

Министерство образования и науки Российской Федерации
Национальный исследовательский технологический университет «МИСиС»
Институт информационных технологий и автоматизированных систем
управления Кафедра Инженерной Кибернетики

Курсовая работа
по дисциплине «Технологии программирования» на тему
«Приложения для ведения базы данных фитнес клуба»

Выполнил:
студент группы БПМ-19-1
Мисютин В.А.

Проверил:
доцент кафедры ИК, к.т.н.
Полевой Д.В.

Москва, 2020

Оглавление

ЗАДАЧА.....	3
ТЕХНИЧЕСКОЕ ЗАДАНИЕ	3
ДОПОЛНИТЕЛЬНЫЕ ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ	4
ПОЛЬЗОВАТЕЛЬСКАЯ ДОКУМЕНТАЦИЯ	4
ТЕХНИЧЕСКОЕ ОПИСАНИЕ.....	11
РУКОВОДСТВО ПО СБОРКЕ	13
ТЕСТОВЫЕ ДАННЫЕ	14

ЗАДАЧА

Написать пользовательское приложение для ведения базы данных фитнес клуба.

Программа должна иметь следующий функционал: ведение (добавление / удаление) услуг, предоставляемых клубом, ведение (добавление / редактирование / удаление) возможных тарифов для клиентов и ведение (добавление / редактирование / удаление) клиентской базы. Основная цель данной программы – обеспечить графически простое ведение базы данных фитнес клуба (тарифы / услуги / клиенты) и снизить возможность случайного изменения данных внутри базы (случайное удаление / случайное редактирование пользователя).

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Реализовать приложение на языке C++14. В стартовом окне приложения должны находиться 2 вкладки:

1) Вкладка «Клиенты» содержит все возможные действия над клиентской базой:

Таблица с полным списком клиентов, поисковая строка по таблице с клиентами, блок с полной информацией про выбранного клиента, кнопки для добавления / удаления клиентов, кнопка для обновления таблицы с клиентами (для случаев, когда в базу данных вносятся изменения из других программ в режиме реального времени), кнопки продления абонемента (если это возможно для действующего тарифа у выбранного клиента), смены тарифа у выбранного клиента, редактирования информации о клиенте (по умолчанию все графы закрыты от вписывания, чтобы снизить вероятность случайного искажения данных).

2) Вкладка «Тарифы» содержит все возможные действия над тарифами и услугами:

Таблица с полным списком тарифов, кнопки для добавления / редактирования / удаления тарифа, , кнопка для обновления таблицы с тарифами (для случаев, когда в базу данных вносятся изменения из других программ в режиме реального времени), блок с полной информацией про выбранный тариф, кнопка для вызова окна с услугами, которая содержит таблицу услуг, соответствующие кнопки для добавления / удаления услуги и обновления таблицы.

Также, для обеспечения лояльности к ошибкам пользовательского использования, должны быть реализованы диалоговые окна, предупреждающие о некорректном использовании программы.

ДОПОЛНИТЕЛЬНЫЕ ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ

Для написания приложения используется графическая библиотека Qt и программная библиотека SQLite для поддержания базы данных.

ПОЛЬЗОВАТЕЛЬСКАЯ ДОКУМЕНТАЦИЯ

После запуска программы открывается окно с двумя вкладками: рисунок 1 – первая вкладка, рисунок 2 – вторая вкладка.

