

1. Introduction

Dans le cadre de ma première année de BTS SIO (Services Informatiques aux Organisations), option SLAM (Solutions Logicielles et Applications Métiers), j'ai eu l'opportunité d'effectuer un stage de huit semaines, du 12 mai au 6 juillet, au sein de l'entreprise Ekino, à Pantin.

Ce stage a été pour moi l'occasion de découvrir le monde professionnel et de m'immerger dans le quotidien d'une équipe de développement web. J'ai été intégré à l'équipe front-end ainsi qu'à l'équipe design, et j'ai participé à la réalisation d'applications web accessibles et esthétiques, répondant à des besoins concrets.

J'ai ainsi pu mettre en pratique les compétences acquises au cours de ma formation, en particulier sur les langages du web et la gestion de projet. J'ai également appris à utiliser de nouveaux outils, à collaborer avec des professionnels expérimentés, et à travailler dans un cadre agile et stimulant.

Ce rapport retrace l'ensemble de cette expérience. Il présente tout d'abord l'entreprise Ekino, puis détaille les différentes missions qui m'ont été confiées, avant de faire le point sur les compétences développées et les enseignements que j'en retire, tant sur le plan technique que personnel.

2. Présentation de l'entreprise

Ekino est une entreprise spécialisée dans le conseil et la réalisation de solutions digitales. Fondée il y a plus de dix ans, elle compte aujourd'hui plus de 600 collaborateurs répartis sur plusieurs pôles d'expertise : consulting, design, data, ingénierie, création, UX/UI, et bien d'autres.

L'entreprise dispose d'une présence internationale, avec des bureaux à Paris, Bordeaux, Ho Chi Minh City, Bangaluru, Hong Kong et Singapour. Cette implantation dans plusieurs zones géographiques permet à Ekino d'intervenir sur des projets variés, en collaborant à la fois avec des équipes locales et internationales. Chaque entité travaille en coordination avec les autres, tout en conservant une certaine autonomie qui permet d'adapter les méthodes de travail aux spécificités de chaque région.

L'approche adoptée par Ekino est transversale et pluridisciplinaire. Chaque projet fait intervenir plusieurs expertises, dans le but de proposer des solutions sur mesure, durables et adaptées aux enjeux métiers du client. Cette organisation permet de garantir un haut niveau de qualité tout en restant agile face aux contraintes spécifiques de chaque projet.

J'ai effectué mon stage sur le site de Pantin, dans un cadre de travail moderne et agréable. J'y ai découvert un environnement professionnel dynamique, où la collaboration et la communication sont au cœur de l'organisation. J'ai été intégré à l'équipe front-end, ce qui m'a permis de participer à des projets concrets de développement web. J'ai également eu l'occasion d'échanger avec les membres de l'équipe design, ce qui m'a permis de mieux comprendre les interactions entre les développeurs et les designers, notamment dans la conception d'interfaces accessibles et ergonomiques.

Ekino accorde une grande importance à l'accessibilité numérique, un sujet sur lequel mon tuteur, Jonathan Hamoudi, est particulièrement impliqué. Grâce à ses conseils, j'ai pu découvrir des bonnes pratiques essentielles que j'ai ensuite intégrées dans mes projets. Cela m'a permis de mieux comprendre les enjeux liés à l'inclusion numérique et de concevoir des interfaces accessibles au plus grand nombre.

Enfin, tout au long de mon stage, j'ai pu assister à plusieurs rituels d'équipe, comme les réunions hebdomadaires ou les temps de partage autour de sujets techniques ou créatifs. Ces moments m'ont permis d'enrichir mes connaissances, d'observer des méthodes de travail collaboratives et de mieux comprendre la culture d'entreprise propre à Ekino. Pour un premier stage dans le domaine du développement, cette immersion a été particulièrement formatrice et motivante.

3. Mes missions chez Ekino

Durant ces huit semaines, j'ai eu l'opportunité de travailler sur plusieurs projets web concrets. Chaque mission m'a permis d'aborder une facette différente du développement front-end, tout en intégrant des notions essentielles comme l'accessibilité, la structuration du code, le versioning, ou encore la logique utilisateur. Voici les principales missions qui m'ont été confiées :

3.1 Application "JSONPlaceholder"

Cette première mission avait pour objectif de me familiariser avec JavaScript, l'utilisation des API, et la manipulation du DOM.

