Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение  
высшего образования   
«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет информационных технологий

*наименование подразделения*

Кафедра прикладной математики

*наименование кафедры*

Отчет защищен с оценкой\_\_\_\_\_\_\_\_\_

“\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020г.

Руководитель от вуза

\_\_\_\_\_\_\_\_\_\_\_\_/Троицкий В. С./

подпись Ф. И. О.

**Отчет**

по курсовой работе

Разработка программы-планировщика «Сделай все»

*общая формулировка задания*

в АлтГТУ

*наименование организации*

Студент гр. ПИ-82\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Жукова А.В.

*индекс группы* *подпись Ф. И. О.*

Студент гр. ПИ-82\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Верещагина И.А.

*индекс группы* *подпись Ф. И. О.*

Студент гр. ПИ-82\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Егорова В.Е.

*индекс группы* *подпись Ф. И. О.*

Руководитель от организации \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Троицкий В. С.

*индекс группы* *подпись Ф. И. О.*

Руководитель от университета \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Троицкий В. С.

*индекс группы* *подпись Ф. И. О.*

2020

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Учебная дисциплина: «Объектно-ориентированное программирование»  Задание: Разработать ПО и оформить ПЗ  Тема: Разработка программы-планировщика «Сделай все»  Разделы работы и сроки их выполнения:   1. Обзор предметной области и постановка задачи. Результатом работы является письменный отчет, содержимое которого будет первой главой в пояснительной записке (23.09.2019- 20.10.2019); 2. Проектирование. Определение обобщенной структуры программы, описание выделенных блоков и связей между ними (чертеж структурной схемы либо схемы работы системы). Описание функционирования разрабатываемой программы, ее структуры с точки зрения описания данных и кода для их обработки (описывается структура и взаимоотношения между классами со ссылками на диаграмму классов). При использовании БД, описать структуру таблиц и связи между ними со ссылками на модель данных. Проектирование пользовательского интерфейса. Если программные средства разработки не заданы, то производится их выбор. Результатом работы является письменный отчет, содержимое которого будет второй и третьей главой в пояснительной записке (21.10.2019-20.11.2019); 3. Разработка ПО, тестирование и отладка. Результатом работы является ПО и письменный отчет (текст программы, комментарии к нему и сопроводительная документация), содержимое отчета станет частью третей главы и приложением номер 1 к пояснительной записке (21.11.2019-20.12.2019); 4. Оформление отчета о проделанной работе (21.12.2019-24.12.2019); 5. Сдача работы руководителю и ее защита (25.12.2019-29.12.2019).   Дата выдачи здания: 23.09.2019  Руководитель работы: доцент кафедры ПМ \_\_\_\_\_\_\_\_В.С.Троицкий | | | | | | | | | | | |
|  |  |  |  |  | **КП 09.03.04.33.000 ПЗ** | | | | | | |
|  |  |  |  |  |
| Изм. | Лист | № Докум. | Подпись | Дата |
| Разраб. | | Жукова А.В.  Верещагина И.А.  Егорова В.Е. |  |  | Разработка программы-планировщика «Сделай все» | Лит. | | | Лист | | Листов |
| Пров. | | Троицкий В.С. |  |  |  | У |  | | 2 |  |
|  | |  |  |  | АлтГТУ, ФИТ, гр. ПИ-82 | | | | | |
| Н. Контроллер | | Троицкий В.С |  |  |
| Утв. | | Кантор С.А. |  |  |

**Оглавление**

[**Введение** 4](#_Toc30737998)

[**Глава I. Обзор предметной области и постановка задачи** 5](#_Toc30737999)

[**1.1. Обзор предметной области** 5](#_Toc30738000)

[**1.2. Постановка задачи.** 8](#_Toc30738001)

[**Глава II. Проектирование** 10](#_Toc30738002)

[**Глава III. Реализация** 15](#_Toc30738003)

[**3.1. Общий подход к реализации:** 15](#_Toc30738004)

[**3.2. Модульная структура.** 15](#_Toc30738005)

[**3.3. Реализация основного функционала** 16](#_Toc30738006)

[**3.4. Распределение работы.** 18](#_Toc30738007)

[**Заключение** 19](#_Toc30738008)

[**Список использованной литературы** 20](#_Toc30738009)

[**Приложение А. Снимки пользовательского интерфейса** 21](#_Toc30738010)

[**Приложение Б. Исходный код программы** 31](#_Toc30738011)

[**Приложение В. Руководство системного администратора** 32](#_Toc30738012)

[**Приложение Г. Руководство пользователя** 33](#_Toc30738013)

# **Введение**

Каждому из нас приходилось планировать распорядок дня, недели или другого временного отрезка, и каждый знает, если держать список дел в голове, то можно попросту забыть обо всем и запутаться в собственных планах.  Ручное ведение записей не дает эффективного результата, так как ручная обработка затрачивает большее время, чем машинная, причем с увеличением объема, уровень затрат увеличивается. Поэтому разработка программы-планировщика в настоящее время актуальна, так как данный тип приложений позволяет хранить различную информацию, осуществлять навигацию по ней и удобное управление.

**При современном темпе** жизни программа-планировщик может пригодиться любому человеку, независимо от его возраста и профессии.

**Цель работы:** изучение принципов разработки графических

приложений, получение навыков разработки приложений с графическим

интерфейсом в средах разработки Microsoft Visual Studio и Qt Creator.

**Главная задача:** написание программного продукта в двух версиях, обладающего удобным интерфейсом и реализующего хранения и обработку данных для нескольких пользователей.

# **Глава I. Обзор предметной области и постановка задачи**

## **1.1. Обзор предметной области**

**Планировщик дел** – программа, относящаяся к прикладному программному обеспечению, предназначена для накопления пользовательской информации, поиска по ней, организации дел, событий и контроля за их выполнением. Планировщики позволяют эффективно распоряжаться рабочим временем. Обладают возможностью автоматизации регулярных действий, составления персональных и групповых расписаний, планирования встреч и т.д.

По сути, эта программа заменила бумажные ежедневники, добавив дополнительный функционал (сортировка списка дел, распределение по категориям, присваивание меток, настройка напоминания).

Существуют планировщики для компьютеров, мобильных устройств и их онлайн-аналоги.

Возможно одновременное использование планировщика и на телефоне, и на компьютере (т.е. синхронизация информации).

Планировщик может работать как в онлайн, так и в офлайн режиме.

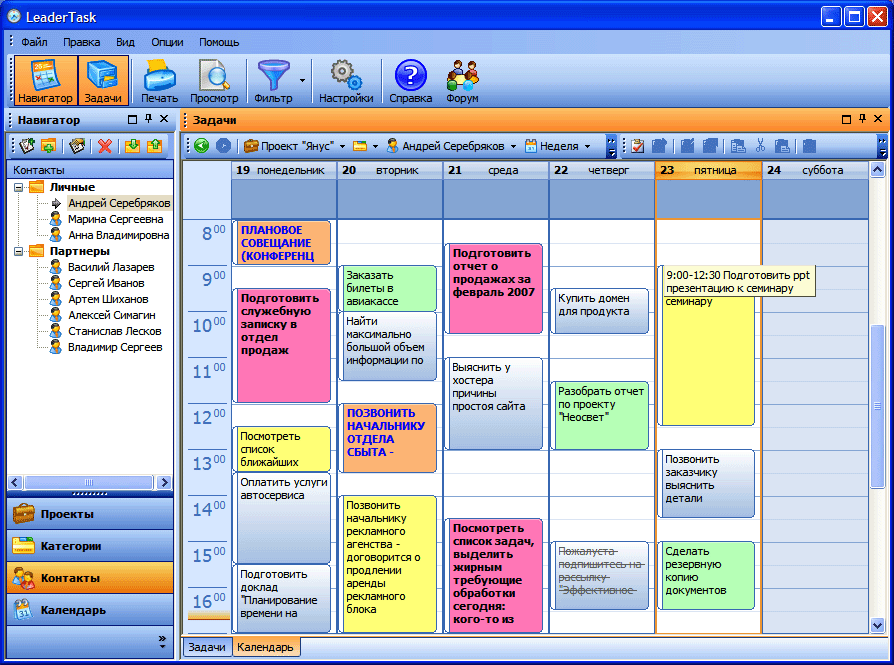


Рис.1 Планировщик Leader Task.