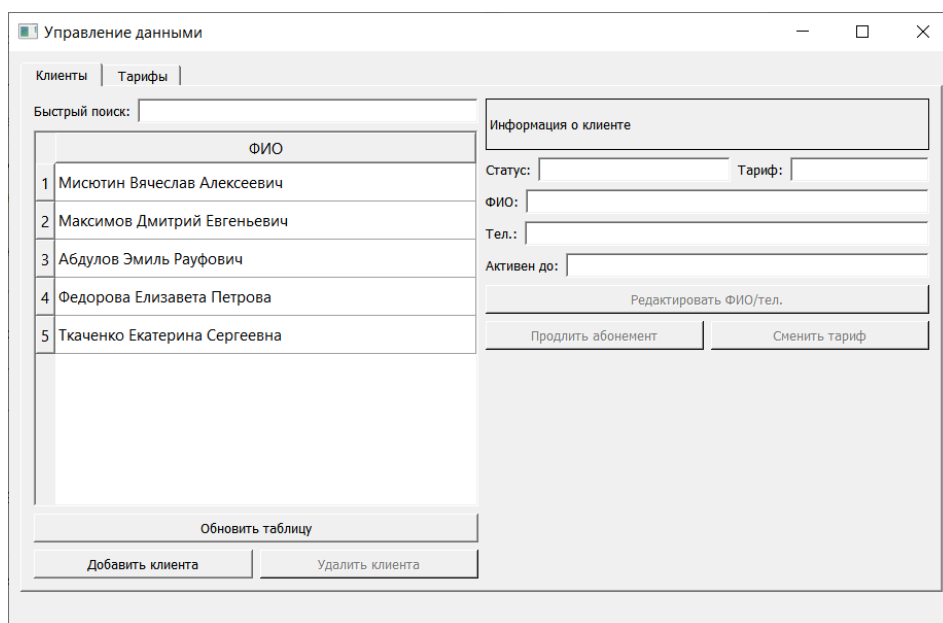


Рис.1 – Вкладка для работы с клиентами

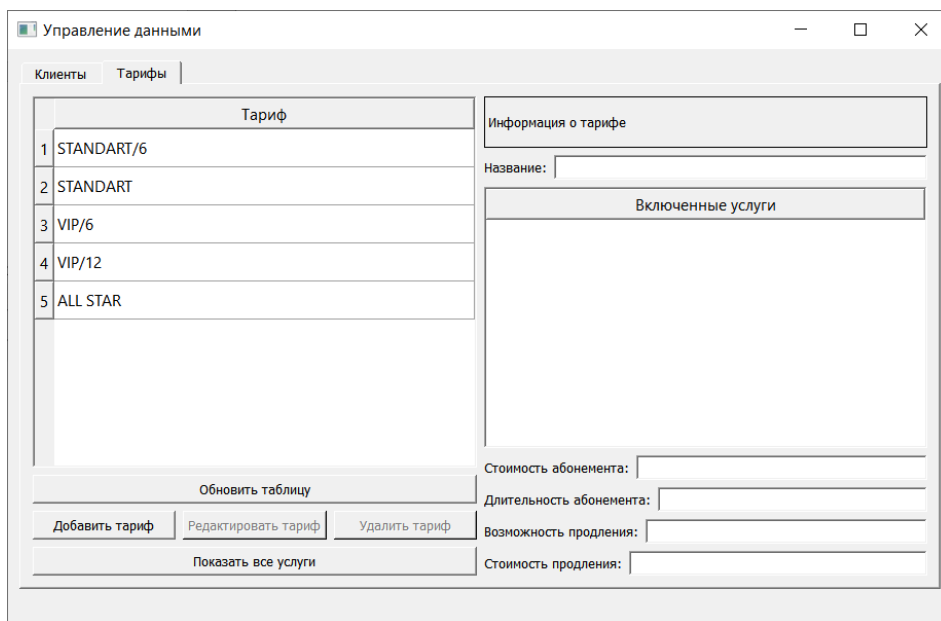


Рис.2 – Вкладка для работы с тарифами и услугами

Первая вкладка «Клиенты» предназначена для работы с клиентами:

- 1) При нажатии пкм. на определенного клиента в таблице (или при использовании кнопок Enter + стрелочки), в блоке «Информация о клиенте» появится вся информация о данном клиенте как это представлено на рисунке 3.

