ce fichier doit contenir des liens vers des sites web de référence, les couleurs que vous pensez utiliser pour le site web et potentiellement des wireframes de vos interfaces graphiques.

### Semaine 2

Création de la page d'accueil et du « layout » réutilisable pour chaque page.

Ce que vous devez faire :

- Créer le composant de la page d'accueil avec toute l'information nécessaire (voir la section Fonctionnalité).
- Créer le « layout » réutilisable de votre site web (« header », « footer », menu de navigation, etc.) comme vu en classe.
- Mettre la base du style pour la page d'accueil et le « layout ».
- Mettre le projet à jour sur Github.

### Semaine 3

Création des 2 pages ou plus

Ce que vous devez faire :

- Créer les composants des pages de projet avec toute l'information et les images nécessaires.
- S'assurer de créer des composants réutilisables pour éviter la duplication de code entre les 2 pages d'évènement.
- Mettre la base du style pour vos pages de projet.
- Mettre le projet à jour sur Github.

## Semaine 4

Rendre la navigation possible en « single page application » et s'assurer que le projet rencontre les bonnes pratiques et les normes de l'industrie.

Ce que vous devez faire :

- Rendre les liens de votre menu fonctionnel pour changer de page en « single page application » tel que vu en classe.
- Compléter le CSS de votre site web. Assurez-vous que votre page fonctionne bien sur appareil mobile.
- Finir de commenter et d'organiser le code pour qu'il suive les bonnes pratiques.
- Mettre le projet à jour sur Github.

## Grille de correction

Le travail de chaque semaine est pondéré de la façon suivante :

Critère	Pondération
<b>Création du projet, recherche d'information et design</b>	25%
<b>Création de la page d'accueil et du « layout »</b>	25%
<b>Création des 2 pages ou plus</b>	25%
<b>Navigation en « single page application » et bonne pratiques</b>	25%

Voici quelques informations supplémentaires pour la correction :

- Vous perdrez des points dans ces catégories si certaines parties de votre projet ne respecte pas bien les critères et la qualité demandé dans les cours et en classe.
- Pour chaque partie, vous serez aussi évalué sur les bonnes pratiques de programmation.
- Un étudiant qui ne contribue pas au projet sur Github sera pénalisé. Assurez-vous que tous les membres de votre équipe contribuent au projet.