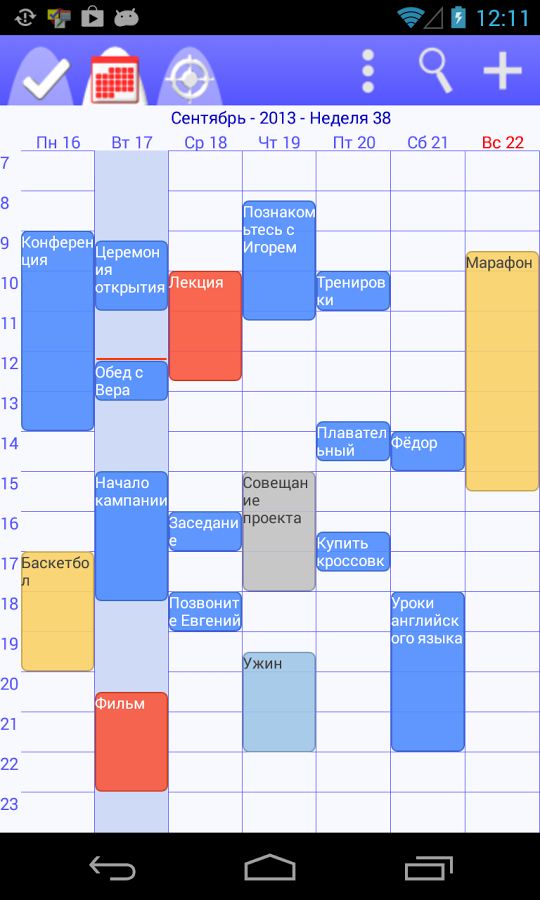


Рис.2 Планировщик для мобильного телефона

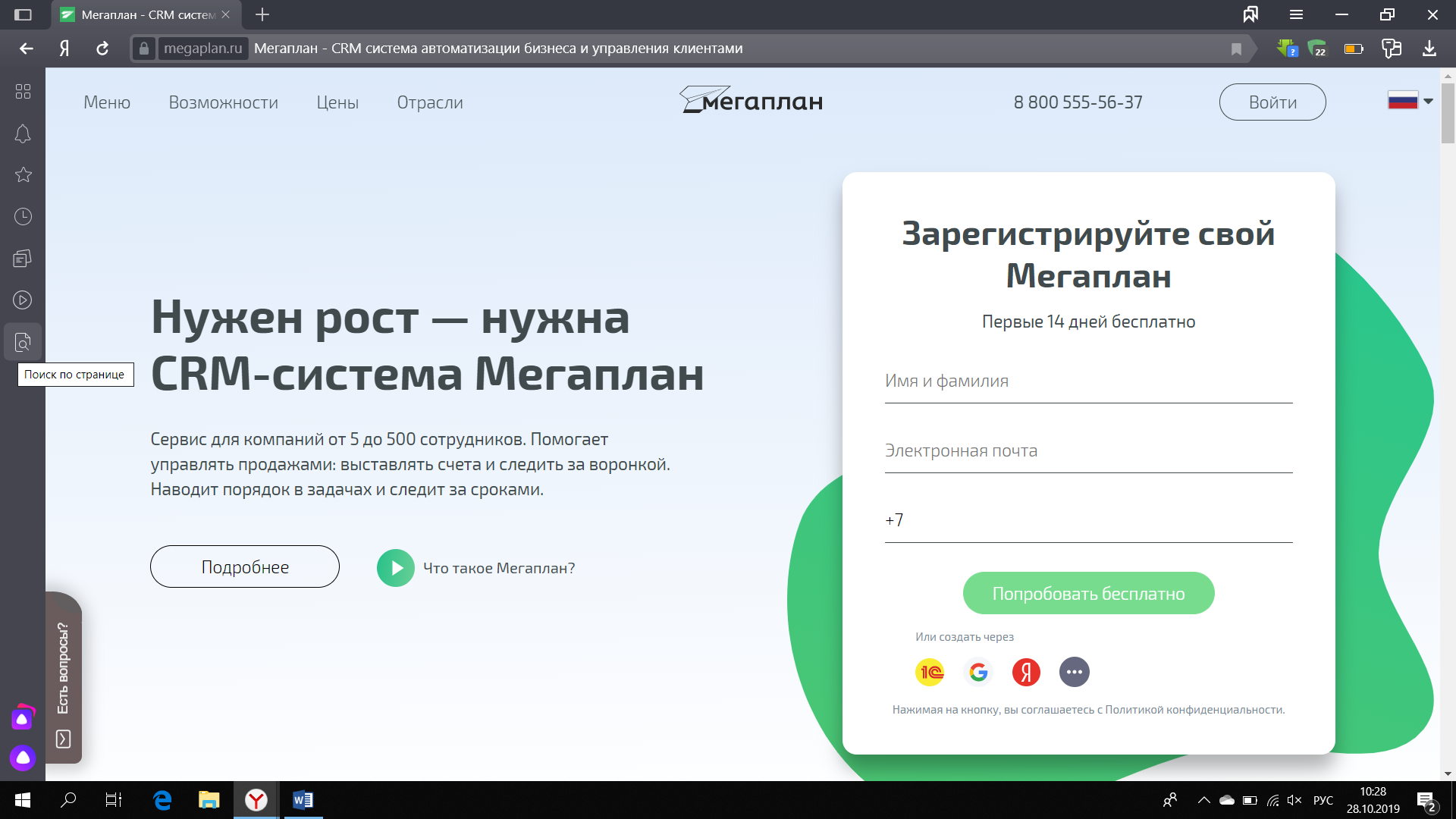


Рис.3 Онлайн – планировщик

**Критерии оценивания конкретного планировщика:**

1. **Удобство интерфейса**: учитываются такие свойства программы, как скорость доступа к нужным функциям, наличие горячих клавиш, интуитивный интерфейс и т.д.
2. **Задачи и календарь**: создание задач, списков, категорий, подзадач, возможности календаря при планировании мероприятий (создании событий).
3. **Уведомления**: настройка уведомлений о событиях, создание липких заметок, способы уведомления.
4. **Совместный доступ**: возможность создания нескольких учетных записей для нескольких пользователей. Из этого вытекает следующий критерий…
5. **Безопасность**: установка пароля на учетную запись.
6. **Мобильность и синхронизация**: доступность для мобильных платформ, интеграция с онлайн-сервисами.
7. **Другие инструменты**: инструменты, дополняющие функциональность планировщика (редактор заметок, менеджер паролей и другое).

## **1.2. Постановка задачи.**

Программа выполняет следующие функции:

1. Синхронизация с календарём;
2. Привязка события или мероприятия к дате;
3. Хранение задач и осуществление контроля за их выполнением;
4. Оповещения;
5. Создание отчета выполнения задач за определенный период;
6. Обеспечение работы нескольких пользователей;

Программа хранит данные о пользователе и список запланированных задач.Взаимодействие с программой осуществляется через учетную запись.

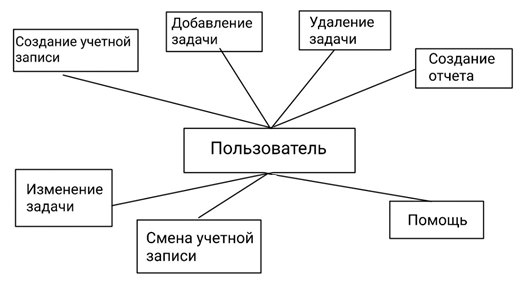


Рис. 4 Схема взаимодействия

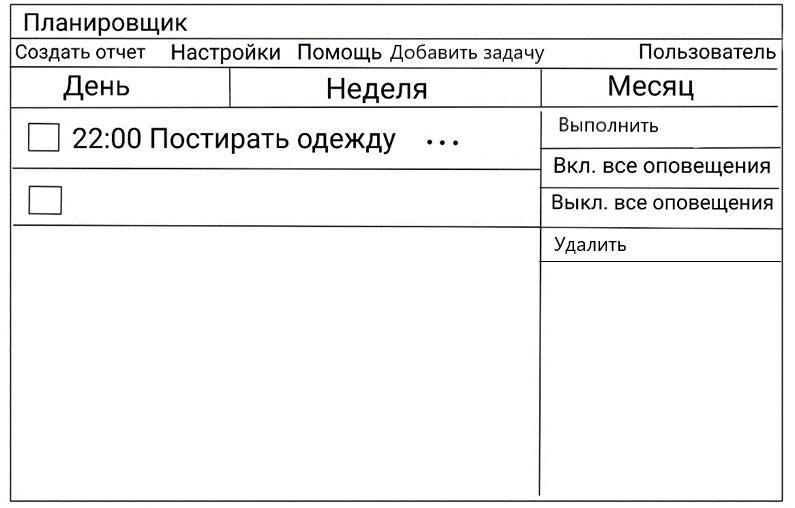


Рис. 5 Макет главного окна

Основной функционал программы:  
1) Создание учётной записи(выбор учётной записи)

Работа планировщика начинается с создания учётной записи (если она не создана) или входа в учётную запись (если она создана).При создании новой записи необходимо ввести придуманный вами имя, логин и пароль. Автоматически создаётся файл для пользователя, который в дальнейшем будет использоваться для занесения задач и данных о пользователе.

2) Добавление задачи

Планировщик имеет такую функцию как добавление задачи. Для того, чтобы добавить задачу, необходимо нажать кнопку добавления. Появится новое окно, в котором заполняются все поля (время, описание задачи, тип оповещения). Новая задача записывается в файл пользователя, учетная запись которого открыта.

3)Удаление записи

Планировщик имеет такую функцию, как «Удаление записи». Для того чтобы удалить созданную вами запись, выберите в меню любую из предложенных вкладок: день, неделю или месяц.

Затем поставить галочку в чекбоксах удаляемых записей и нажать кнопку «Удалить»

4) Изменение записи

Планировщик имеет такую функцию, как «Изменение записи». Для того чтобы изменить созданную Вами запись, выберите в меню любую из предложенных вкладок: день, неделя или месяц.

Отметить изменяемую запись (в чекбоксе) и в новом окне с подробной информацией о данной записи исправить данные.

5) Помощь

В ленточном меню планировщика располагается вкладка «Помощь», которая подсказывает пользователю как работать с программой.

# **Глава II. Проектирование**

Запуск приложения начинается с приветственного окна (объект panel). Проверяется файл data.dat (описание файла ниже) с информацией об учетных записях, если файл пуст – считается, что запуск первый. Если запуск первый, то пользователь переходит к следующему шагу, попадая в окно регистрации (после регистрации переходит на панель входа).

В окне входа предлагаются либо «Вход», либо «Регистрация» (переход к окну регистрации). При входе сверяются введенные логин и пароль с данными из документа data.dat, вывод ошибки при несоответствии, при успешном вводе переход в главное окно.

При регистрации проверяются данные файла data.dat, при возникновении ошибок(ввод недопустимых символов, недостаточное количество символов, ввод пустой строки) выводится ошибка. При успешной регистрации новые данные заносятся в файл data.dat и осуществляется переход к окну входа.

