

- устаревший дизайн может показаться непривлекательным современным пользователям;
- отсутствует возможность редактировать уже добавленную задачу.

Программа должна быть простой в установке и настройке, с современным интерфейсом, поддержкой дедлайнов и статусов задач. Функционал: создание, редактирование, удаление и отметка выполненных задач с автосохранением.

## 4.2 Разработка структуры приложения и алгоритмов функционирования

В данном разделе представлена структура программы и соответствующие диаграммы.



Рисунок 29 – Алгоритм действий пользователя

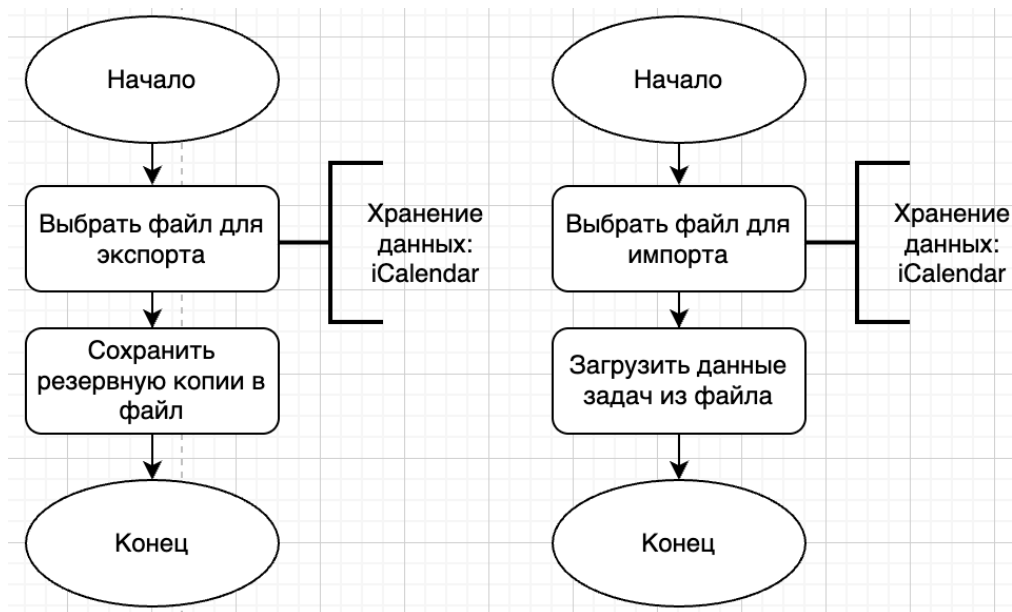


Рисунок 30 – Алгоритмы экспорта и импорта задач

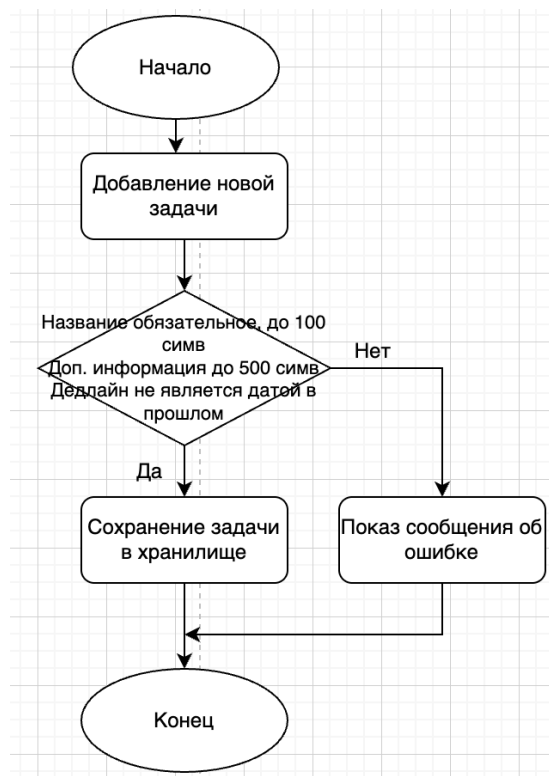


Рисунок 31 – Алгоритм добавления новой задачи

Описание алгоритма действий пользователя:

- Начало – запуск процесса управления задачами;
- Отобразить список задач – в главном окне программы отображается список задач с их параметрами;

- Добавить новую задачу – в отдельном диалоговом окне заполняются параметры задачи: название (обязательно), информация (до 500 символов), дата выполнения (не может быть прошлой);
- Редактировать задачу – при двойном клике на задачу открывается диалоговое окно, аналогичное добавлению, где можно изменить параметры;
- Отметить задачу как выполненную – в диалоговом окне редактирования по нажатию кнопки статус задачи меняется на «Выполнена»;
- Экспортировать задачи в файл iCalendar – данные задач сохраняются в выбранный файл (например, в формате .ical);
- Импортировать задачи из файла iCalendar – данные из выбранного файла загружаются в программу, обновляя список задач;
- Конец – завершение работы с задачами.