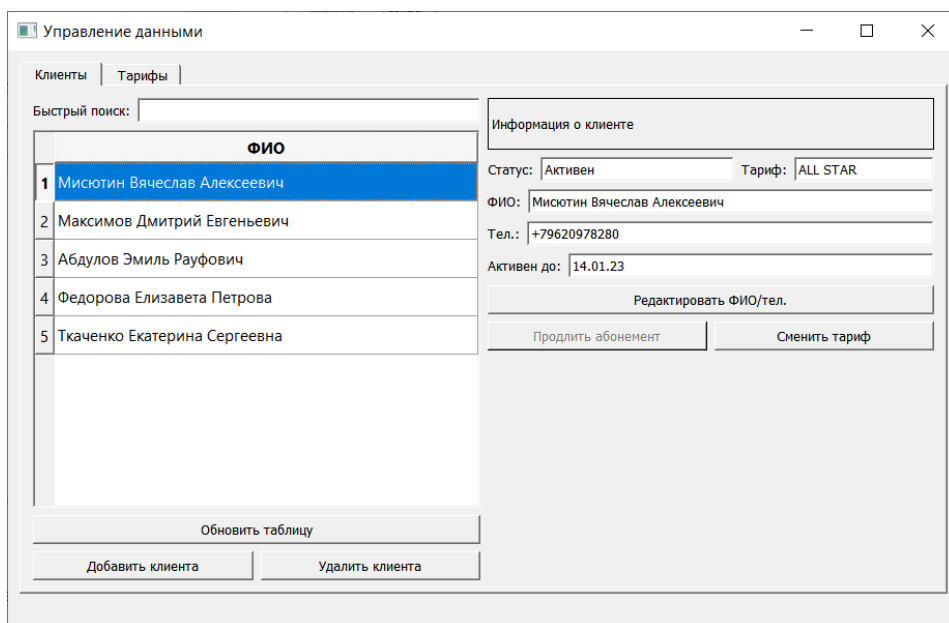


Рис.3 – Вывод информации в блок «Информация о клиенте»

- 2) При вводе текста в графу «Быстрый поиск» происходит поиск по всем клиентам как показано на рисунке 4, с каждым добавленным/удаленным символом будет происходить обновление списка, во время поиска все манипуляции с клиентами происходят также.

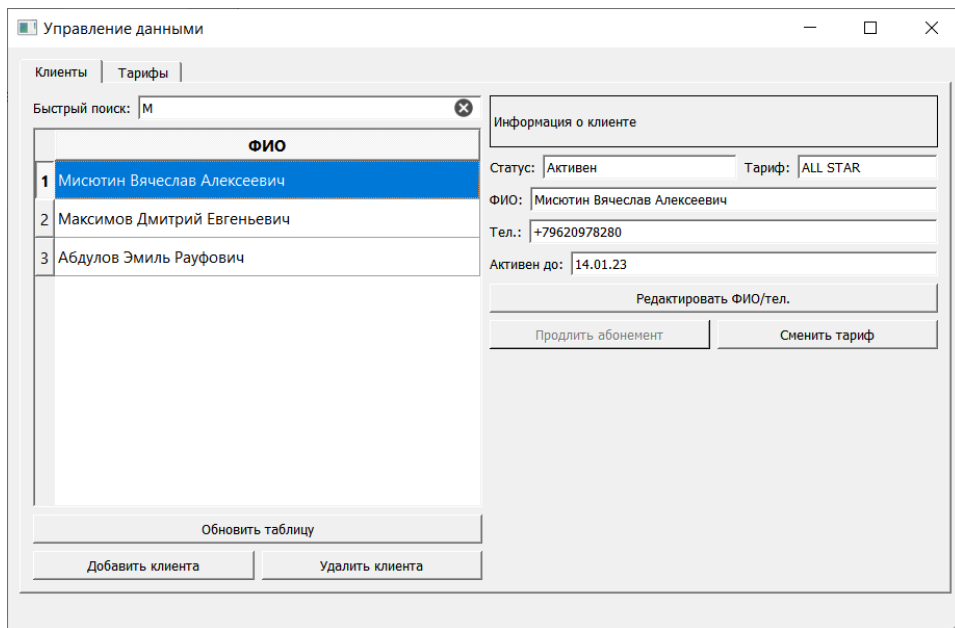


Рис.4 – Работа поисковой строки

- 3) При нажатии кнопки «Добавить клиента» откроется новое окно, которое предложит заполнить информацию о клиенте (текстовые поля для ФИО и телефона, и выпадающие списки для тарифа и кол-ва месяцев), причем выпадающий список в графе «Оплачено месяцев» предлагает только возможные для данного тарифа варианты как это показано на рисунке 5. При нажатии кнопки «Добавить» происходит добавление клиента в базу данных и последующий вывод данных о нем в блок «Информация о клиенте», при нажатии кнопки «Отмена» управление передается главному окну.

