

BELDILMI Guillaume
CORBEL Théo
HAMMANI Assia
GENCE Elio
MAJIDI Imane
MEZIANE Rayene

Conception Web Avancée

Rapport de projet

```

                               BBBBBBBBBBBBBBBBBB
                                B:::::::::::::B
                                B:::::BBBBBB:::::B
                                BB:::::B      B:::::B
uuuuuuu      uuuuuuu      B:::::B      B:::::B
u:::::u      u:::::u      B:::::B      B:::::B
u:::::u      u:::::u      B:::::BBBBBB:::::B
u:::::u      u:::::u      B:::::::::::::BB
u:::::u      u:::::u      B:::::BBBBBB:::::B
u:::::u      u:::::u      B:::::B      B:::::B
u:::::u      u:::::u      B:::::B      B:::::B
u:::::uuuuu:::::u      B:::::B      B:::::B
u:::::::::::::uuBB:::::BBBBBB:::::B
u:::::::::::::uB:::::::::::::B
uu:::::::::uu:::::uB:::::::::::::B
uuuuuuuuu      uuuuBBBBBBBBBBBBBBBBBB
```

Présentation du projet

Nombre d'étudiants : Environ 6 par groupe

Description du Projet : L'application GesTaches permettra aux utilisateurs de créer, afficher, mettre à jour et supprimer des tâches à faire. Chaque tâche aura un intitulé, une date de création, une date d'échéance, une description et une priorité. Les utilisateurs pourront également marquer une tâche comme terminée. A vous de choisir dans quel cadre (Gestion Personnelle, Travail Professionnel, Éducation et Études, Gestion de Projet, Santé et Bien-être Organisation des Événements, Gestion des Tâches Domestiques, ...)

Fonctionnalités du Projet :

1. Page d'Accueil :

- Affiche la liste des tâches à faire.
- Permet de filtrer les tâches par statut (en cours, terminées) et par priorité (élevée, moyenne, faible).
- Permet de trier les tâches par date d'échéance ou par priorité.

2. Page de Création de Tâches :

- Permet aux utilisateurs de créer de nouvelles tâches en spécifiant un nom, une date d'échéance et une priorité.
- Valide les entrées utilisateur et affiche des messages d'erreur en cas d'erreur de saisie.

3. Page de Détails de Tâche :

- Affiche les détails d'une tâche, y compris son nom, sa date d'échéance, sa priorité et son statut.
- Permet aux utilisateurs de marquer la tâche comme terminée.

4. Page de Modification de Tâche :

- Permet aux utilisateurs de modifier les détails d'une tâche existante, y compris son nom, sa date d'échéance, sa priorité et son statut.
- Valide les entrées utilisateur et affiche des messages d'erreur en cas d'erreur de saisie.

5. Page de Suppression de Tâche :

- Permet aux utilisateurs de supprimer une tâche existante après confirmation.

Technologies Requises :

- Angular (avec Angular CLI) pour le développement de l'application front-end.
- Angular Router pour la gestion de la navigation entre les pages.
- Services et composants pour organiser le code.
- Stockage des données : Backend (optionnelle) qui permet de traiter les données avec une base de données. Vous pouvez aussi gérer les données avec des fichiers JSON.

- HTML, CSS (avec des styles Bootstrap, materializecss ou personnalisés).
- Validation de formulaire pour garantir des entrées utilisateur valides.

Livrables requis : Chaque groupe devra définir le cadre choisi et doit soumettre l'application finale accompagnée d'un rapport explicatif qui détaille les divers concepts implémentés dans votre projet

Cadre et implémentation choisie

Pour ce projet, nous avons choisi de développer une application de gestion de tâches dans le cadre de la gestion personnelle (c'est-à-dire une "to-do list"). Tel que nous le demandait le sujet, nous avons utilisé Angular pour le développement de l'application front-end, avec Angular Router pour la gestion de la navigation entre les pages. Nous avons également utilisé des services et des composants pour organiser le code. Ainsi, nous avons créé un service pour la gestion des tâches chargé de la gestion des tâches (sélection, création, modification, suppression, etc.) ainsi que la sauvegarde et le chargement des tâches depuis le un stockage (via un fichier JSON). Nous avons également créé des composants pour chaque page de l'application (liste des tâches, création, détails, modification et suppression de tâches).

Fonctionnement de l'application

L'application est composée de 5 composants (un pour chaque page) et d'un service pour la gestion des tâches. Le service est injecté dans chaque composant pour permettre la gestion des tâches depuis chaque page.

