

# Évaluation en cours de formation

## Développeur web et web mobile

### Énoncé

**Durée indicative :** 21h

**Documents autorisés :** Les cours

**Matériel autorisé :** Tout

**Outil interdit :** Tout LLM (ChatGPT, Claude, etc.)



**Contexte fictif :**

Vous travaillez pour une startup qui veut lancer un site de location d'hébergements. Votre mission est de développer une première version du front-end avec React.

**Objectifs de l'évaluation :**

Les étudiants doivent créer une **application React avec Vite, TypeScript, Sass** qui simule un site de location de logements ou d'hébergements, type "Airbnb simplifié".

**Critères d'évaluation :**

- ✓ Initialisation correcte du projet Vite/TS/Sass
- ✓ Mise en place du routing (home + detail + 404)
- ✓ Affichage dynamique depuis le JSON (cards)
- ✓ Page détail avec affichage des données dynamiques
- ✓ Composants Carousel et Collapse fonctionnels
- ✓ Qualité du code, typage TypeScript, bonne structuration
- ✓ Utilisation de Sass bien structurée

# Compétences du référentiel concernées

Cette évaluation a pour vocation de valider les compétences du titre Développeur Web et Web Mobile ayant pour intitulé :

## Activité Type 1 - Développer la partie front-end d'une application web ou web mobile sécurisée

- Développer la partie dynamique des interfaces utilisateur web ou web mobile

# Spécifications fonctionnelles

## 1. Page d'accueil /

- Doit afficher une liste de **logements disponibles**.
- Chaque logement est affiché sous forme de **card** (avec image, titre, localisation).
- Les données proviennent d'un fichier data/logements.json.

## 2. Page de détail /logement/:id

- Lorsqu'on clique sur une card, on est redirigé vers la page de détail du logement.
- Cette page affiche :
  - Le **carousel** des images du logement
  - Le **titre**, la **localisation**, les **tags**
  - Le **nom de l'hôte + photo**
  - La **note** sous forme d'étoiles
  - Une section **Description** en collapse
  - Une section **Équipements** en collapse (liste)

## 3. Page d'erreur /404

- Si on entre une URL qui ne correspond à aucune route, une **page 404 personnalisée** doit s'afficher avec :
  - Message d'erreur clair
  - Lien pour revenir à l'accueil

## 4. Éléments techniques attendus - Structure recommandée

src/

└─ assets/      # images

└─ components/    # Card, Collapse, Carousel, Navbar, Footer, etc.

— pages/	# Home.tsx, Details.tsx, NotFound.tsx
— styles/	# SCSS globaux
— data/	# logements.json
— App.tsx	
— main.tsx	

---

#### 4. Éléments obligatoires pour valider le projet

Élément	Détails
React avec Vite + TypeScript	Projet correctement initialisé
Sass	Au moins un fichier SCSS avec quelques variables
Routing	Routes dynamiques + 404
Page Home	Affiche dynamiquement des cards depuis le JSON
Page Détail	Récupère les données du JSON en fonction de l'ID
Composant Card	Utilisé dans la page d'accueil
Composant Carousel	Affiche les images une par une, avec boutons précédent/suivant
Composant Collapse	Utilisé pour description et équipements
Gestion des erreurs	Affichage si l'ID passé dans l'URL ne correspond à aucun logement
Responsive	Bonus apprécié mais pas obligatoire

---

#### 5. Suggestions d'organisation du travail

##### Étapes proposées aux étudiants

1. Initialisation du projet avec Vite (React + TS + Sass)
2. Création des pages : Home.tsx, Details.tsx, NotFound.tsx
3. Configuration des routes avec React Router
4. Intégration du fichier logements.json
5. Création du composant Card + affichage dynamique sur la Home
6. Création du composant Carousel
7. Création du composant Collapse
8. Connexion logique entre page Home et page Detail (via :id)
9. Gestion des erreurs (ID introuvable → redirection ou 404)
10. Mise en forme avec Sass (et optionally responsive)

Pour cette évaluation, vous devrez produire et soumettre les éléments suivants :

**1. Lien vers le repository GitHub**

- Le lien vers un repository GitHub accessible en lecture où tout le code source est hébergé.
- Le repository doit contenir le projet complet avec l'historique des commits.
- Le code doit être propre, bien structuré, commenté et facilement compréhensible.
- Un fichier readme qui doit contenir toutes les informations nécessaires au projet.

**2. Préparation à la présentation orale**

- Vous validerez votre projet avec une présentation orale. Présentez votre façon de structurer votre projet, votre code, le résultat final ainsi que les choix que vous avez effectué. Soyez prêt à défendre votre projet lors de la phase de questions.

**Instructions de soumission :**

- Créer un fichier txt ou pdf avec le lien du github
- Nommez le fichier de la manière suivante : CP4\_[VotreNom]\_[VotrePrénom].txt ou pdf
- Soumettez l'archive ZIP selon les instructions spécifiques qui vous seront communiquées par l'équipe pédagogique.