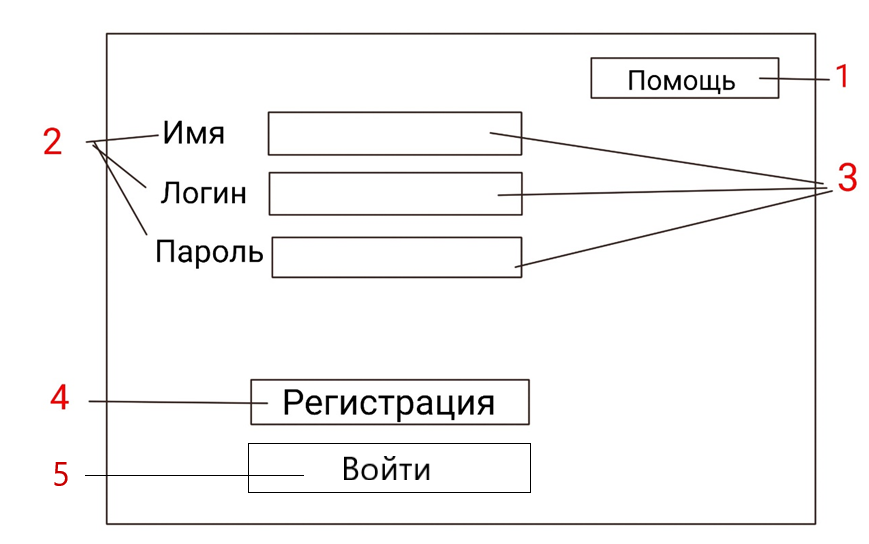


Рис. 6. Макет панели регистрации  
1. кнопка, открывающая инструкцию работы в данном разделе(объект button)  
2. названия полей ввода(объекты label)  
3. поля ввода(объекты textBox)  
4. кнопка регистрации, запускающая проверку введенных данных, при успешном вводе осуществляется переход к окну входа(объект button)  
5. Переход к панели входа (объект button)

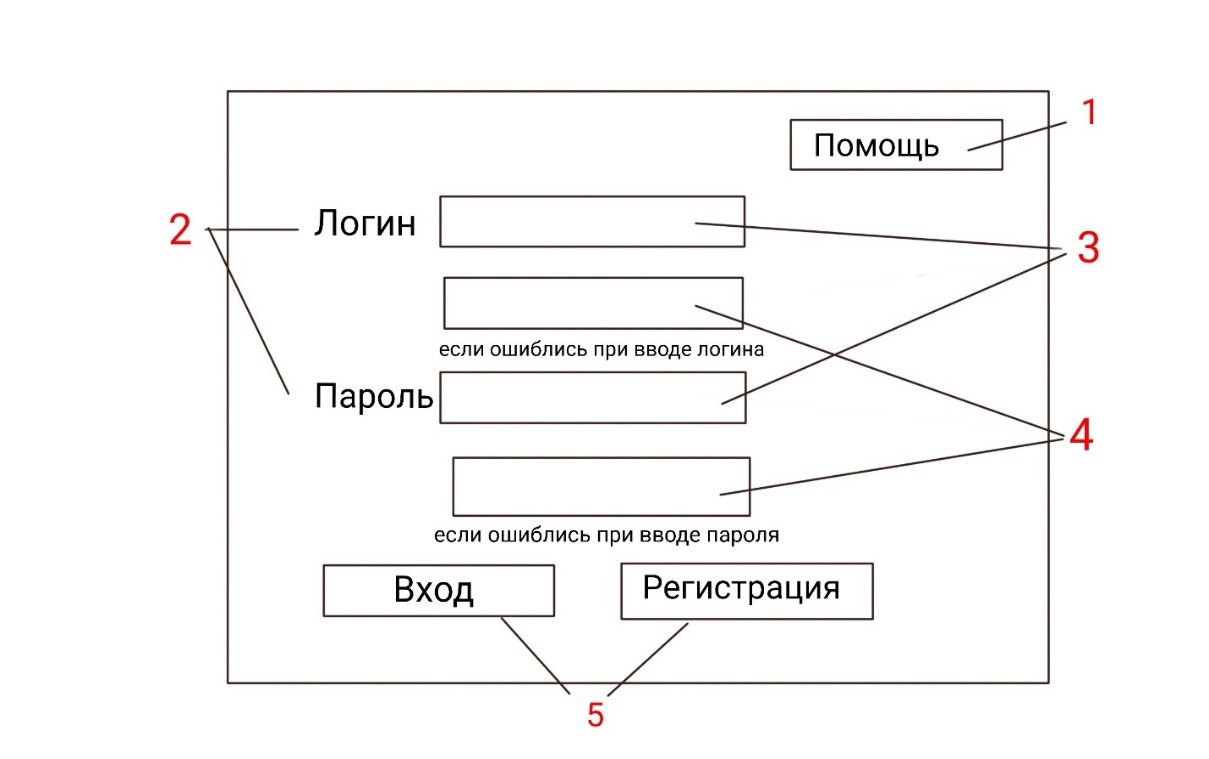


Рис.7. Макет панели входа   
1. кнопка, открывающая инструкцию работы в данном разделе (объект button)  
2. названия полей ввода (объектs label)  
3. поля ввода (объектs TextBox)  
4. сообщения об ошибке, которые выводятся после проверки введенных данных в случае ошибки(объектs label)  
5. кнопки входа (проверка введенных данных и переход к учетной записи) и кнопка перехода к окну регистрации (объектs button)

**Используемые файлы:**

1. Файл data.dat
2. Файлы «Логин пользователя».dat

В файле data.dat хранится список существующих учетных записей. Файл имеет структуру:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Индекс активности | Логин | Пароль | Имя пользователя | Дата создания |

Индекс – флаг, который принимает значение 1 или 0, информирует о том, какая учетная запись активна (если пользователь закрыл программу, забыв выйти с учетной записи, то при ее запуске программа определит по индексу, какая учетная запись была активна и откроет ее);

Логин – это название файла пользователя.

Логин и пароль нужны для входа в учетную запись пользователя.

Файл data.dat используется:

1. При запуске программы (проверяется, существуют ли учетные записи)
2. При входе в учетную запись (Проверка на ввод логина и пароля)
3. При регистрации (заносится новая запись, а также проверяется, не существует ли уже такой же логин или пароль)
4. При удалении учетной записи (удаляются данные)
5. При изменении данных о пользователе (меняются данные)
6. При выходе из учетной записи (меняется индекс активности)

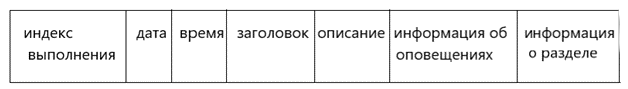
Файлы «Логин пользователя».dat, количество файлов соответствует количеству записей в файле data.dat. Очередной файл «Логин пользователя».dat создается при регистрации пользователя.

В файле хранятся все планы пользователя.

Структура

3.2. Модульная структура.

В программе реализованы следующие модули и взаимодействие между ними



Индекс выполнения указывает, выполнил ли пользователь запланированное.

Дата и время (если есть) события.

Заголовок действие – короткое название плана.

Описание – подробности запланированного.

Информация об оповещении – хранится информация о том, как оповещать пользователь.

Информация о разделе, т.е. задача на день, неделю или месяц.

Файл используется:

1. Вход в учетную запись, для вывода данных на экран
2. При добавлении, удалении, изменении действия или изменении его оповещения
3. При оповещении, для вывода информации о действии

**Описание инструментов:**

Добавление задачи.

1. поля для отметки вида оповещения (объекты checkBox)

2. Время и дата оповещения

3.Заголовок и описание (объекты TextBox)

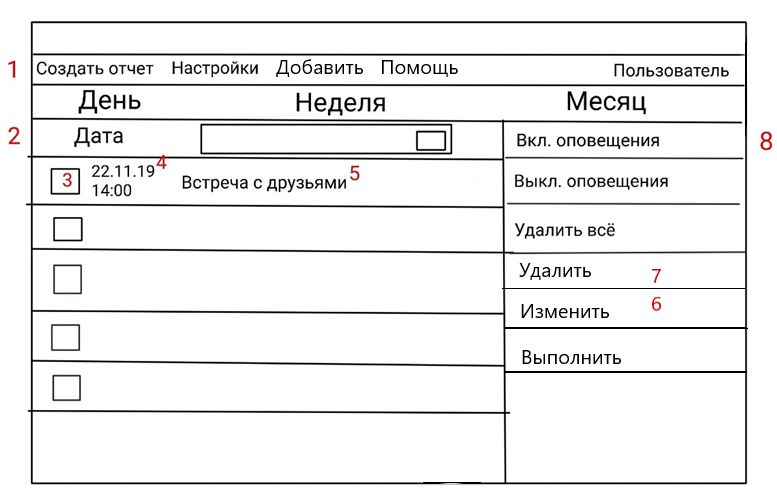
Главная форма:  


Рис 9. Окно дня.