Objectif :

Créer une application web accessible affichant les données de 10 utilisateurs, en utilisant l'API JSONPlaceholder et la documentation MDN.

Étapes réalisées :

- Recherche et compréhension de la documentation API JSONPlaceholder
- Écriture du script de récupération des données en `fetch()`
- Dynamisation du DOM avec les données utilisateurs (nom, email, ville, etc.)
- Ajout d'une structure HTML/CSS simple mais accessible
- Vérification de l'accessibilité avec les outils intégrés aux navigateurs
- Hébergement du projet sur GitHub Pages : [jsonplaceholder-acc](#)

Cette mission m'a permis de mieux comprendre comment structurer mon code, comment interagir avec une API, et surtout l'importance de rendre une interface compréhensible même sans souris ou écran.

3.2 Application "Calculette de monnaie"

Ce projet m'a permis d'approfondir la logique de calcul, la structuration du code, mais aussi la mise en place de tests unitaires.

Objectif :

Créer une application permettant d'entrer une somme due, une somme donnée par le client, puis d'afficher la monnaie à rendre avec le détail (pièces/billets).

Étapes réalisées :

- Analyse du besoin, élaboration d'un plan d'action (découpage des tâches, durée estimée)
- Implémentation des fonctions de calcul en JavaScript
- Utilisation de Jest pour réaliser des tests unitaires sur les fonctions critiques
- Ajout d'un affichage clair et responsive pour les résultats
- Stylisation en CSS avec un soin porté à l'accessibilité (contrastes, navigation clavier)
- Mise en ligne sur GitHub Pages : [MonnaieZen](#)

J'ai particulièrement apprécié cette mission car elle faisait appel à la rigueur et au raisonnement. J'ai appris à anticiper les erreurs possibles, à structurer mon code de façon claire, et à versionner chaque étape du développement.

3.3 Application "WeatherFashion"

Cette mission a été l'une des plus créatives de mon stage. L'objectif était de concevoir une application météo capable de suggérer une tenue vestimentaire en fonction du temps.

Objectif :

Développer une application qui affiche la météo d'une ville et propose des conseils vestimentaires adaptés. Utilisation de l'API OpenWeather.

Étapes réalisées :

- Étude de l'API OpenWeather : récupération des données météo actuelles et des prévisions
- Conception du moteur de suggestion de tenues selon des plages de température et conditions météo
- Création de maquettes dans Figma pour définir une interface claire et attrayante
- Intégration HTML/CSS/JS avec une attention particulière à l'accessibilité et à la lisibilité
- Mise en place d'un formulaire permettant à l'utilisateur de choisir la ville
- Déploiement sur GitHub Pages : [WeatherFashion](#)

Ce projet m'a permis d'aller plus loin dans la réflexion UX/UI, en conciliant esthétique, accessibilité et logique fonctionnelle. J'ai apprécié le fait d'avoir une grande autonomie dans les choix graphiques et techniques tout en restant cadre par les objectifs fixés.

3.5 Application “Contact Manager” en TypeScript

Dans la continuité de mes projets précédents, je clôture mon stage par une application construite entièrement en TypeScript. L'objectif était de découvrir les avantages concrets offerts par TypeScript dans un projet simple, mais structuré.

Objectif :

Développer une application web permettant de gérer une liste de contacts (ajout, modification, suppression), avec une interface propre et accessible, tout en exploitant les fonctionnalités de TypeScript.

