

TEMPLATE

Portfolio Upwork qui Convertit

Votre portfolio sur Upwork est votre vitrine. Les clients le regardent AVANT de vous contacter. 3 projets bien présentés > 10 projets mal présentés. Voici comment structurer chaque projet.

STRUCTURE D'UN PROJET PORTFOLIO

1. TITRE DESCRIPTIF (max 60 caractères)

- MAUVAIS : 'Mobile App'
- BON : 'E-Commerce Mobile App - Flutter | iOS & Android'

Le titre doit dire QUOI vous avez fait + TECHNOLOGIE utilisée.

2. SCREENSHOT/IMAGE ATTRACTIVE

- Utilisez screenshot HAUTE QUALITÉ du produit fini
- Ou mockup professionnel (Figma, Canva)
- Évitez : code source, écrans vides, captures floues
- Idéal : montrer le produit en situation d'utilisation

3. DESCRIPTION STRUCTURÉE (200-300 mots)

Suivez ce template en 4 parties :

a) LE PROBLÈME (1-2 phrases)

"My client needed a mobile app to [specific problem/goal]."

b) LA SOLUTION (3-4 phrases)

"I developed a [type of solution] using [technologies]. The app features [key features]. I implemented [technical highlights]."

c) TECHNOLOGIES UTILISÉES (liste à puces)

- Flutter & Dart
- Firebase (Auth, Firestore, Cloud Functions)
- Stripe payment integration
- Google Maps API

d) RÉSULTATS (1-2 phrases avec CHIFFRES si possible)

"The app was successfully launched on both iOS and Android. The client reported a 40% increase in user engagement after launch." OU "Delivered on time and within budget. Client rated 5 stars."

4. TAGS/SKILLS PERTINENTS

Ajoutez 5-8 tags en lien avec le projet :

- Technologies : Flutter, React, Python, etc.

- Domaines : Mobile App Development, E-commerce, etc.
- Évitez tags génériques : 'Programming', 'Software' (trop vague)

EXEMPLE CONCRET - Projet Ismael

TITRE

Restaurant Ordering App - Flutter | Firebase | Stripe Payment

DESCRIPTION

THE PROBLEM

My client, a restaurant owner in Los Angeles, needed a mobile ordering app to reduce wait times and increase takeout orders during the pandemic.

THE SOLUTION

I developed a cross-platform mobile app using Flutter that allows customers to browse the menu, customize orders, and pay securely through Stripe. The app features real-time order tracking, push notifications, and a loyalty rewards system. I implemented Firebase for backend services including authentication, database, and cloud functions for order processing.

TECHNOLOGIES USED

- Flutter & Dart for cross-platform development
- Firebase (Authentication, Firestore, Cloud Functions)
- Stripe API for secure payments
- Google Maps SDK for delivery tracking
- OneSignal for push notifications

RESULTS

The app was successfully deployed to both App Store and Google Play. Within 2 months, the restaurant saw a 60% increase in takeout orders. The client provided a 5-star review and hired me for ongoing maintenance.

TAGS

Flutter, Mobile App Development, Firebase, Stripe Integration, iOS Development, Android Development, Restaurant App, E-commerce

CONSEILS POUR UN PORTFOLIO QUI CONVERTIT

- **Qualité > Quantité** : 3 beaux projets > 10 projets médiocres
- **Variété** : Montrez différents types de projets si possible
- **Mise à jour régulière** : Ajoutez vos meilleurs nouveaux projets
- **Zéro faute** : Relisez 3 fois, faites relire par quelqu'un
- **Résultats concrets** : Chiffres, témoignages, impact mesurable
- **Projets terminés uniquement** : Pas de WIP ou abandonnés
- **Évitez projets personnels basiques** : 'To-Do List', 'Calculator' (trop junior)
- **Pas de code source brut** : Clients veulent voir RÉSULTAT, pas code

■ ASTUCE D'ISMAEL

"Au début, je n'avais pas de vrais projets clients. J'ai fait 3 projets fictifs MAIS réalistes : une app de livraison, une app de fitness, une app de gestion de budget. J'ai fait des beaux designs sur Figma, implémenté les fonctions principales, pris des beaux screenshots. Résultat : mes premiers clients ont été convaincus par la qualité visuelle. Une fois que j'ai eu de vrais projets clients, j'ai remplacé les projets fictifs."

Démarrer avec 2-3 projets fictifs de qualité > attendre d'avoir des vrais clients pour créer portfolio.