Ленточное меню (элемент MenuStrip): глобальный функционал и переход к нужному разделу  
2. поле для выбора необходимого дня(объект dateTimePicker)  
3. кнопка выделения записи  
4. дата оповещения о действии  
5. заголовок действия  
6. кнопка перехода в редактор данной записи  
7. кнопка удаления записи  
8. кнопка включения оповещений для всех записей данного дня, кнопка отключения оповещений для всех записей данного дня, кнопка удаления всех записей данного дня

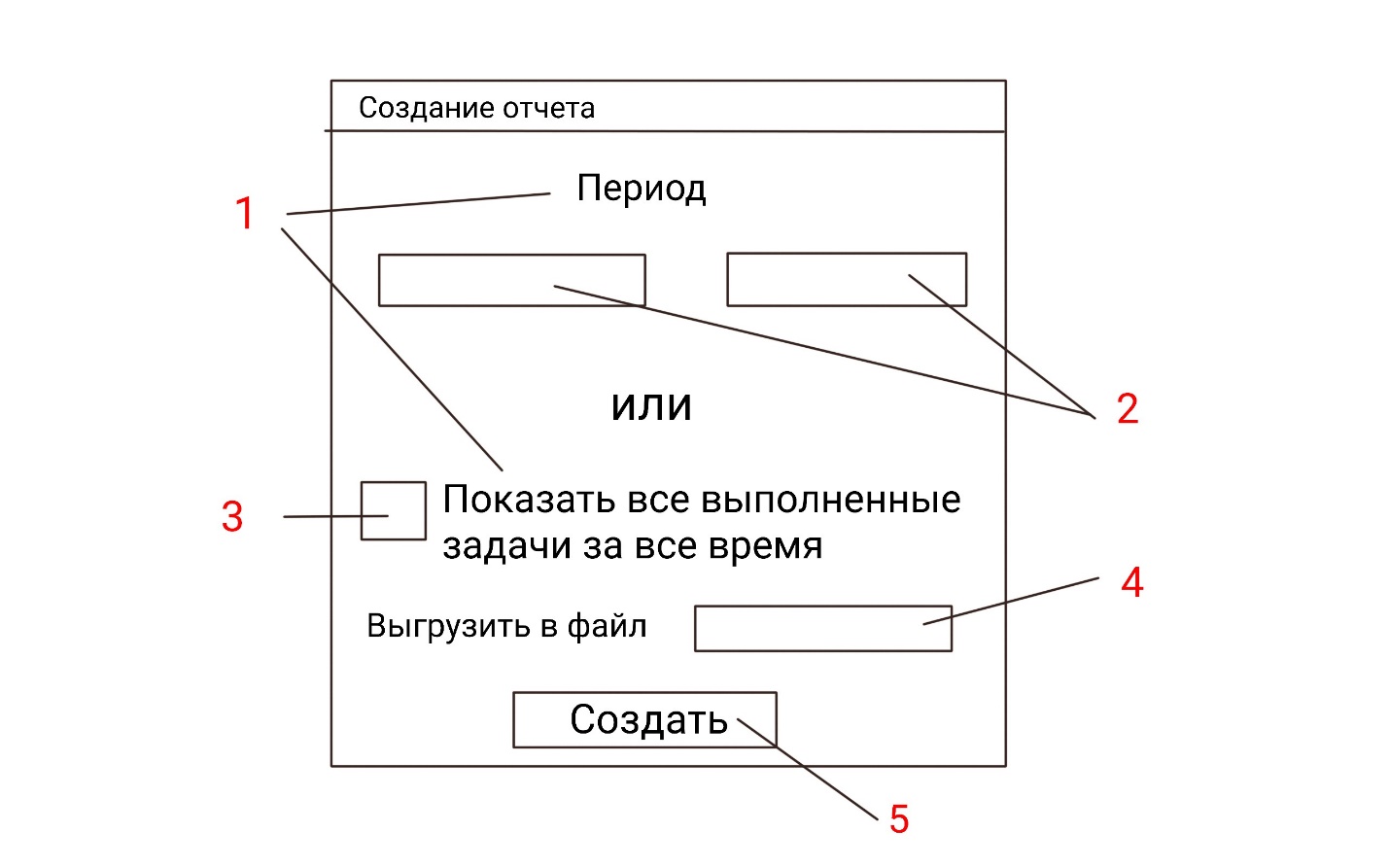
Окно недели будет отличаться дополнительной таблицей, в которой глобальные задачи на конкретный день без времени, окно месяца дополняет таблица задач на месяц без конкретной даты.

Рис 10. Окно создания отчета

1. названия полей  
2. поля ввода периода времени, за который необходимо создать отчет  
3. кнопка выбора, которая выбирает весь период (с даты регистрации до сегодняшней даты)  
4. имя файла  
5. кнопка создания

# **Глава III. Реализация**

## **3.1. Общий подход к реализации:**

Курсовой проект реализован на двух языках:

1. На языке C# в среде разработки Microsoft Visual Studio 2019;
2. На языке C++ в среде разработки Qt Creator.

## **3.2. Модульная структура.**

В программе реализованы следующие модули и взаимодействие между ними

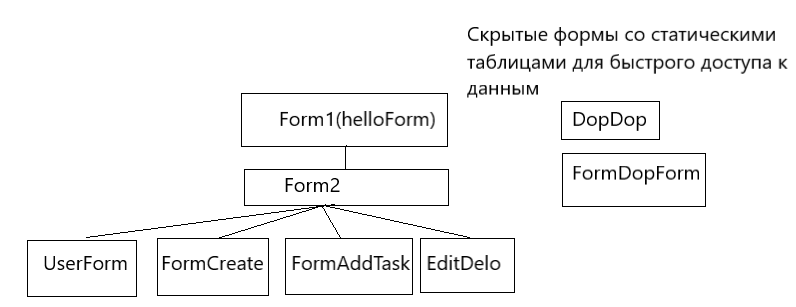


Рис.10.

Пользовательский интерфейс реализуется благодаря окнам и панелям приложения, каждое из которых реализует определенные функции приложения:

Окно приветствия позволяет создать новую учетную запись через панель регистрации panel\_registr и перейти к главному окну через панель входа в учетную запись panel\_vhod (также здесь возможен вызов окна помощи, в котором написана инструкции для регистрации и входа)

Главное окно осуществляет работу с тремя панелями (панель дня, недели и месяца), а также позволяет перейти в окна добавления записи, изменения записи, создания отчета, окно данных пользователя, выход к окну приветствия, окно помощи.

(скриншоты окон и панелей см. приложение А)

## **3.3. Реализация основного функционала**

В основе работы программы лежат два класса DopDop и FormDopForm, которые наследуются от класса Form, они содержат базовые таблицы с информацией о всех учетных записях и делах конкретного пользователя.

DopDop упрощает работу с данными об учетных записях и содержит:

статическое поле DataGridView (таблица для данных из файла data.dat, в QT tablewidget )

статический метод writefiledela() для записи обновлений данных о пользователе в файл data.dat

FormDopForm упрощает работу с записями пользователя, чья учетная запись активна на данный момент и содержит:

статическое поле DataGridView (таблица для данных из файла пользователя в QT tablewidget\_2)

статический writefile() для записи обновлений задач пользователя

Класс form1, с которого начинается запуск программы, наследуется от класса Form.

Класс form1 содержит 3 панели panel\_vhod, panel\_registr, panel1, экземпляры класса Panel (в QT экземпляры класса Frame).

На panel\_vhod(панель входа) размещены два поля ввода (textBox\_password и textBox\_loginusing), кнопки входа и регистрации (button\_reg и utton\_vhod)

На panel\_registr(панель регистрации) размещены три поля ввода (textBox\_name\_reg – имя пользователя, textBox\_pass\_reg - пароль, textBox\_log\_reg - логин), кнопки входа и регистрации (button\_help\_reg и button\_vhod\_reg)

Также класс form1 содержит две статические переменные:

String LogNow – хранит логин пользователя, чья учетная запись активна (в QT тип QString)

int strnum – хранит id пользователя (необходимо для быстрого доступа к данным о пользователе при изменении или удалении данных)

Методы:

private void button\_help\_Click(object sender, EventArgs e) – вызов помощи для первой формы

private void button\_vhod\_Click(object sender, EventArgs e) – вход в учетную запись

private void button\_reg\_Click(object sender, EventArgs e) – нажатие кнопки регистрации на панели входа для перехода к панели регистрации

private void button\_vhod\_reg\_Click(object sender, EventArgs e) – нажатие кнопки входа на панели регистрации для перехода к панели входа

private void HelloForm\_Load(object sender, EventArgs e) – запуск формы, во время запуска приветственной формы осуществляется заполнение таблицы из класса DopDop для последующей работы с данными

private void button\_reg\_new\_Click(object sender, EventArgs e) – регистрация, создание новой учетной записи

Основной класс Form2 (HelloForm) содержит три панели panel\_DAY(для работы с данными на конкретный день), panel\_dlya\_nedeliiii(для работы с данными на конкретную неделю), panel\_dlya\_mesyaca (работа с данными на месяц)

Методы

private void button\_DAY\_Click(object sender, EventArgs e)

private void button\_WEEK\_Click(object sender, EventArgs e)

private void button\_MONTH\_Click(object sender, EventArgs e)

обеспечивают переход от одной панели к другой.

Панели содержат таблицы типа DataGridView, которые отображают выборочные данные из базовых таблиц в зависимости от выбранной даты.

Таблицы с задачами содержат столбец CheckBox-в для выборки конкретных задач и дальнейшей работы с ними. Работа с выбранными задачами осуществляется с помощью кнопок «удалить», «добавить», «выполнить» и т.д.

Строки, отмеченные как «выполнено», меняют свой цвет.

Чтобы изменить данные о пользователе или удалить учетную запись, необходимо открыть окно пользователя(UserForm), в котором содержится основная информация об учетной записи(информация по id выбирается из базовой таблицы класса DopDop). При изменении данных, меняется содержимое базовой таблицы и перезаписывается информация в файл Data.dat

Чтобы добавить запись, нужно перейти в окно добавления – FormAddTask, заполнить поля без ошибок и нажать кнопку добавить.

Для изменения задачи окно EditDelo.

Для создания отчета окно FormCreate.

## **3.4. Распределение работы.**

Егорова В.Е.: 1. Обзор предметной области

Жукова А.В.: 2. постановка задачи.

Егорова В.Е.: 3. Проектирование.

