**Document Client - Application "To-Do List"**

**Introduction**

Ce document a pour but de présenter l'état actuel de l'application "To-Do List", développée dans le cadre du projet final. Il inclut une description des fonctionnalités implémentées, des fonctionnalités manquantes, ainsi que des propositions d'amélioration pour des versions futures. Ce projet a été réalisé en utilisant le framework **Qt** et **QML** et vise à fournir aux utilisateurs une interface simple et intuitive pour gérer leurs tâches quotidiennes.

**Architecture logicielle**

L’étude des cas d’utilisation nous à mené aux différentes fonctionnalités nécessaires à l’application.  
On y retrouve par exemple la possibilité de modifier une tâche existante mais aussi des options réglables par l’utilisateur.

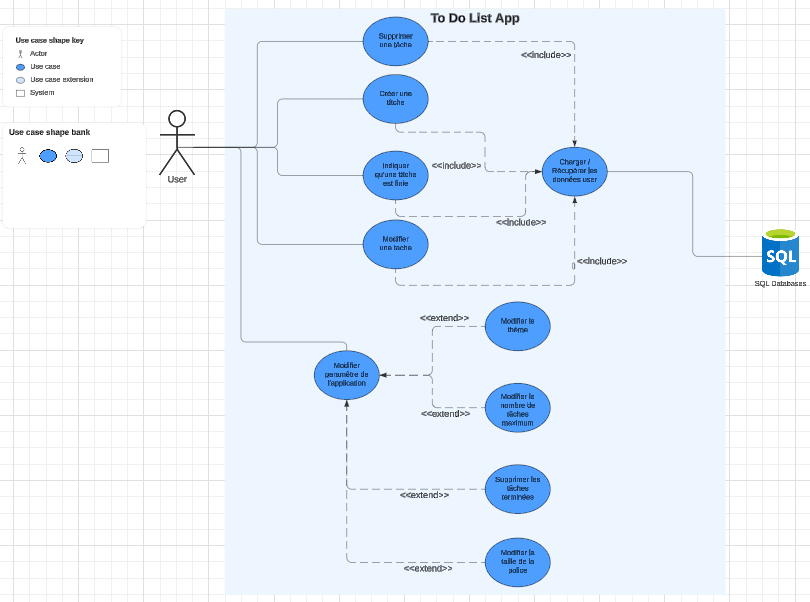


Figure 1 : Diagramme de cas d’utilisation

A partir de la liste des fonctionnalités nous avons pu réaliser un diagramme de classe afin d’identifier les différentes page de l’application ainsi que leurs relations avant de passer à l’implémentation.

