

Une application web simple de gestion de tickets

L'objectif est de concevoir une application web permettant de gérer des tickets.

L'architecture back-end peut rester simple, mais **l'accent doit être mis sur le front-end**.

🎯 Fonctionnalités principales

✓ 1. Affichage de la liste des tickets

Afficher les informations suivantes :

- Titre
- Description
- Date de création
- Statut
- Actions (modifier statut, supprimer)

L'affichage peut se faire :

- Sous forme de **cartes (cards)**

✓ 2. Ajout d'un ticket

Un formulaire doit permettre de créer un ticket avec :

- Titre
- Description
- Statut (Ouvert, En cours, Résolu)

Le formulaire doit s'ouvrir dans **une modale**.

✓ 3. Changement de statut

Possibilité de modifier le statut d'un ticket selon le cycle :
ouvert → en cours → résolu

✓ 4. Filtrage des tickets

Un filtre par statut doit être disponible dans la liste :

- Tous
 - Ouvert
 - En cours
 - Résolu
-

Objectifs UI/UX

L'interface doit être :

- **Esthétique**
 - **Responsive**
 - **Cohérente visuellement**
-

Stack front-end attendue

En utilisant:

- **React**

(Dans ton cas : React + TypeScript)



Wireframe UI – Gestion de tickets

▀ Page principale : Liste des tickets

◆ Header (barre haute)

- titre à gauche
- Nom de l'utilisateur à droite
- Bouton « **Ajouter un ticket** » → en haut à droite

◆ Filtre par statut

- Dropdown ou boutons :
`Tous | Ouvert | En cours | Résolu`

◆ Liste des tickets

Sous forme de tableau ou de cartes :

- Titre
- Description courte
- Statut (badge coloré)
- Date de création
- Actions :
 - **Changer statut**
 - **Supprimer**



▀ Modal : Ajouter un ticket

Champs :

- **Titre** → Input
- **Description** → Textarea
- **Statut** → Dropdown (Ouvert, En cours, Résolu)

Boutons :

- **Créer**
 - **Annuler**
-

Consignes techniques

Back-end

- Le back-end doit permettre :
 - l'ajout d'un ticket,
 - la récupération des tickets,
 - la modification du statut,
 - la suppression.
- Le formulaire doit se fermer automatiquement après création.

Stack backend:

- Langage & framework de ton choix :
Node.js/Nestjs

Base de données :

- Relationnelle : **MySQL**