Жукова А.В.: 4. Определение обобщенной структуры программы, описание выделенных блоков и связей между ними

Вместе 5. чертеж структурной схемы либо схемы работы системы

Верещагина И.А.: 6. Описание функционирования разрабатываемой программы, ее структуры с точки зрения описания данных и кода для их обработки

Вместе 7. Разработка дизайна

Верещагина И.А.: 8. Оформление структуры приложения в Microsoft Visual Studio (определение объектов (кнопок, таблиц и т.д.) в форме)

Жукова А.В.: 9. Взаимосвязь между объектами (написание кода в Microsoft Visual Studio)

Жукова А.В.: 10. Отладка, тестирование в Microsoft Visual Studio

Верещагина И.А.: 11. Оформление структуры приложения в Qt Creator(определение объектов(кнопок, таблиц и т.д.) в форме)

Верещагина И.А.: 12. Взаимосвязь между объектами (написание кода в Qt Creator)

Егорова В.Е.: 13. Оформление отчета о проделанной работе;

Егорова В.Е.: 14. Статья

# **Заключение**

Была создана программа-планировщик «Сделай все», обеспечивающая создание нескольких учетных данных, работу с данными конкретного пользователя, быстрый доступ к расписанию запланированных дел на день, неделю, месяц.

В процессе выполнения задания были подробно изучены методы работы с файлами на языках C# и C++, проведена работа в средах разработки Microsoft Visual Studio 2019 и Qt Creator.

Преимущества при использовании данного приложения:

1. Возможность работы нескольких пользователей;
2. возможность составления отчета завершенных дел за определенный период времени;
3. Приятный дизайн приложения;
4. Интуитивный интерфейс;
5. Оповещения;

Недочёты приложения:

1. Оповещения срабатывают только при запущенном приложении;
2. Нельзя менять дизайн приложения;

Возможны дальнейшие усовершенствования программного продукта с течением времени:

1. Создание более сложной системы оповещений;
2. Добавление настроек дизайна приложения;

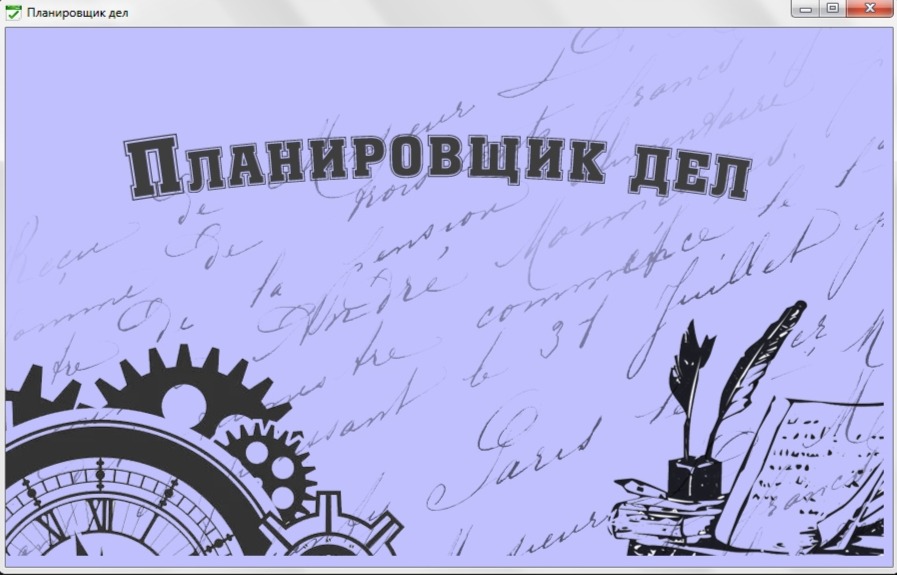
# **Список использованной литературы**

1. Сайт CyberForum [Электронный ресурс]. Режим доступа: http://www.cyberforum.ru/, свободный.
2. Сайт StackOverflow [Электронный ресурс]. Режим доступа: https://ru.stackoverflow.com/, свободный.
3. Сайт MicrosoftDocs.com [Электронный ресурс]. Режим доступа: <https://docs.microsoft.com/ru-ru/dotnet/>, свободный.
4. Сайт Документация По Qt [Электронный ресурс]. Режим доступа: https://doc.qt.io/, свободный.

# **Приложение А. Снимки пользовательского интерфейса**

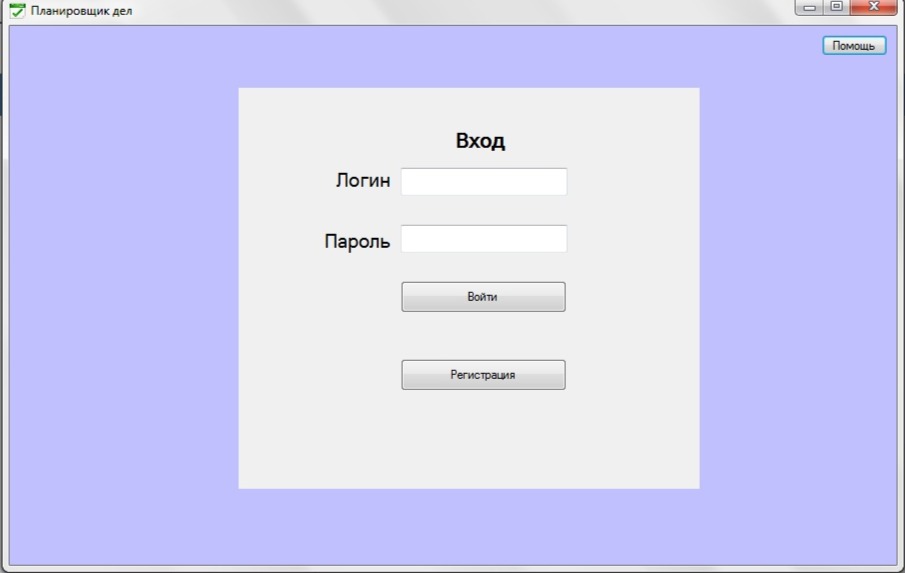
Начальное окно:

Панель приветствия:

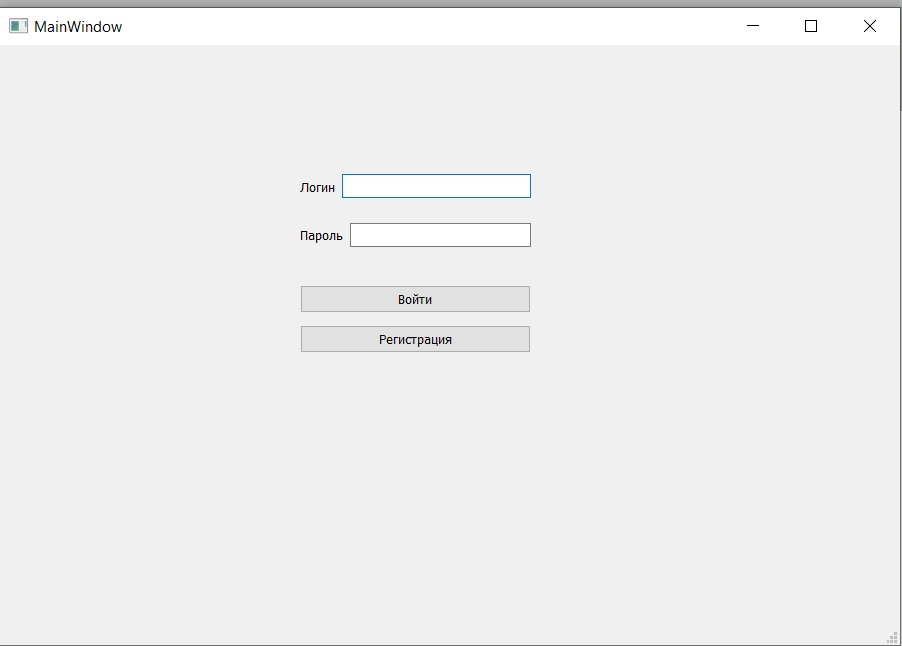


Панель входа:

Microsoft Visual Studio 2019:

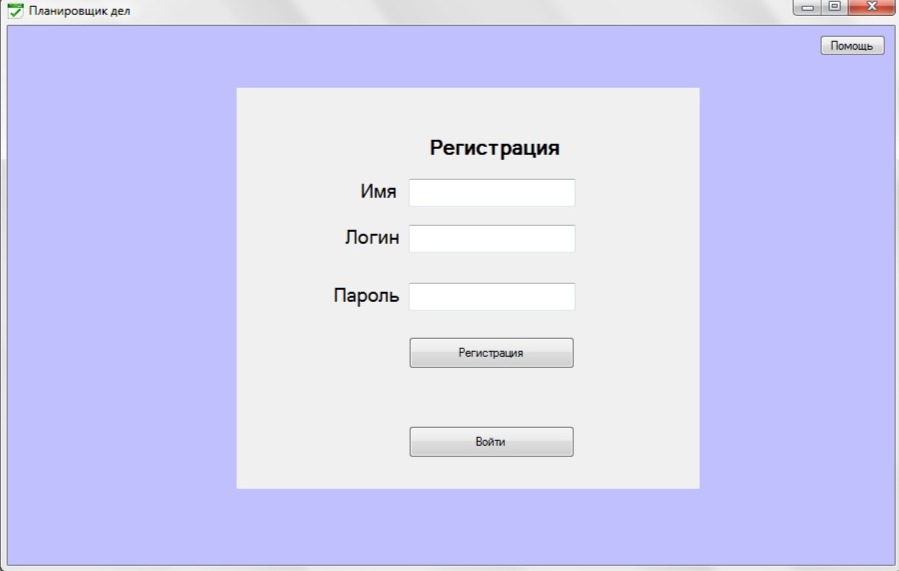


Qt Creator:

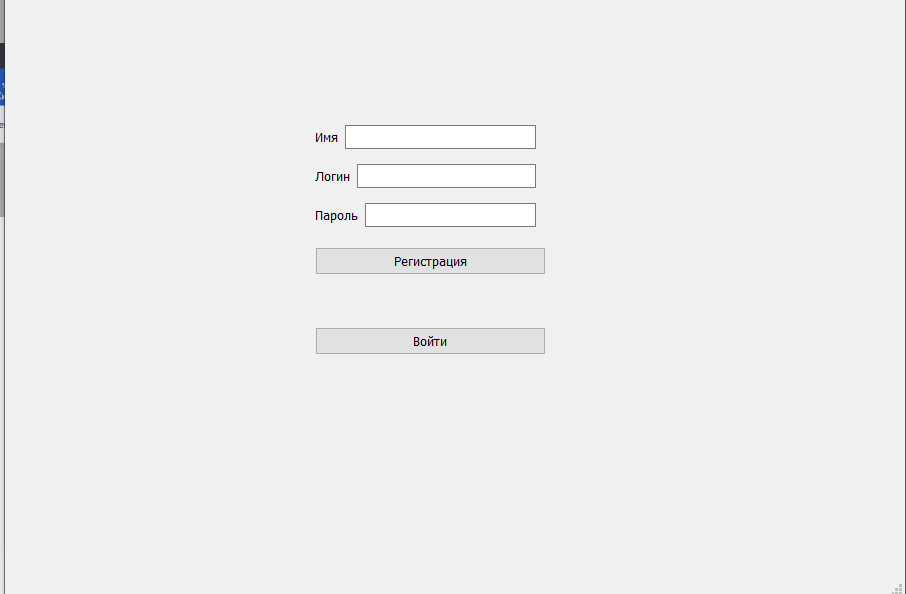


Панель регистрации:

Microsoft Visual Studio 2019:



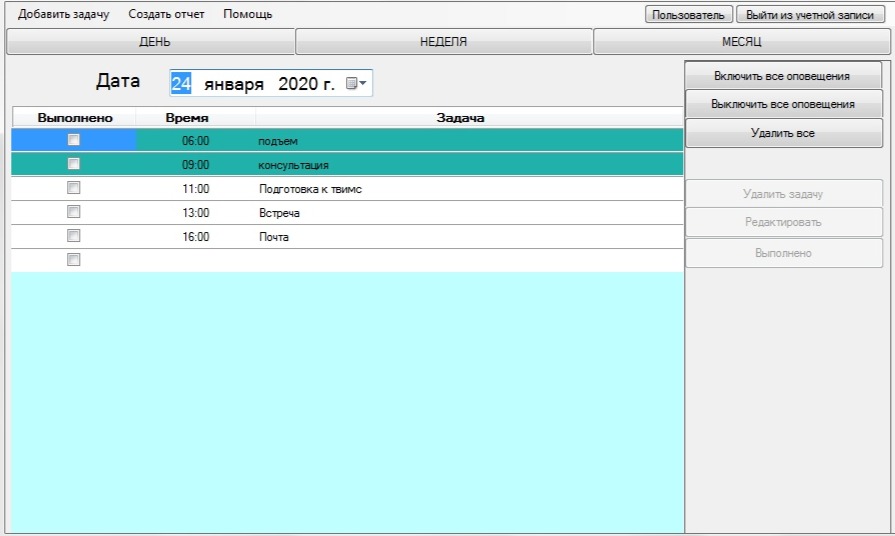
Qt Creator:



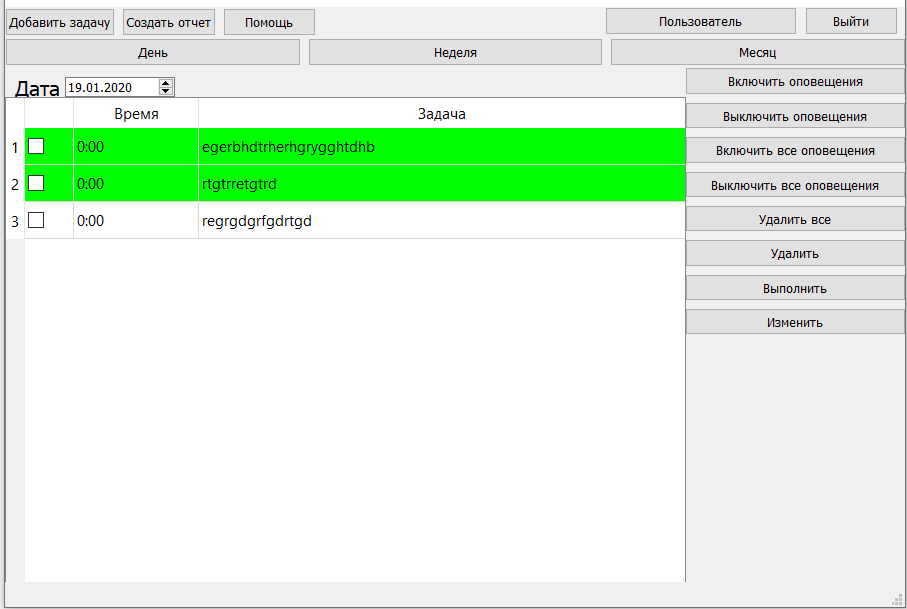
Главное окно:

Панель дня:

Microsoft Visual Studio 2019:

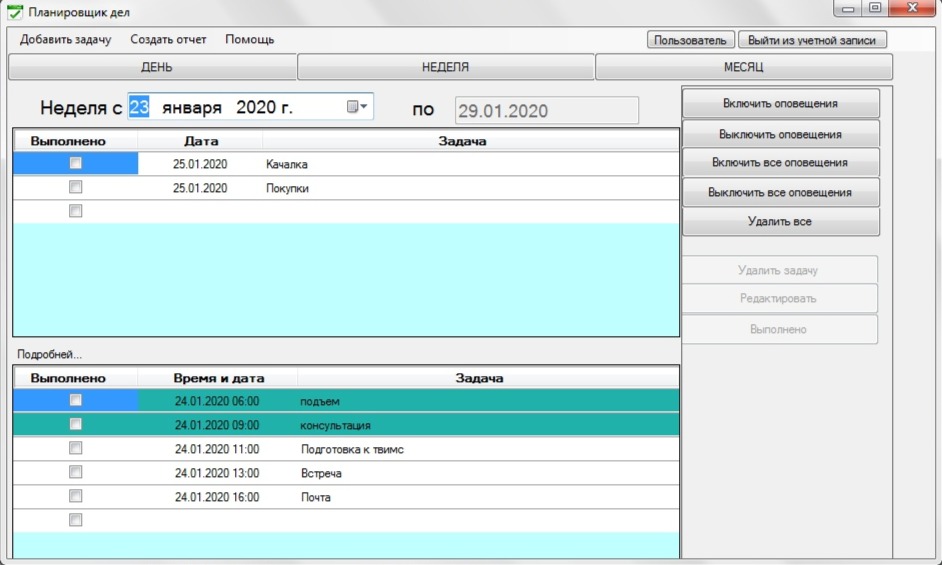


Qt Creator:

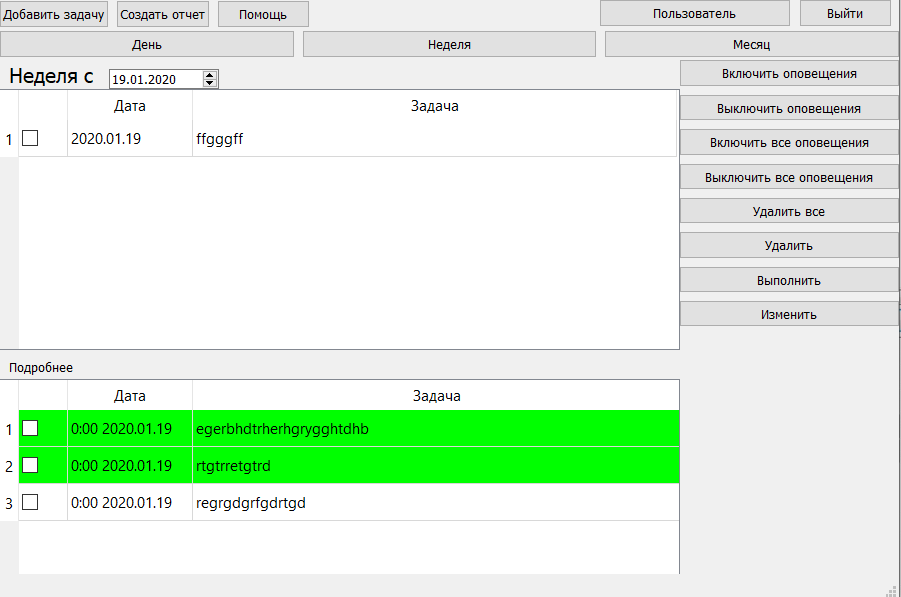


Панель недели:

Microsoft Visual Studio 2019:

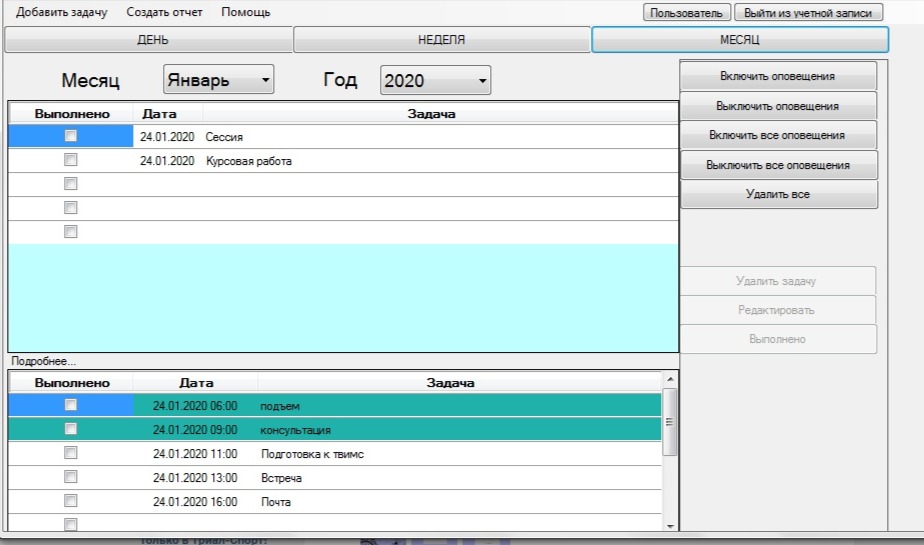


Qt Creator:

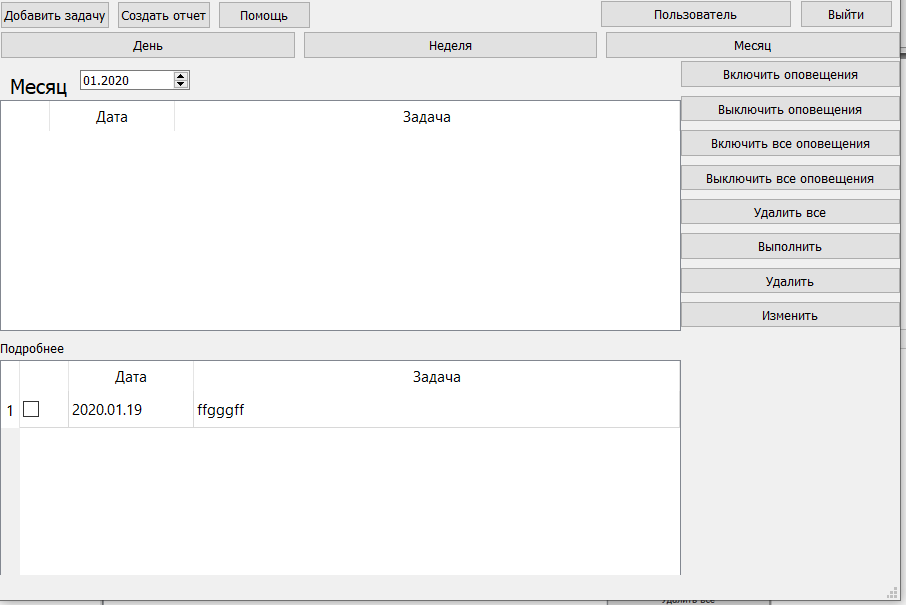


Панель месяца:

Microsoft Visual Studio 2019:



Qt Creator:

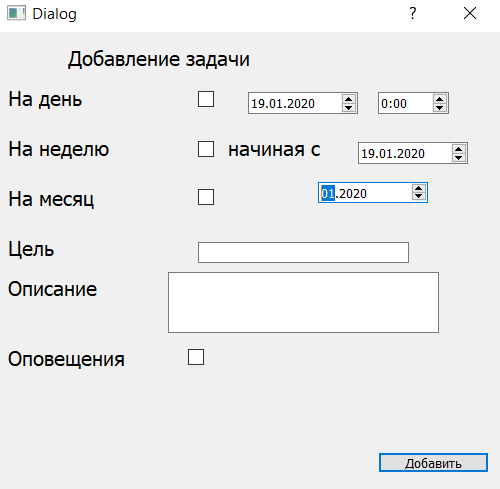


Окно добавления записи.

Microsoft Visual Studio 2019:

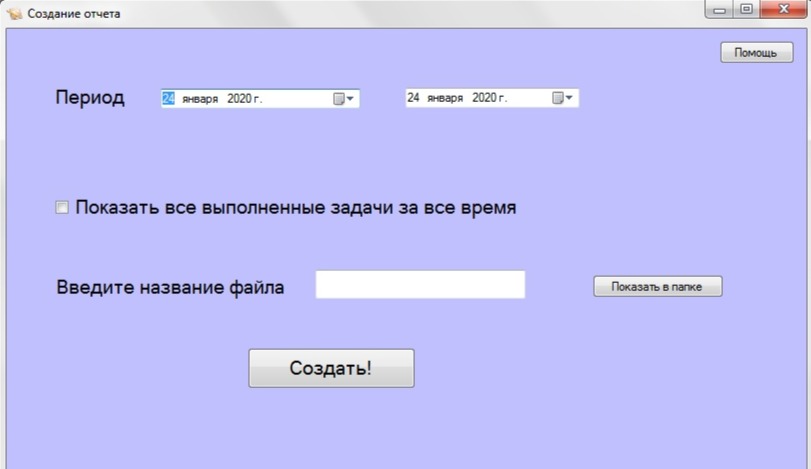


Qt Creator:

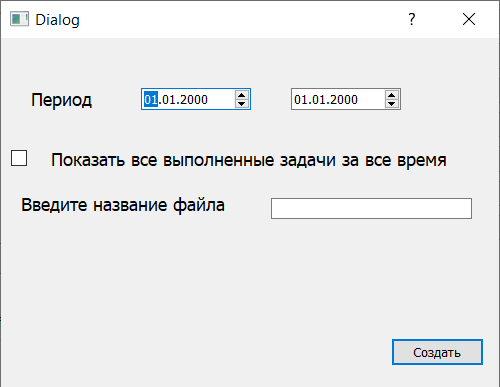


Окно создания отчета.

Microsoft Visual Studio 2019:

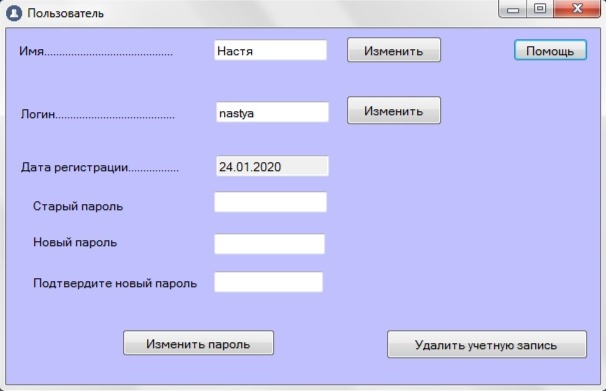


Qt Creator:

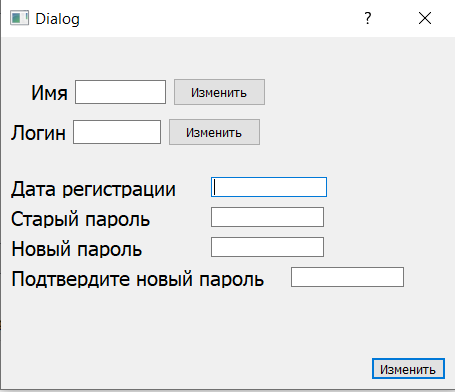


Окно данных пользователя позволяет менять данные о пользователе.

Microsoft Visual Studio 2019:

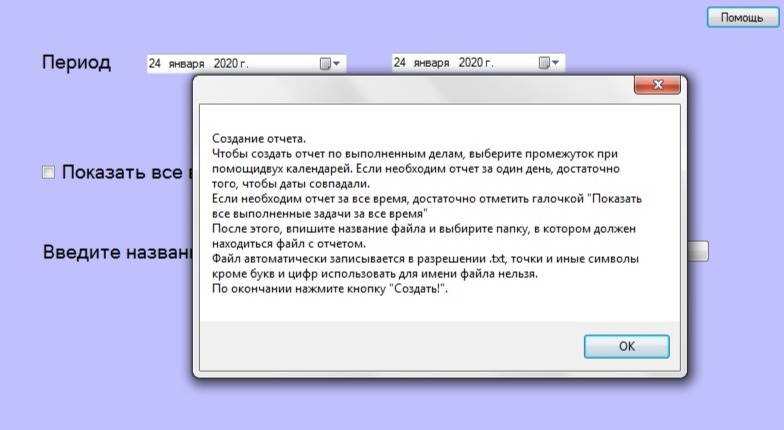
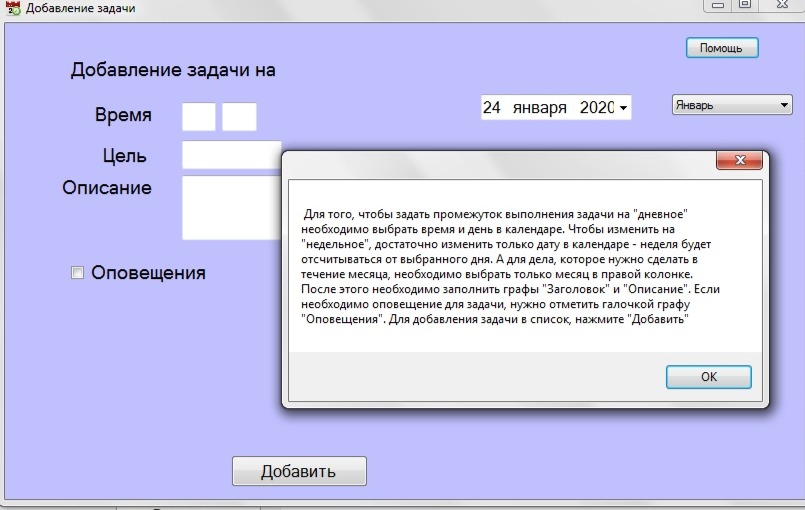
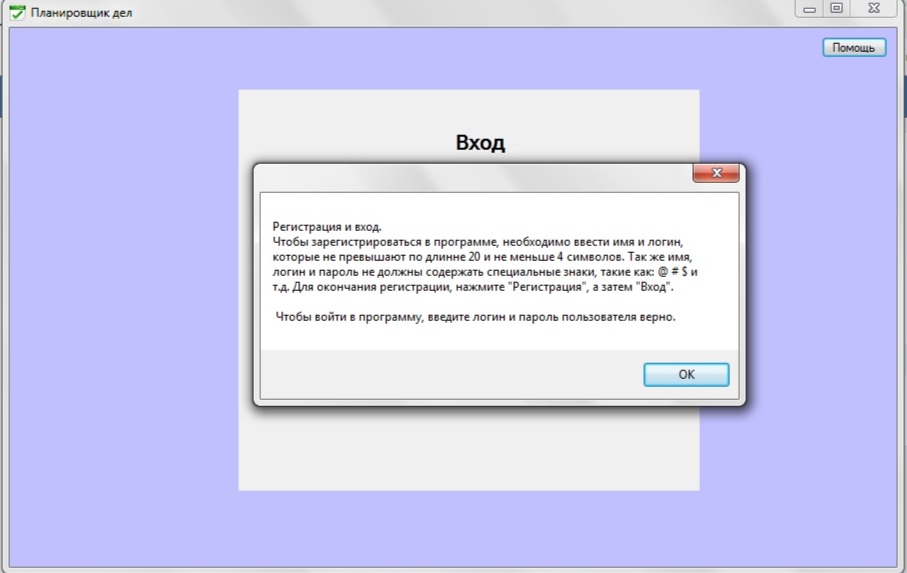


