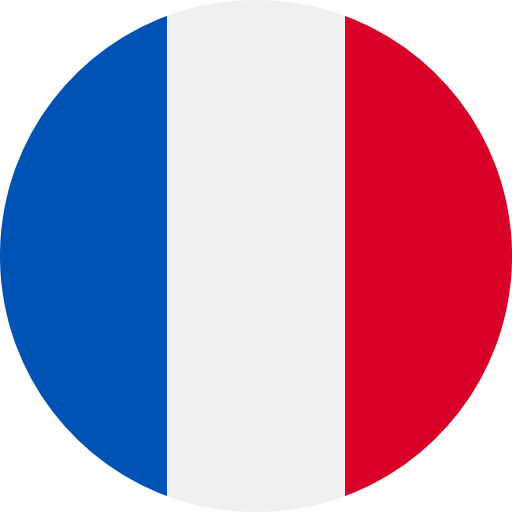
# **Vous êtes développeur·se, et votre centre de formation organise une game jam !**

Assigné

fr



KW

Kevin Wolffcréé : 30/06/25

Créer un mini-jeu web responsive et interactif en HTML, CSS et TypeScript. L’objectif est de mettre en pratique la manipulation du DOM, l’algorithmie de base et le versionnement Git, tout en respectant les contraintes techniques imposées.

## **Référentiels**

* Compétences transversales
* [2023] Concepteur⋅rice développeur⋅se d'applications

## **Compétences visées**

## **Contexte du projet**

**Situation professionnelle :**

​

En tant que concepteur développeur d'application, développer une application web responsive et dynamique en HTML, CSS et TypeScript, à partir d’un brief ou d’une maquette.

Versionner le projet avec Git, le publier sur GitHub Pages et présenter sa réalisation devant le groupe.

​

**Contexte du brief :**

​

En tant que développeur en devenir, vous relevez le défi de créer un petit jeu en utilisant les technologies du web : HTML, CSS et TypeScript. L’objectif est de participer à notre game jam de promotion.

​

Les jeux seront présentés lors d’une journée dédiée où vous pourrez faire découvrir votre création aux autres participants... et découvrir les leurs !

​

Ce projet est l’occasion parfaite de mettre en pratique vos compétences en TypeScript, en algorithmie de base et en manipulation du DOM, dans un contexte à la fois ludique et professionnel.

​

Votre jeu devra être mis en ligne sur GitHub Pages, avec une page contenant à la fois les règles et le jeu lui-même.

​

Idées faciles :

* Tamagochi
* Memory
* Pierre/Feuille/Ciseau
* "Simon" (où il faut reproduire des séquences lumineuses)
* Pendu

​

Idées intermédiaires :

* Jeu de combat 1vs1
* Tape-taupe
* Puissance 4
* Casse brique amélioré / Pong
* Petit jeu de plateforme avec défilement
* Snake

​

Idées difficiles :

* Labyrinthe
* Bataille navale
* Space Invaders
* Frogger
* Doodle Jump
* Flappy Bird

## **Modalités pédagogiques**

**Choisissez votre jeu :**

​

Optez pour un concept qui vous plaît et que vous êtes capable de réaliser dans le temps imparti avec vos compétences actuelles.

​

👉 Mieux vaut un projet simple mais terminé, avec la possibilité d’ajouter des fonctionnalités bonus ensuite, plutôt que de voir trop grand et ne pas finir.

​

💡 Vous êtes libre de réaliser le jeu qui vous plaît, mais il devra respecter le thème suivant : l’univers rétro / rétro-futuriste.

​

**Ne foncez pas tête baissée dans le code :**

​

Prenez le temps de réfléchir à l’ensemble du projet :

* Quelles fonctionnalités sont nécessaires ?
* Quelles mécaniques de jeu voulez-vous mettre en place ?
* Comment allez-vous structurer votre code ?

​

Une conception, une maquette et un cahier des charges sont des étapes clés pour cadrer votre projet et assurer un développement efficace.

​

**Organisez-vous :**

​

Utilisez une todolist, Trello, ou n’importe quel outil de gestion de tâches pour suivre l’avancement de votre projet :

* Tâches à faire
* Tâches en cours
* Problèmes ou bugs rencontrés

​

Note : La maquette n’est pas obligatoire pour la validation du projet, mais elle est fortement conseillée. Elle ne sera pas évaluée, mais elle vous aidera énormément pour la réalisation.

​

***Vous avez 8 après-midi pour réaliser ce projet, à rendre le 11/07/2025.***

## **Modalités d'évaluation**

Évaluation individuelle avec revue du code.

## **Livrables**

- Rendu à déposer sur Simplonline

- Un dépôt GitHub public contenant le code du projet

- L’URL du projet publié en ligne (GitHub Pages)

- Support de présentation du jeu (3 slides : présentation du projet - difficultés rencontrées et solutions adoptées - vos axes de progressions)

## **Critères de performance**

- Interface responsive mobile - tablette - bureau avec une approche mobile first

- Structure HTML sémantique (utilisation de balises appropriées)

- Création d'un projet TypeScript avec sa configuration et un versionnement pertinent

- Organisation claire des fichiers et des assets

- Un commit Git par jour minimum, avec un message de commit explicite