Étapes réalisées :

- Définition d'une **interface Contact** pour structurer les données manipulées (nom + email)
- Création d'un formulaire HTML permettant la saisie des informations
- Développement de la logique TypeScript :
 - Ajout de nouveaux contacts dans un tableau
 - Affichage dynamique des éléments via le DOM
 - Implémentation de la modification et de la suppression de chaque contact
- Compilation du code TypeScript en JavaScript via le fichier tsconfig.json et organisation du dossier dist
- Stylisation de l'interface avec un CSS sobre et épuré
- Tests manuels de l'ensemble des fonctionnalités dans différents navigateurs

3.5 Découverte de nouvelles méthodes et outils

En parallèle de ces projets, j'ai été amené à observer et expérimenter différents outils professionnels :

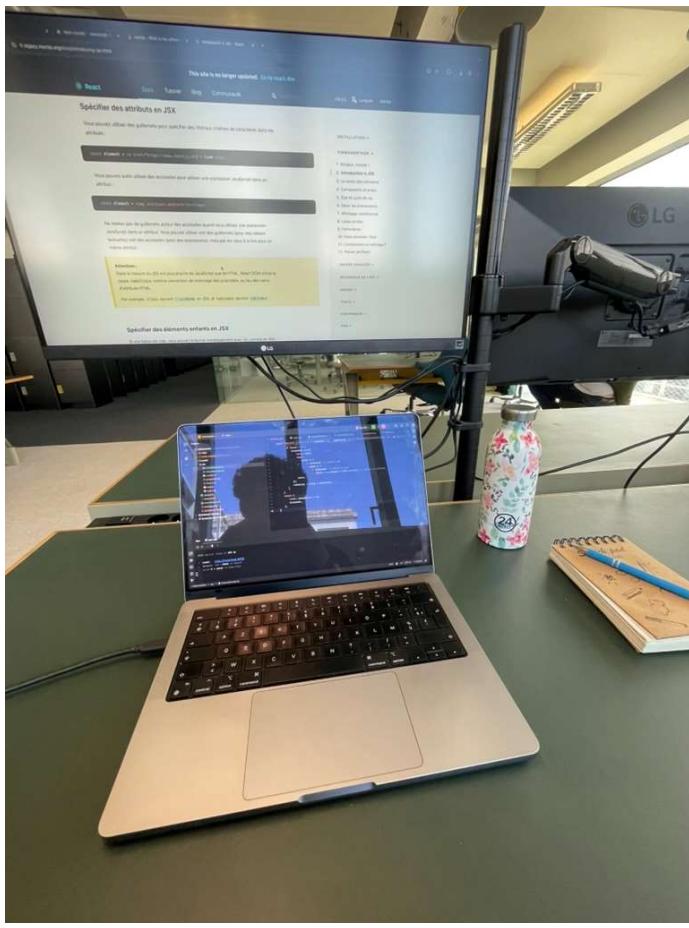
- Introduction à React, Angular et TypeScript via de la documentation externe et discussions avec les développeurs
- Participation à des réunions en mode agile (Scrum), avec présentation du backlog, répartition des tâches, suivi d'avancement
- Utilisation de GitHub, avec des pull requests, des revues de code et une gestion des versions
- Familiarisation avec les pratiques de développement responsable, incluant accessibilité et documentation claire

Enfin, au-delà des aspects techniques, j'ai aussi participé à la vie de l'entreprise : échanges informels, réunions d'équipes « Je s'appelle Squad » tous les 15 jours - où l'on discute d'un thème en particulier, moments de détente, activités créatives comme l'atelier de création de maquette d'un site e-commerce ou encore des discussions sur l'intelligence artificielle – ces activités sont des « Rassemblements front » et ont lieu une fois par mois.

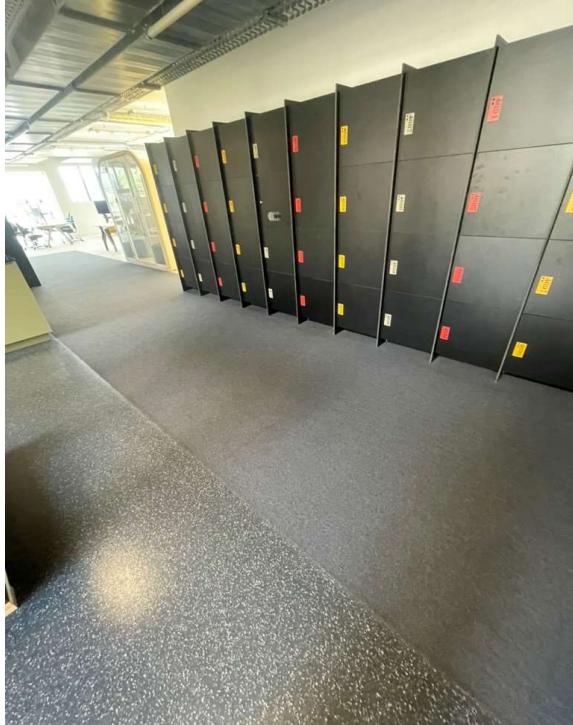
Cette immersion m'a permis de découvrir ce que signifie réellement travailler dans une agence digitale moderne et humaine, où les compétences techniques sont aussi valorisées que l'esprit d'équipe.