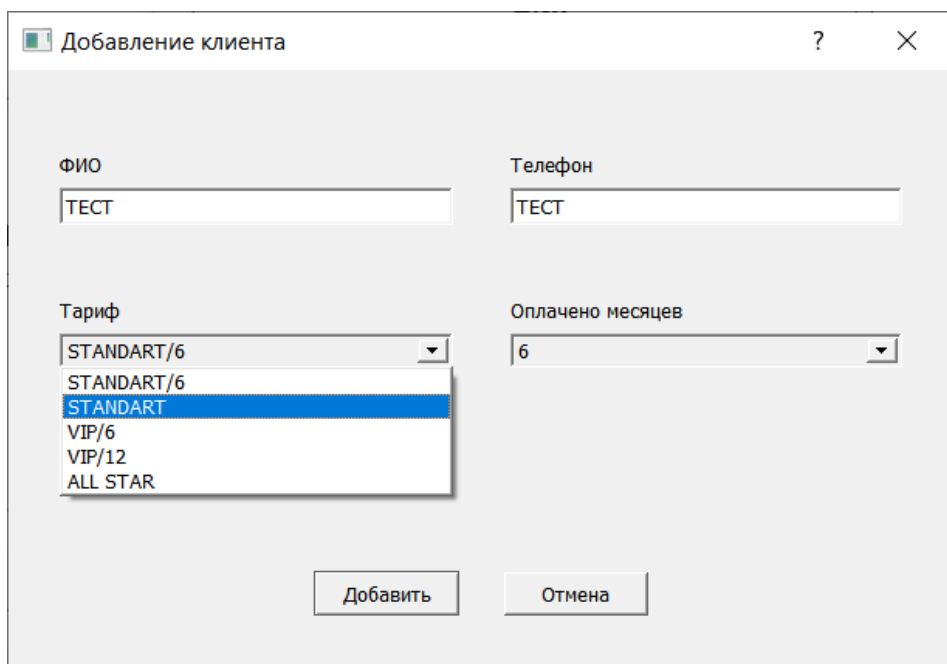


Рис.5 – Окно добавления клиента

Если хотя бы одно из текстовых окон пустое при нажатии кнопки «Добавить», то программа не разрешит вам добавить клиента.

- 4) При нажатии кнопки «Обновить таблицу» все виджеты установятся по умолчанию и произойдет полное считывание базы данных, если вдруг, она была изменена извне.
- 5) При нажатии кнопки «Редактировать ФИО/тел.» соответствующие поля откроются для редактирования, а кнопка изменится на «Сохранить изменения», после повторного нажатия на кнопку, изменения сохранятся.
- 6) При нажатии кнопки «Сменить тариф» появится выпадающий список для выбора тарифа и кнопки «Сохранить» и «Отмена» с соответствующими функциями как это показано на рисунке 6.