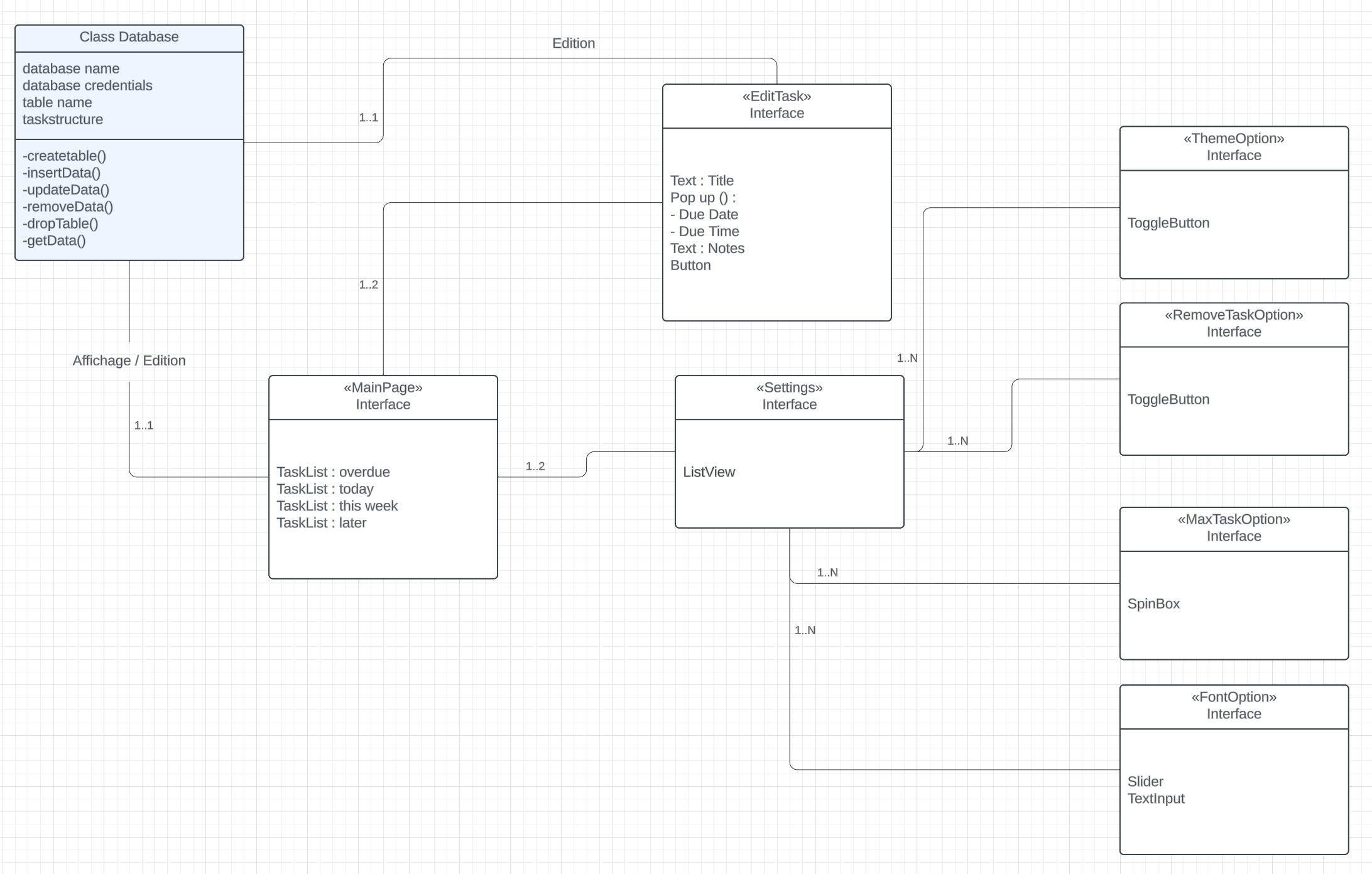


Figure 2 : Diagramme de classes

**Fonctionnalités Implémentées**

**1. Interface Utilisateur**

L'application "To-Do List" utilise une interface utilisateur ergonomique, adaptée à tous types d'utilisateurs grâce à son design intuitif. Voici les éléments principaux :

* U**ne seule fenêtre** : L’application ouvre une seule fenêtre.
* **Taille de fenêtre adaptable** : L'application dispose d'une taille de fenêtre minimale pour assurer une bonne lisibilité. L'utilisateur peut également passer en plein écran.
* **Navigation** fluide entre les pages de l’application.

**2. Gestion des Tâches**

Les fonctionnalités suivantes ont été implémentées pour la gestion des tâches :

* **Ajout de tâches** : Les utilisateurs peuvent ajouter de nouvelles tâches en spécifiant un nom, une date et une heure, ainsi que des détails supplémentaires si nécessaire. Le bouton d'ajout est présent dans l'interface principale pour un accès rapide. Il est également possible d’ajouter une tâche de façon rapide en utilisant la zone présente en haut de la fenêtre. Dans ce dernier cas la date sera automatiquement celle du jour.
* **Organisation des tâches** : Les tâches sont automatiquement classées en trois catégories :
  + "Aujourd'hui"
  + "Cette semaine"
  + "Plus tard"
* **Edition de tâches**: L’utilisateur peut modifier une tâche existante
* **Validation de la date** : L'utilisateur ne peut pas ajouter une tâche pour une date antérieure à la date actuelle.
* **Checkbox pour les tâches complétées** : Chaque tâche dispose d'une case à cocher pour marquer la tâche comme complétée. Une tâche complétée est visuellement distinguée (grisée).

**3. Paramètres**

L'onglet "Paramètres" permet à l'utilisateur de personnaliser son expérience :

* **Changer le thème** (clair/sombre).
* **Changer la taille de la police** pour améliorer l'accessibilité.
* **Supprimer les tâches complétées** : Une option permet de supprimer toutes les tâches qui ont été cochées comme complétées.
* **Limiter le nombre de tâches** : L'utilisateur peut définir un nombre maximum de tâches par catégorie.

**4. Ajout et gestion détaillée des tâches**

Lors de l'ajout d'une tâche, l'utilisateur a accès à des options supplémentaires :

* **Nom de la tâche**.
* **Date et heure de la tâche** via des sélecteurs intégrés(calendrier + horloge).
* **Détails de la tâche** dans un champ de texte multi-lignes avec un bouton pour valider la saisie.

**5. Performance et Compatibilité**

* L'application a été testée pour être **performante**, même avec un grand nombre de tâches.
* L'application est **compatible** avec les principales plateformes de bureau (Windows, macOS).

**6. Installation via Qt Installer Framework**

L'installation de l'application via un **Qt Installer Framework** a été livrée pour cette version du projet.

**Fonctionnalités Manquantes**

Malgré les fonctionnalités principales implémentées, certaines fonctionnalités mentionnées dans le cahier des charges n'ont pas été intégrées dans cette version de l'application :

**1. Persistance des Données en C++**

Le cahier des charges mentionnait l'**option** d'intégrer un système de gestion de base de données en C++ (pour stocker les tâches dans un fichier ou une base de données SQL). Cette fonctionnalité n'a pas été implémentée dans cette version de l'application, ce qui signifie que les tâches ne sont pas sauvegardées entre les sessions de l'application.

**Propositions d'Amélioration**

**1. Sauvegarde des Données**

Dans une future version de l'application, il est recommandé d'intégrer une fonctionnalité de **persistance des données**. L'implémentation d'une base de données locale (via SQLite, par exemple) permettrait à l'utilisateur de retrouver ses tâches même après avoir fermé l'application.

**2. Multilingue**

L'application pourrait être améliorée avec l'ajout d'une **option multilingue**. Cette fonctionnalité permettrait aux utilisateurs non francophones de choisir leur langue préférée.

**3. Synchronisation Multi-Plateforme**

Une version ultérieure pourrait inclure une **synchronisation** des tâches entre plusieurs appareils, permettant à l'utilisateur d'accéder à ses tâches depuis n'importe quel ordinateur, via une connexion à un compte.

**Conclusion**

L'application "To-Do List" dans sa version actuelle fournit les fonctionnalités principales de gestion des tâches attendues et respecte la plupart des exigences fonctionnelles définies dans le cahier des charges. Bien que certaines fonctionnalités plus avancées, comme la persistance des données en C++ , ne soient pas implémentées, l'application reste fluide, intuitive et adaptée à la gestion quotidienne des tâches.

Les améliorations proposées dans ce document offrent une feuille de route claire pour les futures versions de l'application.