## Предисловие

Я понимаю, что в моей реализации можно применить пару паттернов, которые я не применил, но в следствие срока я их не реализовал, прошу сильно не бить палками. ~~Как говорится, дальше платно.~~

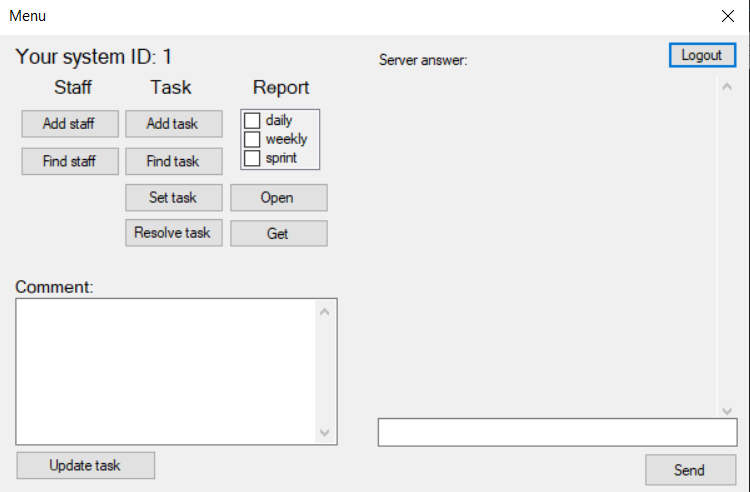
Моё решение представлено в виде клиентского приложения, сервера и библиотеки типов.

## Туториал

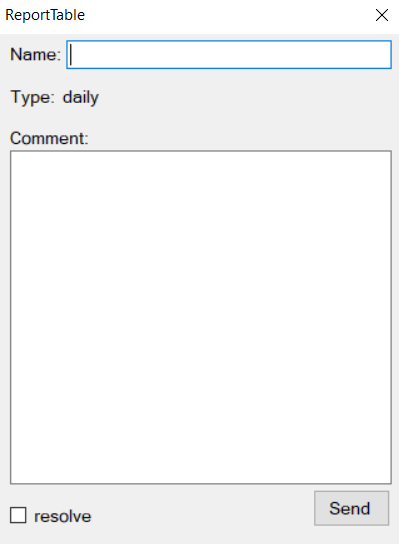
Есть два терминала – серверный, в виде консоли, и клиентский, в виде строки ввода и поля информации.

Серверный терминал имеет все права доступа ко всем файлам, может создавать персонал с любым уровнем доступа.

Уровень доступа имеет несколько уровней. 1 – самый высокий уровень допуска, 2 – чуть меньше и т. д., а 0 – общедоступный ресурс.

Для работы в клиентском приложении необходимо авторизироваться, введя имя (логин) и пароль. Далее мы видим окно приложения:

В нем мы можем искать/добавлять персонал, искать/добавлять/присваивать/завершать задачи. Команда запроса появится в поле над кнопкой Send. Также запрос можно ввести вручную. Для работы с отчетами открывается отдельное окно с возможностью редактирования тела отчета и его названия.



## Запросы

create staff name=… password=… <bossid=…> <securitylevel=… (server console only)> - создать нового сотрудника, дополнительно назначить ID начальника (не изменить в будущем)

create task target=… needsecuritylevel=… - создать новую задачу, цель изменить нельзя.

find task <id=…> <createdtime=… (date)> <lastchangetime=… (date)> <staffed=…> <laststaffid=…> <bybossid=…> - найти задачу с заданными параметрами

find staff id=… - найти сотрудника с указанным ID

update task comment=… (words must be splited by “^&”) – обновить комментарий к задаче

resolve task – завершить выполнение текущей задачи

set task id=… staffid=… - присвоить задачу сотруднику.

delete staff id=… (server console command) – удалить сотрудника из системы.