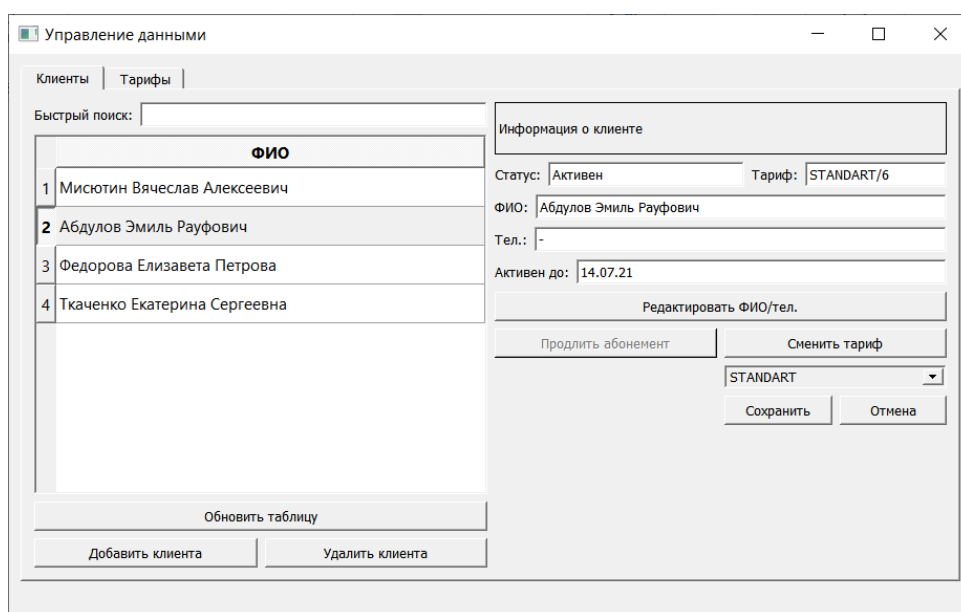


Рис.6 – Смена тарифа у клиента

- 7) При нажатии кнопки «Продлить абонемент» появится выпадающий список для выбора количества месяцев для продления и кнопки «Сохранить» и «Отмена» с соответствующими функциями как показано на рисунке 7.
- 8) Все кнопки из пунктов 6) – 8) по умолчанию не доступны, чтобы они стали активны, нужно выбрать конкретного клиента из списка. Также, если пользователь программы нажал на одну из этих кнопок и не завершил действие редактирования (не подтвердил сохранение соответствующей кнопкой или не нажал кнопку «Отмена»), то при любом следующем действии (не связанном с данным редактированием) сохранения изменений не произойдет, а кнопка примет начальное положение.

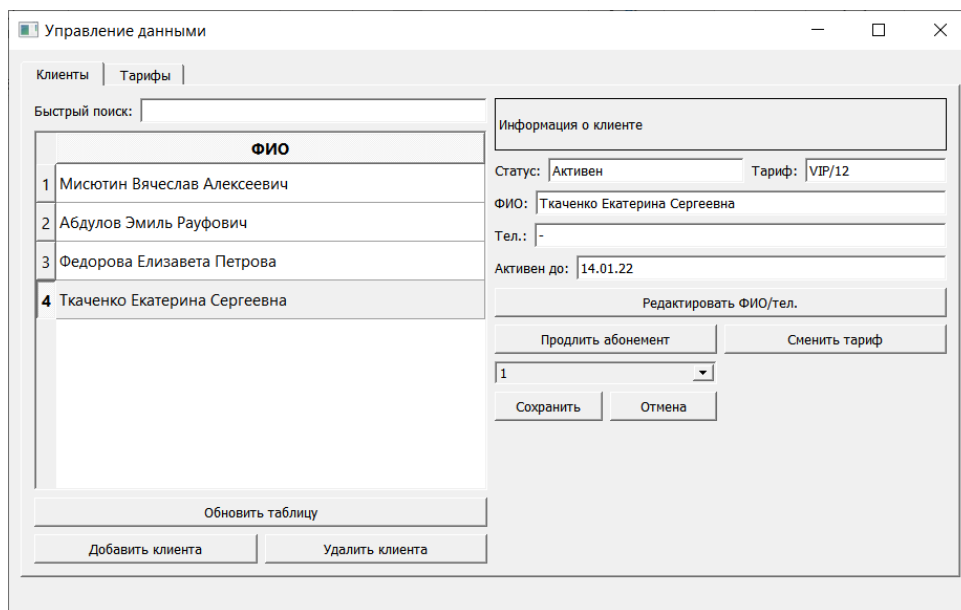


Рис.7 – Продление абонемента у клиента

Вторая вкладка «Тарифы» предназначена для работы с тарифами и услугами:

- 1) При нажатии пкм на определенного клиента в таблице (или при использовании кнопок Enter + стрелочки), в блоке «Информация о клиенте» появится вся информация о данном клиенте как это показано на рисунке 8.

