

# Projet 1 - Le frontend

## Application de gestion de pizzeria

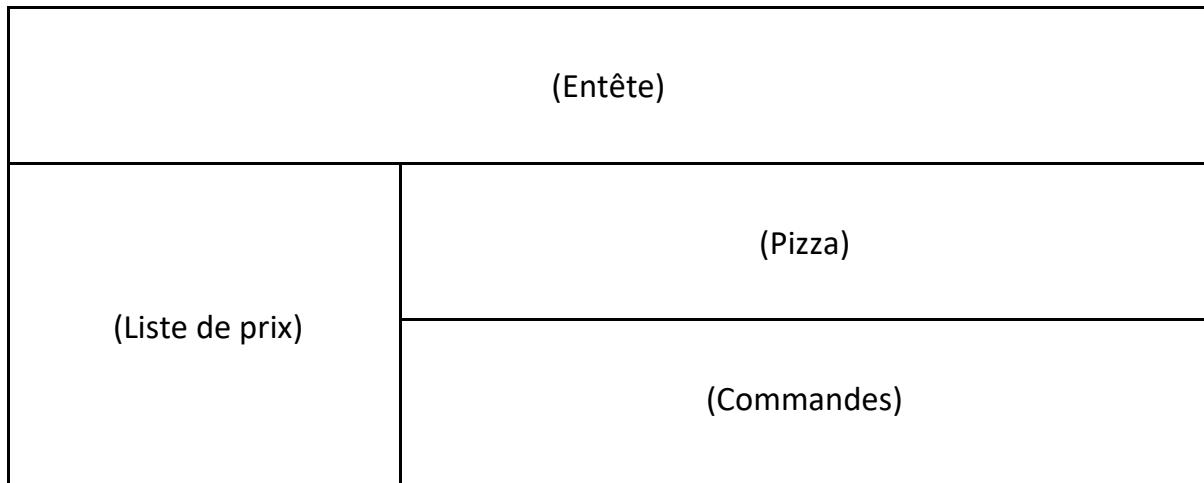
### 1. Contexte

Pour ce projet, vous aurez à construire la couche présentation (“frontend”) d’une application web. Cette couche sera développée en HTML, CSS et JavaScript côté client. L’objectif est de mettre en pratique les notions de ces langages. L’application web à réaliser est une application de gestion de Pizzeria.

### 2. Exigences (55%)

#### Sections de la page (10%)

Votre page web doit s’afficher visuellement selon le modèle suivant. Ce modèle doit être implémenté en CSS avec des “grid”.



La section “Entête” doit occuper une ligne complète.

La section “Liste de prix” doit s’afficher à gauche de la section “Pizza” et “Commandes”. Le ratio des dimensions entre “Liste de prix” et “Pizza” doit être 1:3.

#### Section entête (5%)

La section “Entête” doit afficher le texte “PIZZERIA”. Ce texte doit être centré, utilisant la police de caractère “Arial” et ayant une taille de police plus grande que le reste du texte de la page.

La couleur d’arrière-plan de la section “Entête” doit être gris pâle.

#### Section liste de prix (10%)

La section “Liste de prix” doit afficher les prix de la Pizzéria. L’affichage doit ressembler à ceci :

<b>Menu</b>
Petite pizza ... 8.00\$
Moyenne pizza ... 12.00\$
Grande pizza ... 15.00\$
Garniture viande ... 3.00\$
Garniture légumes... 1,00\$

Les items du menu (ex.: “Petite pizza ... 8.00\$”) doivent être définis dans une liste HTML (`<ul>` et `<li>`).

La couleur d’arrière-plan de la section “Liste de prix” doit être bleu pâle.

### **Section Pizza (15%)**

La section “Pizza” doit afficher la section qui permet de faire les commandes. L’affichage doit ressembler à ceci.

<b>Garniture</b>			
<input type="checkbox"/>	Pepperoni	<input type="checkbox"/>	Oignon
<input type="checkbox"/>	Poulet	<input type="checkbox"/>	Poivron
<input type="checkbox"/>	Viande	<input type="checkbox"/>	Olives
<input type="checkbox"/>	Crevette	<input type="checkbox"/>	Champignon

<b>Format</b>	
<input type="checkbox"/>	Petite
<input type="checkbox"/>	Moyenne
<input type="checkbox"/>	Grande

[ Commander ]

Les variétés de garniture ainsi que les formats doivent utiliser des tableaux HTML (<table>).

