

Auto-évaluation

Premiers pas de développement mobile en Swift



Un dernier doute avant l'envoi de vos livrables ?

Pour vérifier la qualité de votre travail :

- cochez les cases ci-dessous : elles indiquent que vous avez bien pris en compte chaque indicateur de réussite ;
- renseignez, si besoin, la colonne "Notes" avec des commentaires sur vos livrables / vos étapes. Ils seront des points de discussion avec votre mentor pendant votre session de bilan.

Quand toutes les cases de ce document seront cochées, vous pourrez déposer vos livrables sur la plateforme.

Bonne réussite !

Étape	Indicateurs de réussite de l'activité	Notes
Étape 1 : Initialisez le projet sur XCode	<input type="checkbox"/> La dernière version est installée sur l'ordinateur <input type="checkbox"/> Le projet a été initialisé avec le nom "AimantsAnimaux" <input type="checkbox"/> Swift a été bien choisi pour le langage <input type="checkbox"/> SwiftUI a été bien choisi pour l'interface	
Étape 2 : Parcourez les fichiers de base	<input type="checkbox"/> J'ai commencé à comprendre les fichiers de base OU J'ai exploré tous les fichiers de base <input type="checkbox"/> Le projet a été lancé sur l'iPhone 15 dans le simulateur	
Étape 3 : Ajoutez un nouveau fichier au projet	<input type="checkbox"/> Les 3 fichiers ont été créés, correspondant aux 3 écrans de l'application en utilisant le template SwiftUI View <input type="checkbox"/> Le 4 ^e fichier a été créé de type Swift File et porte le nom 'ViewModel' <input type="checkbox"/> Le projet compile avec un "Build succeeded"	
Étape 4 : Ajoutez un fichier existant au projet	<input type="checkbox"/> Le dossier "Images.xcassets" a bien été ajouté au projet	

<p>Étape 5 : Affichez les nouvelles vues</p>	<ul style="list-style-type: none"> <input type="checkbox"/> Le bug a été bien corrigé, c'est-à-dire que dans le fichier "AimantsAnimauxApp", l'appel à la structure "ContentView" a été remplacé par "AnimalsListView" <input type="checkbox"/> La vue de la liste des animaux s'affiche bien <input type="checkbox"/> Lorsque l'on clique sur un animal de la liste, la vue détail s'affiche correctement <input type="checkbox"/> Depuis la liste, lorsqu'on appuie sur "Ajouter" en haut à droite, on arrive sur la vue qui permet d'ajouter un animal 	
<p>Étape 6 : Modifiez du code existant</p>	<ul style="list-style-type: none"> <input type="checkbox"/> Sur le simulateur, sur la vue détaillée d'un animal, le cercle autour de l'image de patte de chien s'affiche <input type="checkbox"/> Les lignes du code en gras ont été ajoutées : <pre> ```code swift : Image(systemName: getImageName(breed: animal.breed)) .resizable() .aspectRatio(contentMode: .fill) .foregroundColor(.green) .padding() .clipShape(Circle()) .overlay { Circle().stroke(.green, lineWidth: 5) } .frame(width: 100, height: 100) ``` </pre> 	

<p>Étape 7 : Ajoutez un nouvel élément</p>	<ul style="list-style-type: none"> <input type="checkbox"/> Sur l'écran détail d'un animal, on peut voir le champ taille avec la valeur correspondante <input type="checkbox"/> Sur le formulaire d'ajout d'un animal, on voit le champ taille 	
<p>Résultats finaux</p>	<ul style="list-style-type: none"> <input type="checkbox"/> L'application se lance correctement après avoir importé les fichiers et structures dans XCode <input type="checkbox"/> L'application fonctionne sans erreurs ni plantages <input type="checkbox"/> La fonctionnalité d'ajout d'un animal se fait depuis la liste des animaux sur la barre de navigation <input type="checkbox"/> Les écrans affichent correctement les informations, conformément aux maquettes 	