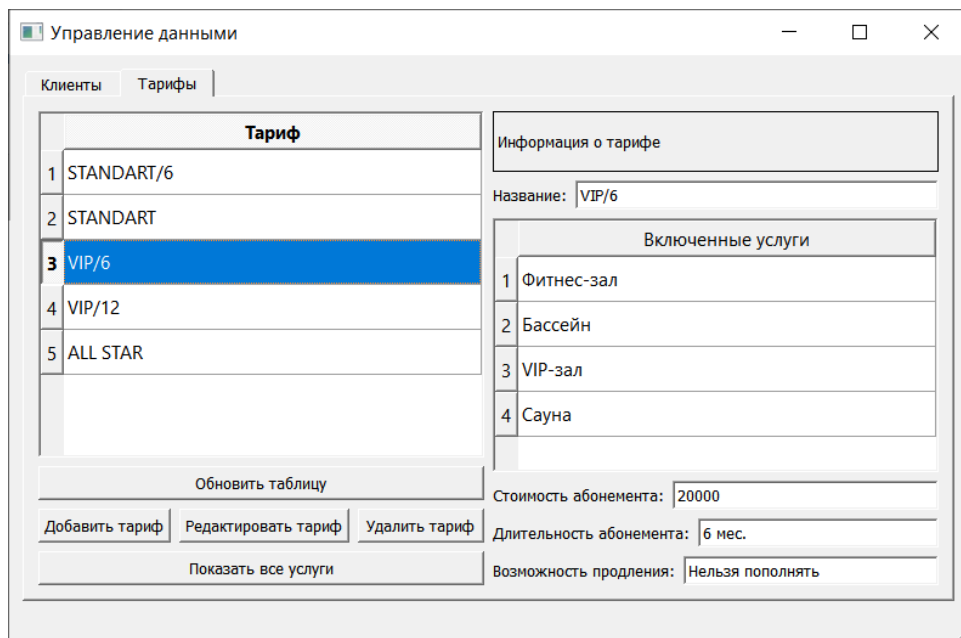


Рис.8 - Вывод информации в блок «Информация о тарифе»

- 2) При нажатии кнопки «Добавить тариф» откроется новое окно, представленное на рисунке 9, которое предложит заполнить информацию о тарифе (текстовые поля для названия, стоимости и стоимости продления, и выпадающие списки для длительности и возможности продления), причем если в графе длительности

выбрать «ежемесячно обновляемый», то поля с возможностью продления и стоимостью продления становятся недоступными, если выбрать любую другую длительность тарифа, то в зависимости от значения поля возможности продления будет зависеть доступность поля со стоимостью продления. Также в появившемся окне будет таблица с услугами, содержащая названия услуг и окно для выбора каждой услуги. Для добавления нового тарифа следует нажать кнопку «Добавить», для возврата к основному окну следует нажать кнопку «Отмена».

	Выбрать	
1	<input checked="" type="checkbox"/>	Фитнес-зал
2	<input type="checkbox"/>	Бассейн
3	<input checked="" type="checkbox"/>	Фитнес-бар
4	<input checked="" type="checkbox"/>	VIP-зал
5	<input type="checkbox"/>	Массаж

Рис.9 – Окно добавления тарифа

Если хотя бы одно доступное текстовое поле будет пустым, то программа выдаст ошибку.

- 3) При нажатии кнопки «Редактировать тариф» (если она доступна, т. е. выбран конкретный тариф в таблице) откроется окно, представленное на рисунке 10, полностью повторяющее окно добавления тарифа, единственная разница в том, что данное окно будет заполнено действующей информацией о тарифе из базы данных. Логика работы с окном остается такой же.