Les “” dénotent des éléments de formulaire de type “radio”. On doit pouvoir choisir une seule garniture viande et plusieurs garniture légumes et un seul format.

“[ Commander ]” dénote un bouton avec le texte “Commander” qui lorsqu'il sera cliqué il devra ajouter une entrée dans la section “Commandes” selon le format spécifié pour la section. L'ajout ne doit pas modifier le HTML des autres entrées déjà présentes.

La couleur d'arrière-plan de la section “Liste de prix” doit être rouge pâle.

### **La section commande** (15%)

La section “Commandes” doit afficher les commandes qui ont été ajoutées avec le bouton “Commander”. L'affichage doit ressembler à ceci.

Commandes	
2025-09-01 13:45 - “Petite” Pizza “Viande - Poivron- olives »	<a href="#">Supprimer</a>
2025-09-01 13:43 - “Grande” Pizza “Poulet – Oignon-olives”	<a href="#">Supprimer</a>

Les entrées les plus récentes doivent être placées au-dessus. Pour l'affichage de la date et de l'heure, vous devez utiliser l'API “Date” de JavaScript :

([https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Date](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date)).

La couleur d'arrière-plan de la section “Liste de prix” doit être vert pâle.

Lorsqu'une pizza est ajoutée à la section « Commandes », un bouton « Supprimer » doit permettre de retirer cette pizza de la liste sans affecter les autres commandes.

## **3. Rapport**

### **Question #1** (2%)

L'utilisation de <ul> et <li> vient avec un visuel par défaut qui n'est pas désirable (il y a un point qui s'affiche avant chaque item <li>). Expliquez comment vous avez été capable d'enlever ce visuel et pourquoi votre approche fonctionne.

### **Question #2** (4%)

Expliquez comment vous avez été capable de faire en sorte qu'une seule garniture viande, plusieurs garnitures légume et un seul format puissent être sélectionnés en même temps.

### **Question #3** (4%)

Expliquez comment vous avez été capable d'implémenter l'ajout d'une entrée dans la section commande sans modifier les items déjà ajoutés.

### **Question #4** (2%)

Expliquez pourquoi il est préférable d'utiliser `addEventListener` plutôt que les attributs HTML (`onclick`, `onchange`, etc.).

### **Question #4** (18%)

1. Comment vous avez créé le bouton « Supprimer » pour chaque pizza ajoutée dynamiquement.
2. Comment vous avez identifié et supprimé uniquement la pizza correspondante.
3. Comment votre approche évite la duplication de gestionnaires d'événements et reste performante.

## **4. Modalités**

### **Équipe**

Ce projet peut être fait en équipe de 2 étudiants.

### **Remise**

Le laboratoire est à remettre le 18/02/2026 avant 23h59.

Vous devez remettre un fichier ZIP contenant :

- L'ensemble des fichiers HTML, CSS et JavaScript
- Un rapport au format docx ou pdf répondant aux questions de la section “Rapport”, Ce rapport doit être nommé “rapport\_nomEquipe” selon le format choisi.
- Une seule remise par équipe.

### **Consignes générale**

- L'utilisation d'IA pour ce travail est interdite.
- Vous ne pouvez pas utiliser de librairie JavaScript ou CSS pour ce travail. L'ensemble du HTML, CSS et JavaScript remis doit avoir été écrit par vous.

## **Barème**

La correction se fait selon le barème suivant :

<b>Thème</b>	<b>Description</b>	<b>Poids</b>
<b>Développement des exigences (65%)</b>	Sections de la page	10%
	Section entête	5%
	Section liste de prix	10%
	Section Pizza	20%
	Section commandes	10%
	bonnes pratiques HTML, CSS et JavaScript	10%
<b>Rapport (30%)</b>	Question 1	2%
	Question 2	4%
	Question 3	4%
	Question 4	2%
	Question 5	18%
<b>Qualité du rapport (5%)</b>	Mise en page, présentation, introduction, français	5%