Клиент-серверное приложение

“Корпоративный планировщик задач сотрудников”.

Назначение:

Постановка и контроль выполнения задач компании с возможностью доступа к ним через внутренний сервер компании.

Состав приложения:  
1. Клиентская часть с GUI интерфейсом и возможностью подключения к серверу.

2. Серверная часть с подключенной базой данных и возможностью обработки запросов от клиентской части.

Подробнее о клиентской части:

Интерфейс пользователя создается с применением фреймворка Qt.

Зайти в приложение можно либо под обычным сотрудником компании, либо под администратором.

Во время попытки входа проверяется связка логин/пароль через отправку этих данных на сервер.

Администратор и сотрудник видят разный интерфейс окна.

Возможности сотрудника:

1. Запрос от сервера и просмотр всех задач.
2. Просмотр списка только своих задач.
3. Создание новой задачи с назначением на себя.
4. Взятие чужой задачи на себя. Назначить свою задачу другому нельзя.
5. Изменение статуса своей задачи.
6. Контроль оставшегося времени на выполнение.

Возможности администратора:

1. Запрос от сервера списка сотрудников.
2. Создание нового сотрудника.
3. Создание задачи и назначение её на любого сотрудника.
4. Изменение статуса любой задачи.
5. Возможность сдвига времени выполнения (дедлайна).
6. Отправка на сервер команды на завершение работы серверной части.

Подробнее о серверной части:

Графической части нет. После включения работает в режиме обработки запросов от клиента.

Управляет базой данных, в которой хранятся данные о сотрудниках и назначенных задачах.

Проверяет данные авторизации – логина и пароля.

По запросу от клиентской части, отправляет данные, связанные с сотрудниками и задачами, а также записывает в базу изменения по этим данным.

Думаю, постараться сделать так, чтобы запросы от клиентов могли обрабатываться в несколько потоков на сервере, а доступ к базе защитить мьютексом или иным способом.

Базу планирую реализовать SQL или NoSQL – Mongo DB. Не решил ещё.

Если получиться, реализовать на клиенте отдельный поток, который будет периодически отправлять в фоне запрос на сервер на наличие новых задач, и если такие есть как-то визуально оповещать пользователя.

Скорее всего, клиентская и серверная часть будут в разных репозиториях.

Для серверной части планируется, как обычно, сделать автоматическую сборку через github, т.к. там будет только C++, а вот для клиентской части автоматическая сборка может не получиться, т.к. там планируется использовать фреймворк Qt.