Изменение тарифа

Название тарифа: VIP/12 Длительность тарифа: 12 Стоимость тарифа: 35000

Возможность продления: Да Стоимость продления (за месяц): 4500

	Выбрать	Услуги
1	<input checked="" type="checkbox"/>	Фитнес-зал
2	<input checked="" type="checkbox"/>	Бассейн
3	<input type="checkbox"/>	Фитнес-бар
4	<input checked="" type="checkbox"/>	VIP-зал
5	<input type="checkbox"/>	Массаж
6	<input checked="" type="checkbox"/>	Сауна

Добавить Отмена

4) Рис.10 – Окно редактирования тарифа

- 5) При нажатии кнопки «Обновить таблицу» все виджеты установятся по умолчанию и произойдет считывание базы данных, если вдруг, она была изменена извне.
- 6) При нажатии кнопки «Показать все услуги» откроется окно, представленное на рисунке 11, которое предоставляет полный перечень действий над услугами подобно тем, что были для тарифов и клиентов («Добавить услугу», «Удалить услугу», «Обновить таблицу», «Вернуться к тарифам» (возврат к основному окну)). При нажатии кнопки «Добавить услугу» появится окно, изображенное на рисунке 12. При попытке добавить существующую услугу программа выдаст ошибку.

Услуги

	Название услуги
1	Фитнес-зал
2	Бассейн
3	Фитнес-бар
4	VIP-зал
- ...	

Обновить таблицу

Добавить услугу

Удалить услугу

Вернуться к тарифам

Рис.11 – Окно для работы с услугами

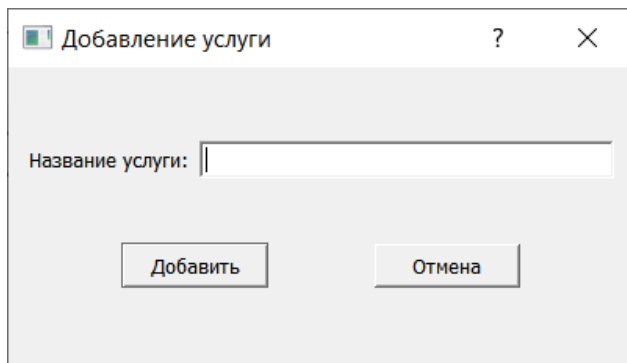


Рис.12 – Окно добавления услуг

Также при любой попытке удаления тарифа/клиента программа всегда уточняет, действительно ли вы хотите совершить удаление (защита от случайного удаления).

ТЕХНИЧЕСКОЕ ОПИСАНИЕ

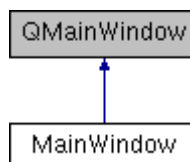
Основной функционал содержится в главном окне данной программы:

Класс MainWindow

Класс описывает логику работы основного окна

#include <[mainwindow.h](#)>

Граф наследования: MainWindow:



Сигналы

void [sentId](#) (QString id)

Сигнал для передачи id о выбранном тарифе в окно класса "EditWindow".

Открытые члены

void [parse_db](#) ()

Метод для заполнения таблицы именами клиентов.

void	<u>clear_info</u> ()
	Метод для очистки блока "Информация о тарифе".

void	<u>parse_info</u> (int row)
	Метод для заполнения блока "Информация о тарифе".

void	<u>parse_tariff</u> ()
	Метод для вывода всех тарифов в таблицу.

void	<u>parse_tariff</u> (int row)
	Метод для заполнения блока "Информация о тарифе".

void	<u>clear_info_tariff</u> ()
	Метод для очистки блока "Информация о тарифе".

void	<u>FIO_phone_readonly</u> (bool pos)
	Метод для контроля возможности редактирования полей "тел." и "ФИО".

void	<u>search</u> (const QString &arg1)
	Метод для поиска по подстроке.

Главное окно транслирует текущее состояние базы данных (тарифы, клиенты) и также главное окно содержит основные функции для манипуляции этими данными. Из главного окна вызываются модальные окна для решения локальных задач (добавление/редактирование отдельного объекта).