4. Mon expérience chez Ekino : une immersion progressive dans le développement web

Dès les premiers jours, j'ai été chaleureusement accueilli dans les locaux d'Ekino à Pantin. Mon tuteur, Jonathan Hamoudi, m'a rapidement mis à l'aise en me présentant à plusieurs membres des équipes. J'ai pris possession du matériel mis à disposition (MacBook Pro) et découvert les outils de travail utilisés en interne : Git pour le versionnage, Slack pour la communication, Trello pour le suivi des tâches. Tout était organisé, clair, et propice à l'apprentissage.



Programmation et documentation - photo personnelle



Caisiers - photo personnelle



Bulle d'isolement - photo personnelle

La première étape de mon immersion a été de me documenter et de m'auto-former. J'ai passé beaucoup de temps à lire, à regarder des tutoriels, à explorer la documentation sur MDN ou des forums comme Stack Overflow. Mon objectif était simple : comprendre l'environnement dans lequel j'allais évoluer. Jonathan m'a fait découvrir Angular et React, deux frameworks que je n'avais jamais utilisés auparavant. Cette première phase m'a aussi permis de consolider mes bases en JavaScript et de découvrir la logique des APIs.

Rapidement, j'ai été amené à créer une première application basée sur l'API JSONPlaceholder. Le principe était simple : récupérer une liste d'utilisateurs et afficher leurs informations de manière accessible. Grâce à `fetch()` et à la manipulation du DOM, j'ai pu construire une interface dynamique, soignée, tout en respectant les bonnes pratiques d'accessibilité (balises sémantiques, navigation au clavier, etc.). J'ai rencontré des difficultés, notamment sur la gestion des requêtes ou les erreurs liées aux appels réseau, mais j'ai su les surmonter en m'appuyant sur des ressources en ligne et l'aide de mes collègues.

Cette montée en compétence s'est poursuivie avec un projet plus ambitieux : **MonnaieZen**, une application de calcul de monnaie. À partir d'une somme due et d'un montant donné, l'application devait calculer et afficher le rendu exact, détaillé billet par billet, pièce par pièce. Pour cela, j'ai écrit une fonction de calcul optimisée, que j'ai accompagnée de **tests unitaires avec Jest**. Cela m'a permis de découvrir l'intérêt des tests dans le développement : vérifier le bon fonctionnement d'un code sans avoir à tout relancer à chaque modification. J'ai également pris conscience de l'importance d'un **versionnage propre** : chaque étape importante faisait l'objet d'un commit. Enfin, le style de l'application a été particulièrement soigné : je voulais que l'expérience soit fluide, claire, avec une interface intuitive.

C'est dans cette logique que j'ai enchaîné avec le projet **WeatherFashion**, une application qui m'a demandé davantage de réflexion. L'idée était de récupérer la météo en temps réel grâce à l'API OpenWeather, et de suggérer une tenue adaptée en fonction des conditions. J'ai ainsi mis en place un moteur de suggestions vestimentaires, basé sur des paliers de température et les types de temps (pluie, soleil, etc.). Ce projet m'a appris à penser **architecture**, à structurer mon code de manière modulaire et lisible, à mieux gérer les erreurs et les cas limites (ville

introuvable, API indisponible...). J'ai aussi documenté tout le développement dans un journal de bord personnel, ce qui m'a aidé à garder une vision claire de mon avancement.