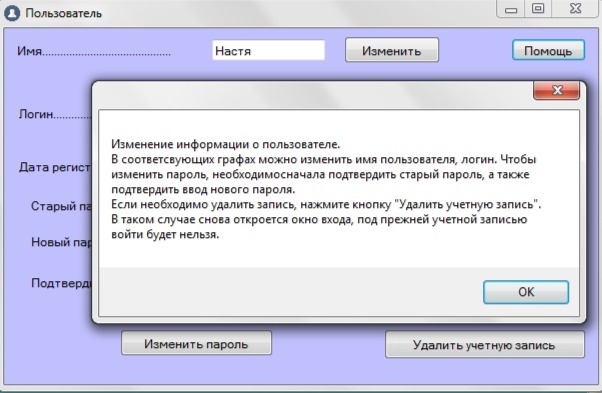
Qt Creator:



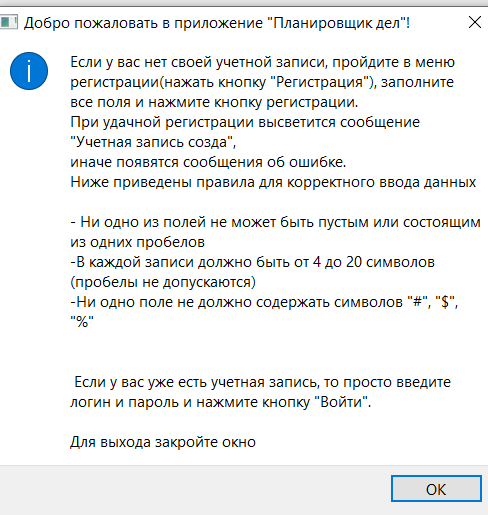
Окна помощи.

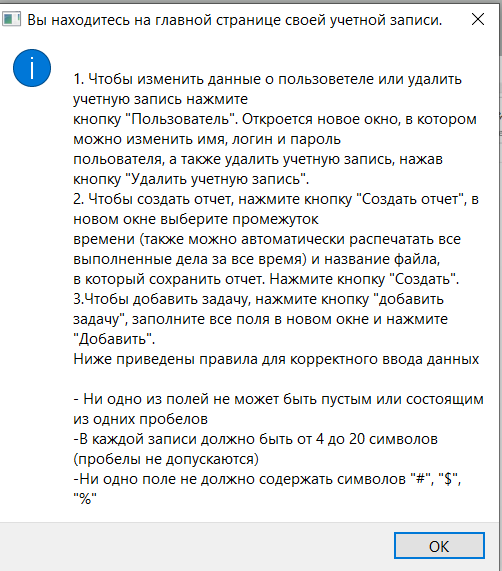
Microsoft Visual Studio 2019:





Qt Creator:





# **Приложение Б. Исходный код программы**

1. VS версия – исходный код находится в папке:

\WindowsFormsApp1\WindowsFormsApp1

2. QT версия – исходный код находится в папке:

\kursovaya

# **Приложение В. Руководство системного администратора**

На переносном диске содержится архивированная папка с программой и

ее исходным кодом. Для работы с данной программой необходимо

разархивировать данную папку и запустить WindowsFormsApp1\WindowsFormsApp1\bin\Debug\ WindowsFormsApp1, находящийся по адресу kursovaya\kursovaya.

Для функционирования программы на ПК требуется:

* ОС Windows XP SP3 или выше;
* Оперативная память 20МБ или выше;
* Жесткий диск: 200МБ для хранения программы;
* Дополнительное место для хранения вводимых в базу данных;
* .NET Framework версии 2.0 или выше;
* Наличие мыши, монитора;
* Разрешение экрана 800 x 600 или более.

# **Приложение Г. Руководство пользователя**

Регистрация и вход.

Чтобы зарегистрироваться в программе, необходимо ввести имя и логин, которые не превышают по длине 20 и не меньше 4 символов.

Так же имя, логин и пароль не должны содержать специальные знаки, такие как: @ # $ и т.д.

Для окончания регистрации, нажмите "Регистрация", а затем "Вход".

Чтобы войти в программу, введите логин и пароль пользователя верно.

Главное окно для работы с делами пользователя.

Чтобы посмотреть задачи на день, нажмите "День". Пользователь может выбирать день,

чтобы посмотреть список запланированных или уже выполненных дел. Так же, пользователь может выбирать при помощи галочек записи, которые необходимо удалить

(при выделении хотя бы одной задачи, кнопка "Удалить" становится доступной). Таким же образом можно отмечать уже выполненные задачи. Чтобы отредактировать задачу, необходимо

ее выбрать и нажать кнопку "Редактировать". За раз редактировать можно только одну задачу!

Чтобы посмотреть список дел, запланированных на неделю, нажмите "Неделя". Пользователь самостоятельно выбирает день, от которого отсчитывается неделя.

В первой таблице будут указаны задачи, запланированные на недельное выполнение.

Во второй таблице будет представлен список дневных дел за указанную неделю. Так же, как и во вкладке "День", кнопки "Удалить", "Редактировать" и "Выполнено" работают аналогично.

посмотреть список дел, запланированных на месяц, нажмите "Месяц". Пользователь самостоятельно выбирает месяц и год.

В первой таблице будут указаны задачи, запланированные на месячное выполнение.

Во второй таблице будет представлен список дневных дел за указанный месяц. Так же, как и во вкладке "День", кнопки "Удалить", "Редактировать" и "Выполнено" работают аналогично.

добавить новую задачу в любой список, нажмите кнопку "Добавить задачу" в верхнем ленточном меню. Чтобы создать отчет по конкретному промежутку времени, нажмите кнопу "Создать отчет". Для того, чтобы изменить информацию о пользователе, сменить пароль, нажмите кнопку "Пользователь". Чтобы сменить пользователя, нажмите кнопку "Выход из пользователя".

Редактирование записи.

Все поля автоматически заполняются информацией о задаче, которую выделил пользователь.

Для того, чтобы изменить промежуток выполнения задачи на \"дневное\" необходимо выбрать время и день в календаре.

Чтобы изменить на "недельное", достаточно изменить только дату в календаре - неделя будет отсчитываться от выбранного дня.

А для дела, которое нужно сделать в течение месяца, необходимо выбрать только месяц в правой колонке.

После этого можно изменить графы "Заголовок" и "Описание", если это необходимо. Так же можно изменить, выполнена ли задача. Если необходимо оповещение для задачи, нужно отметить галочкой графу "Оповещения". Для добавления задачи в список, нажмите "Изменить”.

Добавить запись.

Для того, чтобы задать промежуток выполнения задачи на "дневное" необходимо выбрать время и день в календаре.

Чтобы изменить на "недельное", достаточно изменить только дату в календаре - неделя будет отсчитываться от выбранного дня.

А для дела, которое нужно сделать в течение месяца, необходимо выбрать только месяц в правой колонке.

После этого необходимо заполнить графы "Заголовок" и "Описание". Если необходимо оповещение для задачи, нужно отметить галочкой графу "Оповещения". Для добавления задачи в список, нажмите "Добавить".

Создание отчета.

Чтобы создать отчет по выполненным делам, выберите промежуток при помощи двух календарей. Если необходим отчет за один день, достаточно того, чтобы даты совпадали.

Если необходим отчет за все время, достаточно отметить галочкой "Показать все выполненные задачи за все время". После этого, впишите название файла и выберите папку, в котором должен находиться файл с отчетом. Файл автоматически записывается в разрешении .txt, точки и иные символы кроме букв и цифр использовать для имени файла нельзя. По окончании нажмите кнопку "Создать!".

Изменение информации о пользователе. В соответствующих графах можно изменить имя пользователя, логин. Чтобы изменить пароль, необходимо сначала подтвердить старый пароль, а также подтвердить ввод нового пароля.

Если необходимо удалить запись, нажмите кнопку "Удалить учетную запись". В таком случае снова откроется окно входа, под прежней учетной записью войти будет нельзя.