

Todo List

Etape #07 : Form Refactoring

- Créez une branche **etape07** dans votre dépôt Git et passez sur cette branche.
- Afin d'éviter les duplications de code, faites en sorte que la classe de widget **TaskForm** soit à la fois employé pour l'ajout et la modification d'un objet de type **Task**.

Pour ce faire :

- Puisque **TasksForm** ne constituera dorénavant plus un écran à part entière, mais un composant intégré à un écran, supprimez le widget **Scaffold** de **TaskForm**.
- Ajoutez 2 paramètres au widget **TaskForm** :
 - **formMode** (valeur requise, de type **enum Edit** ou **Add**),
 - **task** (valeur optionnelle mais utile si **formMode** est égale à **Add**, de type **Task**)
- Intégrez le widget **TaskForm** dans **TaskDetails** (ex: en tant que valeur de l'attribut **body** du widget **Scaffold** de **TaskDetails**).
- Afin d'améliorer l'**UX**, affichez le widget **TaskForm** (en mode **Add**) sous forme de fenêtre modale à l'aide d'un widget de type **Dialog** dans l'écran **TasksMaster** lorsque l'utilisateur clique sur le **FloattingActionButton** (cf. <https://api.flutter.dev/flutter/material/Dialog-class.html>).
- Le fonctionnement de **TaskForm** devra s'adapter au mode d'utilisation (ajout / modification), et afficher / masquer les champs de façon cohérente (ex : en cas d'ajout, le champ correspondant à l'attribut **completed** n'a pas à être affiché).