Un autre aspect très formateur a été ma découverte de **Figma**. Grâce à Jonathan, j'ai eu la chance de passer une journée avec l'équipe design, où j'ai observé leur méthode de travail : création de maquettes, réflexion autour des parcours utilisateurs, choix des couleurs, et surtout respect des critères d'accessibilité. Cette expérience m'a permis de mieux comprendre les enjeux de l'UX/UI design. J'ai d'ailleurs appliqué ces principes à WeatherFashion, en réalisant moi-même une maquette complète et en veillant à l'harmonie des contrastes, à l'espacement des éléments, et à la lisibilité générale.

Dans cette dynamique d'apprentissage continu, Jonathan m'a proposé un dernier exercice : développer une application complète en TypeScript. Le but était de me confronter à un environnement plus structuré, typé, et proche des pratiques professionnelles modernes. L'idée était simple en apparence — créer une application de gestion de contacts — mais le défi résidait dans l'utilisation de TypeScript, un langage que je n'avais encore jamais utilisé en autonomie.

Sous ses conseils, j'ai donc conçu une application permettant d'ajouter, modifier et supprimer des contacts, à partir d'un formulaire HTML relié à une logique entièrement codée en TypeScript. J'ai défini une interface Contact pour encadrer la structure de mes données, manipulé le DOM en typant correctement les éléments HTML, et mis en place un affichage dynamique et fluide. Chaque interaction devait être simple et directe : un formulaire clair, des boutons d'action pour chaque contact, et une mise à jour immédiate de l'affichage. J'ai également appris à configurer un projet TypeScript avec un fichier `tsconfig.json`, à organiser les fichiers de sortie dans un dossier `dist`, et à respecter une structure de code propre et évolutive.

Ce projet m'a beaucoup appris. Non seulement j'ai découvert les apports concrets du typage statique dans la détection d'erreurs, mais j'ai aussi pris conscience des bonnes pratiques de lisibilité, de modularité et de rigueur que l'on attend d'un développeur front-end en entreprise. Et surtout, cette mission m'a permis de clôturer

mon stage sur une note très constructive, en appliquant tout ce que j'avais appris, dans un contexte réaliste mais guidé.

Mais mon stage ne s'est pas résumé au développement. J'ai aussi pris part à la vie d'équipe. J'ai assisté à des réunions techniques, où des collègues présentaient des sujets variés : nouveaux frameworks, retours d'expérience, ou même intelligence artificielle. Un vendredi, j'ai participé à un atelier autour de l'IA : on devait deviner si un bout de code avait été écrit par un humain ou par une intelligence artificielle. L'exercice était à la fois ludique et piégeux. Il a également nourri ma veille technologique, que je menais déjà autour de l'informatique quantique et de l'intelligence artificielle. Ce stage m'a aidé à clarifier mes envies : je souhaite approfondir mes recherches sur l'IA, domaine que je trouve intéressant et pertinent.



Rassemblement Front du vendredi – photos personnelles

En somme, mon stage chez Ekino a été une vraie immersion dans le monde du développement web professionnel. Il m'a appris à coder mieux, à travailler en équipe, à m'organiser et à m'autoformer. Il m'a surtout donné envie d'aller plus loin, de continuer à apprendre, et de m'investir encore plus dans cette voie. J'en garde un excellent souvenir et l'envie sincère de revenir un jour dans un environnement de travail aussi stimulant.

5. Compétences acquises et apports du stage

Ce stage chez Ekino m'a permis d'évoluer à plusieurs niveaux, aussi bien sur le plan technique que sur le plan personnel. En travaillant sur des projets concrets et en intégrant une équipe expérimentée, j'ai pu me rendre compte des attentes du monde professionnel et de la rigueur nécessaire pour évoluer dans le domaine du développement web.

5.1 Compétences techniques développées

- **Maîtrise du HTML, CSS, JavaScript** : j'ai renforcé mes bases et appris à structurer mon code de façon plus claire et maintenable.
- **Utilisation d'APIs** : que ce soit avec JSONPlaceholder ou OpenWeather, j'ai appris à interagir avec des services externes et à traiter des données en temps réel.
- **Tests unitaires avec Jest** : une vraie découverte qui m'a permis de comprendre l'intérêt des tests dans un projet bien structuré.
- **Maquettes avec Figma** : création d'interfaces et mise en forme de parcours utilisateur.
- **Notions de React, Angular et TypeScript** : même si je n'ai pas développé avec ces technologies, j'ai eu l'occasion de les découvrir à travers des échanges et démonstrations.
- **Versionnement avec Git** : utilisation avancée de GitHub avec commits réguliers, branches, pull requests, etc.
- **Accessibilité numérique** : intégration des bonnes pratiques pour rendre les interfaces utilisables par tous.

