

Guide d'étapes clés - Créez la page d'accueil d'une agence de voyage avec HTML & CSS

Comment utiliser ce document ?

Sur cette page, vous trouverez un exemple de découpage en étapes pour faire votre projet. Vous y trouverez :

- des recommandations pour réussir chaque étape ;
- les problèmes récurrents et points de vigilance à garder en tête ;
- une estimation de votre avancement sur l'ensemble du projet.

Ce découpage est simplement une suggestion pour vous organiser. Vous n'êtes pas obligé de compléter les étapes dans l'ordre.

Gardez en tête que votre progression sur les étapes n'est qu'une estimation, et sera différente selon votre vitesse de progression.

Recommandations générales

Dans ce projet, vous utiliserez **Figma** pour accéder aux maquettes desktop, tablette et mobile du site. Créez-vous un compte et connectez-vous à Figma afin d'avoir accès aux détails des éléments dans le menu de droite. Attention cependant : vous pourrez accéder au code CSS de chaque élément. Vous pouvez repartir de ce code pour votre intégration si vous le souhaitez, mais il sera incomplet et nécessitera des modifications. Notamment, les dimensions sont toutes en pixels, mais vous aurez besoin d'en adapter certaines dans des valeurs relatives (unités "em" notamment).

Étape 1 : Mettez en place votre environnement de développement

5 % de progression

Une fois cette étape terminée, je devrais avoir :

- l'éditeur de code VSCode installé ;
- un fichier index.html ;
- un dossier "css" avec un ou des fichiers CSS ;
- un dossier "assets" ou "images" contenant les images du projet.
- le projet créé sur un repo GitHub.

Recommandations :

1. Commencez par installer VSCode.
2. Initialisez le projet avec Git, et publiez le repo sur GitHub.
3. Importez les variantes de la police Raleway depuis Google Fonts.
 - Quelles variantes correspondent à la maquette ? Pour le savoir, testez-les sur un mot.
4. Pour pouvoir utiliser la librairie Font Awesome :
 - a. Créez-vous un kit sur Font Awesome en suivant [ce lien](#) et ajoutez-le au projet ;
 - b. Vérifiez que ça fonctionne correctement en ajoutant une icône Font Awesome à votre projet.

Points de vigilance :

- En cas de difficultés, posez-vous les questions suivantes :
 - Le style CSS ne fonctionne pas correctement.
 - Avez-vous bien lié le fichier CSS dans le code HTML ?
 - La police ne s'affiche pas correctement.
 - Avez-vous importé les polices Google Fonts avant de les utiliser ?
 - Les icônes Font Awesome ne s'affichent pas correctement.
 - Avez-vous ajouté un kit Font Awesome au projet en amont ?
- Il y a des erreurs, des problèmes d'affichage ou bien quelque chose d'autre ne fonctionne pas dans votre code ? C'est une chose qui arrive souvent en développement. C'est pourquoi c'est important de savoir comment debugger son code en utilisant des outils adaptés tels que l'inspecteur de code. Pour cela, soyez attentif au chapitre "Testez votre code avec les devtools Chrome et Firefox" du cours "Mettez en place votre environnement front-end" donné dans les ressources.
- Assurez-vous que le repo GitHub soit public pour que votre mentor ait bien accès au projet.

Ressources :

- Le cours [Mettez en place votre environnement front-end](#) ;
- La [documentation officielle](#) pour utiliser Font Awesome.

Étape 2 : Découpez votre maquette

10 % de progression

Avant de démarrer cette étape, je dois avoir :

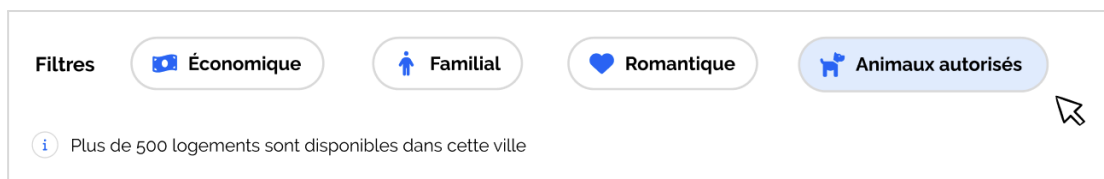
- Mis en place l'environnement de développement.