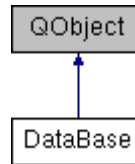
Также в программе существует отдельный класс для работы с базой данных, который в первую очередь нужен для подключения или создания по заданному примеру базы данных:

Класс DataBase

Класс описывает логику работы с базой данных

```
#include <database.h>
```

Граф наследования: DataBase:



Открытые члены

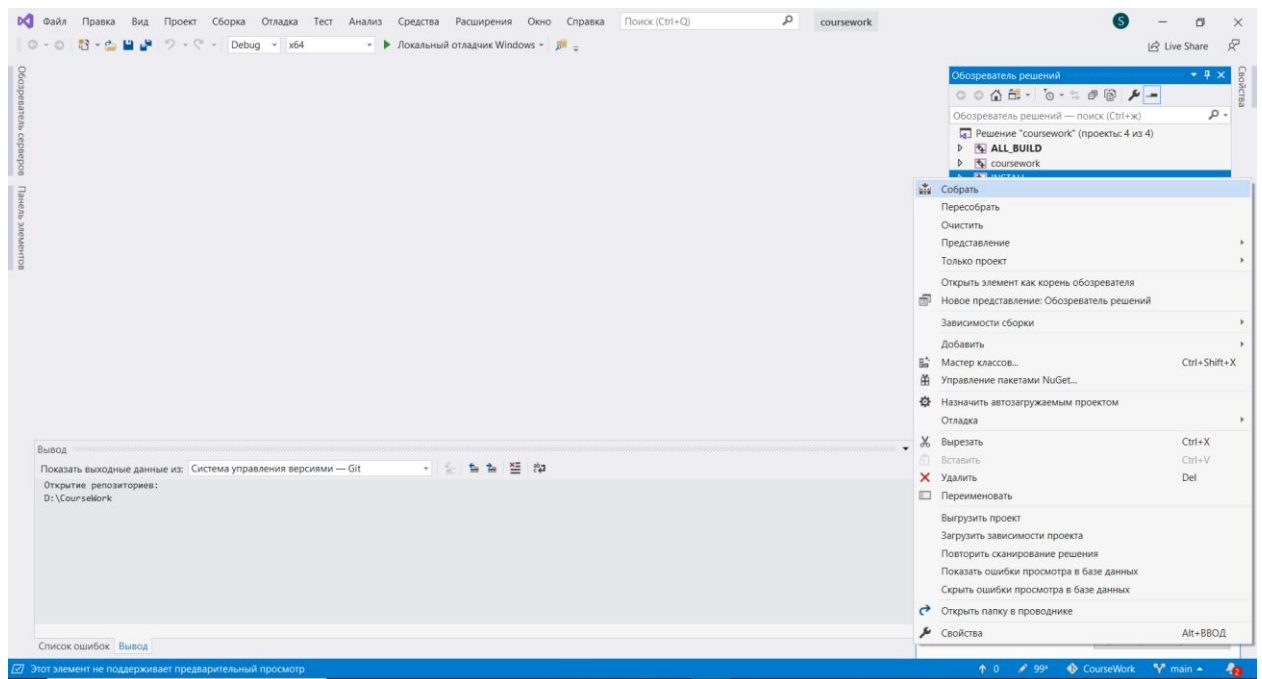
void	connectToDataBase ()
	Метод для подключения базы данных.

void	closeDataBase ()
	Метод для прекращения работы с базой данных.

Полная документация, созданная с помощью Doxygen доступна в папке *docs* репозитория. Чтобы посмотреть её в html-формате нужно запустить *docs/html/index.html*.

РУКОВОДСТВО ПО СБОРКЕ

1. Загрузите исходники из репозитория https://mysvn.ru/clavka40rus1/misutin_v_a/
2. Собрать проект через CMake по папке *gym/asd/* (при необходимости можно указать CMAKE_INSTALL_PATH). Если при сборке не удастся найти директорию Qt (не переменная среда), то в CMakeLists.txt необходимо указать путь до папки с библиотеками.
3. Через IDE (Visual Studio) собрать INSTALL часть проекта.



4. Теперь можно переходить в директорию установки и пользоваться приложением (coursework.exe).

ТЕСТОВЫЕ ДАННЫЕ

Тестовые данные (частично заполненная база данных) доступна в папке *default db*.

Чтобы запустить программу не с пустой базой данных, а с готовой (тестовой) нужно перетащить файл *default db/coursework.db* в папку инсталляции к исполняемому файлу (с заменой если нужно).