## ETUDE DE CAS

## *Web Agency et son CMS*

## Nom et prénom du stagiaire (à compléter) :

## **Gallego Baptiste**

***Pour réaliser cette étude de cas, nous vous recommandons de vous référer plus particulièrement au module de formation JavaScript – Programmation événementielle et asynchrone.***

# Scénario

Une web-agency développe son propre CMS.

Elle vous a recruté comme stagiaire pour notamment créer le composant **Menu** qui permettra de créer des menus horizontaux ou verticaux à la demande.

# Travaux à réaliser

## Etape 1/2

Le composant sera codé sous la forme d’une classe **Menu.** Il sera 100% indépendant de son conteneur. Les contraintes sont les suivantes :

* Le composant ne communiquera vers l'extérieur que via des événements.
* Le programme principal communiquera avec ce composant uniquement en appelant ses méthodes.
* La racine du composant sera un **HTMLUListElement** qui comportera une série d'items (**HTMLIElement**) précisée à sa création.

**Vous implémenterez le constructeur suivant :**

## **constructor(labels = [], esp = 20, disp = 0)**

où :

* **labels** est un tableau de **string** (les items du menu).
* **esp** est l’espacement en pixels entre deux items (devrait être pair).
* **disp** définit l’orientation du menu (**0** = horizontal, **1** = vertical).

et la méthode :

## **getObjDOM()**

qui retourne l'objet DOM **HTMLUListElement** encapsulé.

Un clic sur un des items déclenche l'événement **menu\_click** dont la propriété

**detail** contient l'objet :

## **{index: index}**

où **index** est l'indice dans **labels** de l'item cliqué.

**AIDE**

Pour créer un événement personnalisé, vous utiliserez le constructeur **CustomEvent()** et pour le déclencher, vous utiliserez la méthode **dispatchEvent()**.

## Pour vérifier le fonctionnement de votre composant **Menu**, testez ce programme principal qui DOIT fonctionner :

**let labels = ["Accueil", "Produits", "Contact"];**

**let menu = new Menu(labels);**

## **let menuObj = menu.getObjDOM(); menuObj.addEventListener('menu\_click', evt =>**

**console.log(`Clic sur ${labels[evt.detail.index]}`)); out.appendChild(menuObj);**

## Etape 2/2

Ajoutez les méthodes suivantes permettant de modifier le menu après sa création :

## **setEsp(esp)**

Elle permet de modifier l’espacement entre les items.

## **setDisp(disp)**

Elle permet de modifier l’orientation du menu.

## **addItem(label)**

Elle permet d’ajouter un item à la suite des autres.

**AIDE**

## Pour vérifier le fonctionnement de votre composant **Menu**, testez ce programme principal qui DOIT fonctionner et modifier le menu après 5 secondes :

**let labels = ["Accueil", "Produits","Contact"];**

**let menu = new Menu(labels);**

**let menuObj = menu.getObjDOM(); menuObj.addEventListener('menu\_click', evt =>**

**console.log(`Clic sur ${labels[evt.detail.index]}`)); out.appendChild(menuObj);**

**setTimeout(() => menu.setEsp(100).setDisp(1).addItem('Test'), 5000);**

# Rendu des travaux

Le candidat doit fournir un dossier zip à son nom contenant:

1. Le fichier Menu.js qui contient le code source commenté du composant **Menu**

**Indiquez ici votre lien googleDrive vers le fichier zip contenant votre travail :**

**………………………………………………………………………………………………..**

**ATTENTION** : penser à ouvrir le fichier au partage par lien et  activer « remplacer par tous les utilisateurs disposants du lien »

# Compétences à valider

* Etre capable d’analyser le cahier des charges d’un composant indépendant.
* Etre capable de coder en JS un composant 100% indépendant en POO.