Une fois cette étape terminée, je devrais avoir :

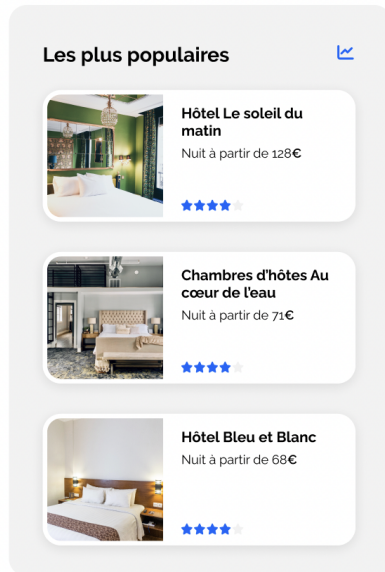
- Une maquette découpée, représentant la structure du code HTML.

Recommandations :

- Suivez le webinar [Découpez une maquette pour préparer l'intégration](#) pour réaliser la découpe de la maquette Booki.
- Posez-vous les questions suivantes :
 - Avez-vous pu identifier tout ce qui est visible sur la maquette (les éléments comme le logo, la fonction recherche, les cartes hébergements et activités etc.) ?
 - Comment sont regroupés les différents éléments ?
 - Pour chaque élément, quelle est la balise HTML qui lui est associée ?
- Définissez à quel moment positionner les éléments de façon horizontale, ou bien de façon verticale. En effet, chaque bloc ne contient que des éléments positionnés soit horizontalement. Exemple :



Soit verticalement, exemple :



Une fois que vous avez réalisé le découpage de votre maquette, discutez-en avec votre mentor. Cette étape va vous permettre de vous poser les bonnes questions, et de vérifier que vous n'avez rien oublié.

Points de vigilance :

Ne perdez pas trop de temps à réaliser un découpage "parfait". Le principal est d'avoir une vision rapide des principaux enjeux de la maquette.

Ressources :

- Le webinar [Découpez une maquette pour préparer l'intégration](#).
- Le cours [Créez une maquette pour le développement web](#).

Étape 3 : Intégrez le header du projet

20 % de progression

Avant de démarrer cette étape, je dois avoir :

- Réalisé le découpage de la maquette.

Une fois cette étape terminée, je devrais avoir :

- Le code de l'en-tête de la page.

Recommandations :

- Concentrez vous sur la version desktop du site afin que celle-ci soit pleinement fonctionnelle. Vous travaillerez sur les versions tablette / mobile et aborderez la notion "media queries" plus tard (Étape numéro 11).

- Pour réaliser le positionnement entre le logo Booki et les parties Hébergements/Activités, vous pouvez utiliser Flexbox (ou Grid).

Points de vigilance :

- Attention à ne pas oublier la bordure bleue qui s'affiche au survol.
- Attention, la bordure bleue s'affiche au-dessus en version desktop, et en dessous en version mobile.
- Distinguez les moments où il faut utiliser une propriété margin plutôt que padding.
- Il est préférable d'utiliser des pixels plutôt que des pourcentages pour les valeurs des marges et des paddings.
- Les balises HTML ont des propriétés CSS par défaut (par exemple, un titre "h1" est affiché en gras par défaut).
- Il faut savoir que par défaut, la balise "body" comporte des marges.

Ressources :

- Le chapitre "[Faites votre mise en page avec Flexbox](#)" du cours Apprenez à créer votre site web avec HTML5 et CSS3 ;
- Le petit jeu [Flexbox Froggy](#) pour vous entraîner à écrire du code CSS.

Étape 4 : Ajoutez le formulaire de recherche

30 % de progression

Avant de démarrer cette étape, je dois avoir :

- Intégré le header de la page.

Une fois cette étape terminée, je devrais avoir :

- Un formulaire de recherche intégré à la page HTML.

Recommandations :

1. Identifiez bien la construction de ce formulaire qui est constitué de 3 parties, comme on le voit sur la maquette.
2. Pour le moment, concentrez-vous sur la partie desktop, l'intégration de la partie responsive pourra se faire plus tard.
3. Ensuite, avec du CSS, il faudra afficher ou masquer le mot ou l'icône en fonction de l'écran utilisé.

Points de vigilance :

- Attention à ne pas appliquer une bordure sur l'intégralité du champ de recherche, pour que le visuel corresponde à la maquette.

- Attention au style de la barre de recherche : sur desktop, le bouton de recherche comprend le texte “Rechercher”, alors que sur mobile, c’est une loupe. Pour gérer cela, il va falloir intégrer les deux éléments (le mot + l’icône) en HTML.

Ressources :

- Le chapitre [Créez des formulaires](#) du cours Apprenez à créer votre site web avec HTML5 et CSS3.

Étape 5 : Ajout de la partie Filtres

35 % de progression

Avant de démarrer cette étape, je dois avoir :

- Intégré le formulaire de recherche de la page.

Une fois cette étape terminée, je devrais avoir :

- La partie Filtre entièrement réalisée.

