

# MODELE D'UN CAHIER DES CHARGES D'UN PROJET LOGICIEL

## Contexte et définition du problème

Solliciter les services d'un artisan, plombier, électricien, menuisier, ...a toujours posé des problèmes pour les personnes demandant de tels services. La crédibilité, La compétence, le rapport qualité-prix ont toujours été des objectifs parfois contradictoires. La société X a pensé à ce dilemme en proposant une application permettant d'outiller les internautes avec une base informationnelle sur les artisans, leurs offres, leurs disponibilités et leur efficacité. Entre autres, ce que les gens pensent d'eux à travers les avis et les commentaires.

## Objectif du projet

Création d'une application web permettant de présenter les artisans et leurs travaux et d'en faciliter leur contact et la prise des devis.

## Cibles du projet

Cette application s'adresse :

- ✓ Aux artisans pour se présenter et élargir leur clientèle
- ✓ Aux utilisateurs, toute personne demandant un service de ces artisans, pour s'en informer et leur demander des devis d'intervention
- ✓ A l'administrateur de la plateforme permettant de tenir à jour les informations du système et surveiller les avis et les commentaires

## Description fonctionnelle des besoins

La future application doit satisfaire la liste des besoins fonctionnels suivants :

- ✓ Authentification.
- ✓ Création d'un compte : l'utilisateur et l'artisan peuvent s'inscrire.
- ✓ Mise à jour du compte :
  - L'admin peut modifier ses informations.
  - L'utilisateur peut modifier ses informations.
  - L'artisan peut modifier ses informations.
- ✓ Consultation des artisans : l'admin ou l'utilisateur ou l'artisan peut consulter les artisans.
- ✓ Gestion des artisans :
  - L'admin peut ajouter les artisans.
  - L'admin peut modifier les informations des artisans.
  - L'admin peut bloquer les artisans.
- ✓ Gestion des portfolios :
  - L'admin ou l'artisan peut ajouter les portfolios.
  - L'admin ou l'artisan peut modifier les informations dans les portfolios.
  - L'admin ou l'artisan peut supprimer les portfolios.
- ✓ Gestion des villes :
  - L'admin peut ajouter les villes.
  - L'admin peut supprimer les villes.
- ✓ Gestion des gouvernorats :
  - L'admin peut ajouter les gouvernorats.
  - L'admin peut supprimer les gouvernorats.
- ✓ Gestion des utilisateurs :
  - L'admin peut ajouter les utilisateurs.
  - L'admin peut modifier les informations des utilisateurs.
  - L'admin peut bloquer les utilisateurs.
  - L'admin peut consulter les utilisateurs.
- ✓ Suppression des avis et des commentaires : l'administrateur peut nettoyer la plateforme de tout commentaire abusif
- ✓ Demande de devis : l'utilisateur peut demander un devis selon son besoin.
- ✓ Evaluation des artisans : l'utilisateur peut donner son avis ou commenter le travail de l'artisan.
- ✓ Recherche des artisans :
  - Le visiteur peut filtrer les artisans selon son besoin.
  - Le visiteur peut filtrer les artisans par localisation.

- Le visiteur peut filtrer les artisans par catégorie.

## Description NON fonctionnelle des besoins

Plusieurs contraintes doivent être respectées à travers ladite application telles que :

- ✓ L'ergonomie : notre application doit présenter une interface simple pour que l'utilisateur ait une manipulation aisée.
- ✓ La performance : notre application doit être performante c'est-à-dire à travers ses fonctionnalités, répondre à toutes les exigences des usagers d'une manière optimale et dans un temps de réponse raisonnable.
- ✓ La convivialité : Les interfaces graphiques de l'application doivent être adaptables aux différents écrans des appareils mobiles.
- ✓ Sécurité :
  - L'accès à l'application doit être sécurisé par un login et un mot de passe
  - Les mots de passe de connexion sont sécurisés par une fonction de hachage avant d'être stockés dans la base.
  - L'application doit fournir une sécurité des sessions des utilisateurs dans les normes requises.
  - Le système doit garantir à l'utilisateur l'intégrité et la confidentialité de ses données.