1. Présentation des livrables **15 MINUTES** :
   1. Contexte du projet : Sarah ma CTO m’a envoyé la maquette Figma de Loïc du site **Booki**, notre entreprise, qui veut que les usagers puissent trouver des hébergements et des activités dans la ville de leur choix.
   2. Montrer le résultat du site en montrant les trois versions : 1440px, 1023px, 375px.
   3. Montrer le code sur VSCODE : highlight sur la partie responsive, le dossier avec les fichiers css appelés dans global, les filtres, le footer et aspect-ratio pour les formats d’images.
   4. Bilan du projet : j’ai beaucoup appris en peu de temps vu que j’ai dû m’adapter pour trouver des solutions aux problèmes que je rencontrais pour arriver au résultat escompté.
2. Discussion **10 MINUTES** :
   1. Me préparer sur les questions validité code W3C (normalement good)
   2. Web sémantique (les langages HTML et CSS pour façonner une page)
   3. L’importance de passer son code aux validateurs (vérifier s’il n’y a pas de répétitions, s’il n’y a pas de bugs et que le site agit comme on le veut)
   4. Fonctionnement de VScode (trql)
   5. Spécificité en CSS
   6. Etude et découpe de la maquette (ça va je connais mon code)
   7. L’importance de séparer le HTML et le CSS (pour plus de clareté et bien géré les priorités)
3. Débriefing
   1. DEMANDER CONTACTS ETC SUR METZ ET Luxembourg

**1. Le Web sémantique**

* **Définition** : Le web sémantique structure les données pour qu'elles soient compréhensibles par les machines (ex : moteurs de recherche). Cela améliore l'accessibilité et l'indexation.
* **Pourquoi utiliser les balises sémantiques ?**
  + Renforce l'accessibilité pour les lecteurs d'écran.
  + Facilite le référencement naturel (SEO).
  + Rend le code plus clair et maintenable (ex : <article>, <section>, <header>).

**2. L’importance de passer son code aux validateurs**

* **Qu’est-ce qu’un validateur ?**
  + Outils comme [W3C Validator](https://validator.w3.org/) pour le HTML/CSS.
* **Pourquoi ?**
  + Détecter les erreurs de syntaxe ou d'accessibilité.
  + Assurer la compatibilité multi-navigateurs.
  + Produire un code propre, pérenne et conforme aux standards.

**3. Le fonctionnement de l'IDE (Visual Studio Code)**

* **Pourquoi choisir VS Code ?**
  + Léger, rapide, personnalisable (extensions, thèmes).
* **Fonctionnalités importantes pour un développeur web** :
  + Coloration syntaxique pour HTML/CSS.
  + Extensions comme Prettier (formatage automatique) ou Live Server (prévisualisation).
  + Terminal intégré pour lancer des commandes (Sass, Git, etc.).

**4. La spécificité en CSS**

* **Définition** : La spécificité détermine quelle règle CSS s'applique lorsqu'il y a des conflits.
* **Ordre de priorité** :
  + Inline styles (style="...") > ID (#id) > Classes/Pseudos/Attributs (.class, :hover) > Balises (h1, p).
  + L’utilisation de !important force une règle à s’appliquer, mais doit être évitée.
* **Conseils** : Garder une structure logique et éviter les sélecteurs trop spécifiques.

**5. L’étude et la découpe de la maquette**

* **Pourquoi est-ce crucial ?**
  + Comprendre la structure avant de coder évite les erreurs.
  + Identifier les zones clés : header, footer, contenu principal, etc.
* **Méthodologie** :
  + Séparer en sections claires.
  + Réutiliser des composants pour optimiser le code.
  + Respecter les espacements, tailles et alignements de la maquette.

**6. L’importance de séparer le HTML du CSS**

* **Avantages** :
  + Séparation des préoccupations : HTML pour la structure, CSS pour le style.
  + Réutilisation facile : un seul fichier CSS pour plusieurs pages HTML.
  + Maintenance simplifiée : modification des styles centralisée.
  + Favorise un travail en équipe plus fluide (back-end vs front-end).