5.2 Méthodologie et organisation

- **Approche agile (Scrum)** : réunions de suivi, backlog, gestion de tâches, itérations...

- **Planification et découpage du travail** : établir une feuille de route, estimer les temps de développement.
- **Documentation** : rédiger des commentaires, expliquer mes choix, garder une trace de l'évolution du projet.

5.3 Compétences humaines et professionnelles

- **Travail en équipe** : apprendre à échanger avec mes collègues, poser des questions, proposer des solutions.
- **Autonomie** : savoir avancer seul tout en sachant quand demander de l'aide.
- **Communication** : que ce soit à l'écrit (issues GitHub, documentation) ou à l'oral (réunions, présentations).
- **Ouverture** : découvrir d'autres métiers (UX, UI, accessibilité), comprendre le rôle de chacun dans un projet global.

5.4 Impact sur mon projet professionnel

Ce stage a été un vrai déclencheur : il m'a confirmé que je voulais poursuivre dans le développement, avec peut-être une sensibilité particulière pour le front-end, le design d'interface et l'accessibilité. J'ai découvert une ambiance de travail dans laquelle je me sens bien : dynamique, bienveillante, stimulante. J'aurais voulu que ce stage dure plus longtemps tant je m'y suis épanoui.

Je ressors de cette expérience avec plus de confiance, de motivation, et surtout une vision plus claire de ce que je veux construire pour la suite de mon parcours.

6. Conclusion

Ce stage chez Ekino a été une expérience marquante, tant sur le plan humain que professionnel. Il m'a permis d'appliquer mes connaissances, de découvrir de nouvelles méthodes de travail, mais surtout de confirmer mon envie de poursuivre dans le développement web et l'univers du numérique.

Travailler dans une structure aussi dynamique m'a fait prendre conscience de l'importance de la collaboration, de l'organisation et de l'ouverture d'esprit dans la réussite d'un projet. J'ai aussi appris à avoir confiance en mes capacités et à ne pas hésiter à demander de l'aide quand cela est nécessaire.

Je quitte ce stage avec de nombreuses compétences en plus, mais aussi avec beaucoup de motivation pour la suite de ma formation et mon avenir professionnel.

7. Remerciements

Je tiens à remercier chaleureusement l'ensemble de l'équipe d'Ekino Pantin pour m'avoir accueilli avec bienveillance et professionnalisme tout au long de mon stage.

Je remercie tout particulièrement **Jonathan Hamoudi**, mon maître de stage, pour sa confiance, ses conseils avisés et sa pédagogie. Grâce à lui, j'ai pu progresser sur de nombreux aspects techniques tout en découvrant la réalité du travail en équipe dans un environnement stimulant.

Je souhaite également adresser mes remerciements à **Emmanuelle Elisabeth, Yoann Guesnerot, Matthieu Marseille, Erick Boulevard, Iker Bedecarrax, Dimitri Louikine et Hamza Charafi**, membres de l'équipe front, pour leur accompagnement, leur patience et leurs retours toujours constructifs. Chacun m'a transmis à sa manière des connaissances précieuses qui m'accompagneront bien au-delà de ce stage.

Merci également à **Mathilde Lima-De-A-Albuquerque et Marie Niollet**, de l'équipe design, pour m'avoir permis de découvrir leur univers, leurs méthodes et leur sens du détail. Cette immersion dans le design m'a permis de mieux comprendre l'importance de l'accessibilité et de l'expérience utilisateur dans un projet web.

Enfin, je remercie **Jean-François Vert**, mon tuteur pédagogique au lycée Turgot, pour son suivi tout au long de cette période.