Recommandations :

Avec un bon découpage de la maquette, vous devriez pouvoir intégrer cette partie correctement avec Flexbox.

Points de vigilance :

- Distinguez les moments où il faut utiliser une propriété margin plutôt que padding.
- Il est préférable d’utiliser des pixels plutôt que des pourcentages pour les valeurs des marges et des paddings. En effet, l'utilisation de pourcentages sur des marges/paddings ne correspondrait pas au résultat attendu, car les variations de taille risqueraient d’être trop fortes d’un format d’écran à un autre.

Ressources :

- Le chapitre [Découvrez le modèle des boîtes](#) du cours Apprenez à créer votre site web avec HTML5 et CSS3.

Étape 6 : Réalisez la “card” présente dans “Hébergements à Marseille”

40 % de progression

Avant de démarrer cette étape, je dois avoir :

- Intégré les filtres de la page.

Une fois cette étape terminée, je devrais avoir :

- Une première carte hébergement. Elle sera utile à l'étape 8.

Recommandations :

1. Identifiez la structure d'une carte "hébergement". Il y a plusieurs éléments, positionnés horizontalement et verticalement.
2. Si aucune propriété CSS n'est appliquée sur une image, celle-ci va s'afficher dans sa taille d'origine. Comme pour de très nombreux éléments, donner une largeur en "%" à l'image est une bonne idée. Il sera nécessaire de définir une hauteur en pixels.
3. Une fois dimensionnée, l'image devrait être déformée. La propriété CSS `object-fit` permettra de corriger cela.

Points de vigilance :

- Les images doivent être intégrées via le HTML.
- Ne pas oublier :
 - l'ombre portée sur la carte ;
 - les attributs alt ;
 - les border-radius sur les images.

Ressources :

- L'article [How to – Cards de W3C](#) avec quelques snippets de code pour réaliser une card ;
- L'article [MDN](#) sur la propriété "object-fit".

Étape 7 : Réalisez la "card" présente dans "Les plus populaires"

50 % de progression

Avant de démarrer cette étape, je dois avoir :

- Réalisé une première "card" présente dans "Hébergements à Marseille".

Une fois cette étape terminée, je devrais avoir :

- Les deux cards utilisées pour les parties "Hébergements à Marseille" et "Les plus populaires". Il sera alors temps de combiner le tout !

Recommandations :

1. Identifiez la structure d'une carte "Les plus populaires". Il y a plusieurs éléments, positionnés horizontalement et verticalement. Cette carte présente une forme plus horizontale que la carte "hébergement".
2. Si aucune propriété CSS n'est appliquée sur une image, celle-ci va s'afficher dans sa taille d'origine. Comme pour de très nombreux éléments, donner une largeur en "%" à l'image est une bonne idée. Il sera nécessaire de définir une hauteur en pixels.
3. Dans la partie droite de la carte, les textes sont positionnés en haut, et les icônes "étoiles" sont positionnées en bas de la carte.

Points de vigilance :

- Ne pas confondre le positionnement et les marges. Une marge permet de définir de petits décalages, mais ne sert pas à "positionner" les éléments. Positionner des éléments relève plus du rôle de Flexbox (exemple: "Je veux que ces deux éléments soient complètement espacés l'un de l'autre sur l'axe vertical).

Ressources :

- L'article [How to – Cards de W3C](#) avec quelques snippets de code pour réaliser une card.
- L'article [MDN](#) sur la propriété "object-fit".

Étape 8 : Gérez l’affichage des conteneurs “Hébergements à Marseille” et “Les plus populaires”

60 % de progression

Avant de démarrer cette étape, je dois avoir :

- Réalisé une première “card” présente dans “Hébergements à Marseille” et une première “card” présente dans “Les plus populaires”.

Une fois cette étape terminée, je devrais avoir :

- Une grande partie du site mise en page.

Il ne restera plus que la partie “Activités à Marseille” et le pied de page à réaliser.

Recommandations :

1. Utilisez flex et les pourcentages pour gérer l’affichage des deux blocs “Hébergements à Marseille” et “Les plus populaires”. Les ratios sont environ d’un tiers et deux tiers (penser à garder quelques pourcents pour l’espace entre les deux blocs). À vous de transformer ça en pourcentages.
2. Faites d’abord la mise en page entre les deux conteneurs avant de vous attaquer à la mise en page dans chacun des conteneurs.
3. Ensuite, ajoutez chacune des cards dans chacun des conteneurs avant de réaliser la mise en page.