Le service de gestion des tâches (`manager.service`)

Le service de gestion des tâches contient la classe `Task` qui représente une tâche, elle sera la base de l'application. Cette classe contient les attributs suivants :

- `name` : le nom de la tâche
- `description` : la description de la tâche
- `priority` : la priorité de la tâche (valeurs possibles : `Low` (faible), `Medium` (moyenne) ou `High` (élevée))
- `status` : le statut de la tâche (valeurs possibles : `Ongoing` (en cours) ou `Done` (terminée))
- `dueDate` : la date d'échéance de la tâche (au format `YYYY-MM-DD`)
- `creationDate` : la date de création de la tâche (au format `YYYY-MM-DD`)

De plus, le service contient la classe singleton `ManagerService` qui permet de gérer les tâches à partir de chaque composant. Elle contient les attributs privés suivants :

- `tasks` : un tableau contenant toutes les tâches

- `selectedTaskId` : l'index de la tâche sélectionnée (cela permet de garder en mémoire la tâche sélectionnée lors de la navigation entre les pages)

Elle contient également les méthodes suivantes :

- `setSelectedTaskId(id: number)` : permet de définir l'index de la tâche sélectionnée
- `getSelectedTaskId()` : permet de récupérer l'index de la tâche sélectionnée
- `addTask(task: Task)` : permet d'ajouter une tâche à la liste des tâches
- `getTasks()` : permet de récupérer le tableau contenant toutes les tâches
- `getTask(id: number)` : permet de récupérer une tâche à partir de son index
- `removeTask(id: number)` : permet de supprimer une tâche de la liste à partir de son index
- `loadTasks()` : permet de charger la liste des tâches depuis le fichier JSON
- `saveTasks()` : permet de sauvegarder la liste des tâches dans le fichier JSON

Remarque : chaque modification de la liste des tâches (ajout, suppression, modification) est suivie d'une sauvegarde de la liste dans le fichier JSON.

Les composants

Chaque component correspond à une page de l'application. Ils sont tous liés au service de gestion des tâches pour permettre la gestion des tâches depuis chaque page.

Chaque page est structurée de la manière suivante : le titre de la page suivi d'un panel de commande (avec la classe `cmd-panel`, contenant les divers boutons d'action, filtres, etc. propre au contexte de la page) et d'un panel de contenu (avec la classe `<component>-container`, contenant le contenu de la page). Le panel de commande est présent sur toutes les pages, tandis que le panel de contenu est différent selon la page. Cette structure permet de garder une certaine cohérence entre les pages et ainsi de faciliter l'écriture du code de style.

Le component `list.component`

Le contenu de ce composant est composé d'une liste des tâches, chaque tâche est représentée par un bouton qui affiche le nom de la tâche, sa priorité, son statut et sa date d'échéance. Un clic sur ce dernier permet d'afficher la page de détails de la tâche. Le panel de commande est composé d'un bouton "Create" qui permet d'aller à la page de création de tâche, et de quatre listes déroulantes qui permettent de filtrer les tâches par statut (en cours, terminées) et par priorité (élevée, moyenne, faible) et de trier les tâches par date d'échéance ou par priorité en ordre croissant ou décroissant. Chaque modification de valeur dans ces listes déroulantes entraîne un tri de la liste des tâches.

Le component `create.component`

Le contenu de ce composant est composé d'un formulaire de création de tâche sur lequel il est

demandé à l'utilisateur de saisir le nom, la description, la priorité et la date d'échéance de la tâche, à la fin de ce dernier se trouve un bouton "Confirm" qui permet de créer la tâche. Le panel de commande est juste composé d'un bouton "Back" qui permet de revenir à la page d'accueil.

La validation du formulaire vérifie que l'échéance est bien une date valide et qu'aucun champ du formulaire n'est vide. Si une erreur est détectée, un message d'erreur est affiché grâce à l'appel de la fonction `alert()` de JavaScript.

Le component `details.component`

Le contenu de ce composant est composé des détails de la tâche sélectionnée (nom, description, priorité, statut, date de création et date d'échéance). Le panel de commande est composé, quant à lui, d'un bouton "Back" qui permet de revenir à la page d'accueil, d'un bouton "Set as done" qui permet de marquer la tâche comme terminée (qui ne s'affiche que si la tâche est encore en cours) et de deux boutons "Edit" et "Delete" qui permettent respectivement d'aller à la page de modification et de suppression de la tâche.

Le component `edit.component`

Le contenu de ce composant est composé d'un formulaire de modification de tâche dont tous les champs sont pré-remplis avec les détails de la tâche sélectionnée (nom, description, priorité, statut et date d'échéance), un bouton "Confirm" en bas de ce dernier permet de sauvegarder les modifications. Le panel de commande est juste composé d'un bouton "Cancel" qui permet de revenir à la page de détails de la tâche sans sauvegarder les modifications.

La validation du formulaire est la même que celle du formulaire de création de tâche.

Le component `delete.component`

Ce component est assez simple, il affiche simplement un message de demande de confirmation de suppression de la tâche sélectionnée en affichant ses détails. Dans son panel de commande, il contient un bouton "Delete" qui permet de supprimer la tâche sélectionnée et un bouton "Cancel" qui permet de revenir à la page de détails de la tâche.

Le routing

Le routing de l'application permet de naviguer entre les différentes pages de l'application. Il est défini assez simplement selon le schéma suivant : le chemin `/` correspond à la page d'accueil (la liste des tâches), et les chemins `/create`, `/details`, `/edit` et `/delete` correspondent respectivement aux pages de création, de détails, de modification et de suppression de tâches.