Ce stage a été une expérience extrêmement enrichissante, tant sur le plan humain que professionnel, et je suis sincèrement reconnaissant à tous ceux qui y ont contribué.

8. Annexes

8.1 – Application JSONPlaceholder

HTML (extrait)

```
<ul id="user-list" aria-label="Liste des utilisateurs avec commentaires"></ul>
```

JavaScript - Requête API et manipulation du DOM

```
fetch('https://jsonplaceholder.typicode.com/users')
  .then(response => response.json())
  .then(users => {
    users.forEach(user => {
      const userLi = document.createElement("li");
      userLi.innerHTML = `<strong>${user.name}</strong> (<a href="mailto:${user.email}">${user.email}</a>)`;
      userList.appendChild(userLi);
    });
  });
});
```

8.2 – Application MonnaieZen

JavaScript - Fonction de calcul de monnaie

```
export const COUPURES_EURO = [500, 200, 100, 50, 20, 10, 5, 2, 1, 0.5, 0.2, 0.1, 0.05, 0.02, 0.01];

function enCentimes(montantEuro) {
  return Math.round(parseFloat(montantEuro) * 100);
}

export function calculerMonnaie(montantDu, montantDonne) {
  const duCent = enCentimes(montantDu);
  const donneCent = enCentimes(montantDonne);

  if (Number.isNaN(duCent) || Number.isNaN(donneCent)) {
```

```

        throw new Error("Montant invalide");
    }

const renduCen = donneCen - duCen;
if (renduCen < 0) {
    throw new Error("Somme insuffisante");
}

const details = [];
let reste = renduCen;
for (const coupure of COUPURES_EURO) {
    const valeurCen = Math.round(coupure * 100);
    const quantite = Math.floor(reste / valeurCen);
    if (quantite > 0) {
        details.push({ valeur: coupure, quantite });
        reste -= quantite * valeurCen;
    }
}
return { total: renduCen / 100, details };
}

```

8.3 – Application WeatherFashion

HTML (extrait)

```

<form id="form-city">
    <label for="city">Ville :</label>
    <input type="text" id="city" required placeholder="Paris" />
    <button type="submit">Voir la météo</button>
</form>
script.js (extrait)
function displayOutfitAd(current) {
    const temp = current.temp;
    let advice = "";

    if (temp < 5)
        advice = "Grosse veste, bonnet, gants";
    else if (temp < 12)
        advice = "Veste légère ou pull";
    else if (temp < 20)
        advice = "T-shirt avec veste";
    else

```

```
    advice = "Casquette, lunettes de soleil";  
  
    adviceZone.textContent = `Conseil tenue : ${advice}`;  
}
```

JavaScript - Suggestion de tenues selon météo

```
function displayOutfitAd(current) {  
  
    const temp = current.temp;  
  
    let advice = "";  
  
    if (temp < 5)  
  
        advice = "Grosse veste, bonnet, et gants obligatoires.";  
  
    else if (temp < 12)  
  
        advice = "Veste légère ou pull chaud.";  
  
    else if (temp < 20)  
  
        advice = "T-shirt avec une petite veste.";  
  
    else  
  
        advice = "Casquette, lunettes de soleil : il fait beau !";  
  
  
    document.getElementById('advices').textContent = advice;  
}
```

8.2 – Application Contact Manager

TypeScript - Typage et logique

```
interface Contact {  
  
    name: string;  
  
    email: string;
```

```
}

const contacts: Contact[] = [];

function renderContacts(): void {

    const list = document.getElementById('contact-list') as
HTMLULListElement;

    list.innerHTML = '';

    contacts.forEach((contact, index) => {

        const li = document.createElement('li');

        li.innerHTML = `

<strong>${contact.name}</strong> (${contact.email})

<button class="edit-btn">Modifier</button>

<button class="delete-btn">Supprimer</button>

`;

        list.appendChild(li);
    });
}
```

Remarque personnelle :

J'ai veillé à améliorer la lisibilité de mon code au fil des projets, notamment en structurant mieux mes fichiers, en utilisant des variables CSS, JS cohérentes en anglais, et en gardant un style uniforme pour chaque projet. Cette évolution m'a permis de gagner en efficacité et en professionnalisme.