Points de vigilance :

- Le responsive pourra encore être géré plus tard, mais gardez en tête que l’ordre d’affichage des deux conteneurs change en fonction de la version : mobile ou desktop.
- N’oubliez pas le titre, l’icône et le lien “Afficher plus”.
- Les couleurs de fond s’inversent entre la version mobile et la version desktop.

Ressources :

- Le chapitre [Faites votre mise en page avec Flexbox](#) du cours Apprenez à créer votre site web avec HTML5 et CSS3.

Étape 9 : Intégrez le conteneur “Activités à Marseille”

70 % de progression

Avant de démarrer cette étape, je dois avoir :

- Intégré les deux sections “Hébergements à Marseille” et “Les plus populaires” de la page.

Une fois cette étape terminée, je devrais avoir :

- Le projet presque terminé. Il ne restera plus que le footer à réaliser.

Recommandations :

1. Appuyez-vous sur les précédentes recommandations données pour la réalisation des différentes cartes.
2. Dans cette section, la structure est un peu différente puisque c'est une présentation en 4 colonnes.
3. La hauteur de chacune des activités est identique. Il faudra alors mettre en place (comme dans les étapes précédentes) une première activité, et intégrer les 3 autres à partir de la première.

Points de vigilance :

- Les images doivent être intégrées via le HTML et non le CSS.
- N'oubliez pas les attributs alt. Cela fait plusieurs fois que ce point est évoqué, mais il est important de le rappeler !

Ressources :

Les ressources de l'étape précédente vous seront utiles ici aussi.

Étape 10 : Implémentez le footer

80 % de progression

Avant de démarrer cette étape, je dois avoir :

- Intégré la section "Activités à Marseille" de la page.

Une fois cette étape terminée, je devrais avoir :

- Terminé le code du projet.

Il faudra maintenant vérifier si le code est conforme aux validateurs W3C.

Recommandations :

1. Une fois encore, vous pouvez utiliser Flexbox pour la mise en page.
2. Il faudra bien vous appuyer sur le découpage de votre maquette, et identifier les différents éléments ainsi que les différents blocs.
3. Le footer se présente sous 3 colonnes, chaque colonne ayant une taille identique aux deux autres.

Points de vigilance :

- Si vous utilisez des `ul` pour réaliser les liens, vous avez par défaut un padding-left qui s'applique. Pensez à l'enlever, car cela va provoquer un décalage par rapport à ce qui se trouve juste au-dessus.

Étape 11 : Mettez en place le responsive design

90 % de progression

Avant de démarrer cette étape, je dois avoir :

- La version desktop du site, pleinement fonctionnelle, se comportant correctement de la résolution 992 px à la résolution 1 400 px.

Une fois cette étape terminée, je devrais avoir :

- Un projet pleinement compatible avec toutes les tailles d'écran possibles.

Recommandations :

1. Pensez à définir une largeur maximum à 1 400 px pour gérer correctement les écrans avec une grande résolution.
2. Définissez bien les deux media queries (tablette, mobile) :
 - à la fin du code CSS ;
 - dans le bon ordre ; cela veut donc dire que dans le fichier CSS, de haut en bas, on devrait avoir :
 - le code CSS global,
 - puis la media queries tablette,
 - et la media queries mobile.

Points de vigilance :

- Pensez à implémenter meta viewport dans le code HTML.

Ressources :

- Le chapitre du cours HTML5 et CSS3 à propos des [media queries](#) ;
- L'article [Utilisation de la balise meta viewport pour contrôler la mise en page sur mobile](#).

Étape 12 : Vérifiez la qualité de votre code

100 % de progression

Avant de démarrer cette étape, je dois avoir :

- Terminé d'intégrer les maquettes, sur tous les formats d'écran.

Une fois cette étape terminée, je devrais avoir :

- Terminé le projet !

Il ne vous restera plus qu'à préparer la soutenance.

Recommandations :

1. Concentrez vos efforts sur les erreurs remontées. Vous pouvez regarder les warnings, mais vous n'êtes pas obligé de les traiter.
2. Faites attention au nommage du code (des classes CSS, par exemple). Vous pouvez utiliser l'anglais ou le français, mais évitez de mixer les deux langues.
3. Utilisez le kebab case, par exemple ``.main-wrapper`` . C'est LA convention CSS la plus répandue.
4. Préférez généralement l'utilisation des pixels pour les margins/paddings, et les pourcentages pour les widths, dans le but de permettre la bonne gestion des différentes résolutions. Dans certains cas, l'utilisation de pixels pour les widths peut être quelque chose de pertinent (gestion de la taille d'une icône, par exemple).

Ressources :

- Les outils de validation de code :
 - [Valideur HTML du W3C](#) ;
 - [Valideur CSS du W3C](#).

Projet terminé !