#### **4.2.1 Алгоритм импорта данных из .ical файла**

При выборе .ical файла с помощью диалога открытия сначала из него считывается текст (iCalendar формат является текстовым описанием событий). Затем данный текст передаётся в библиотеку icalendar для извлечения данных в JSON-формат. При успешной загрузке данных производится замена данных в хранилище на импортированные задачи. При ошибке работы с данными сохраняются старые задачи, во избежание повреждения данных.

#### **4.2.2 Алгоритм вывода списка задач**

При перезагрузке списка (например, запуске программы, добавлении новой задачи и других событиях) в начале очищается элемент QListWidget. После этого читается список задач из хранилища, и с помощью цикла на каждую задачу создаётся новый компонент TaskItem (состоит из QVBoxLayout, QLabel). В этот компонент передаются данные задачи (название, доп. текст, статус и так далее). Если у задачи указана дата, производится проверка, что дата не в

прошлом. Если дата в прошлом – задача считается пропущенной и статус автоматически меняется на «Просрочена». После этого TaskItem создаёт нужные элементы с данными задачи, и добавляется в список. По завершении отображения всех задач в списке становятся доступны действия с задачами.

### 4.2.3 Разработка экранных форм приложения

В данном разделе представлены разработанные экранные формы приложения:

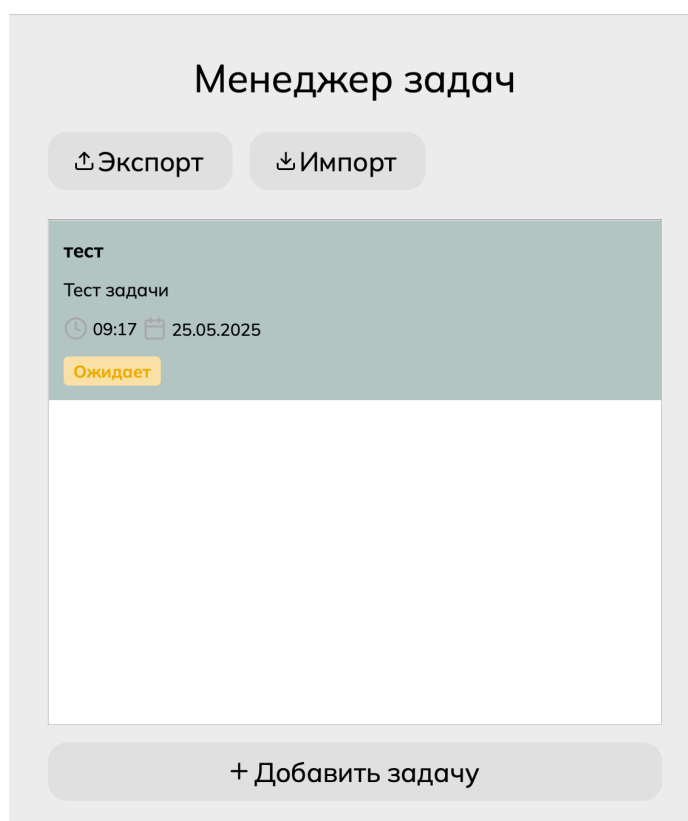


Рисунок 32 – Макет главного окна программы

## ЗАКЛЮЧЕНИЕ

В ходе практики я получил представление о процессе разработки прикладного программного обеспечения, включая методы проектирования, используемые инструменты и требования при разработке технической документации (техническое задание, программа и методика испытаний, руководство пользователя, отчёт). Я приобрёл навыки, необходимые для создания ПО и оформления документации для приёмо-сдаточных испытаний, что позволит применять их в профессиональной деятельности.

В процессе проектирования осуществлена разработка формализованных алгоритмов работы программного комплекса, включая создание функциональных диаграмм, а также детальную проработку сценариев взаимодействия пользователя с системой. Особое внимание было уделено организации корректного рабочего окружения: выполнена установка и настройка системы контроля версий Git, создан репозиторий на платформе GitHub, установлена среда разработки Python с необходимым набором библиотек и средой разработки Visual Studio Code.

На этапе проектирования пользовательского интерфейса были созданы макеты в Figma, что позволило наглядно представить взаимодействие пользователя с приложением. Разработка кода велась с применением принципов объектно-ориентированного программирования.

В приложении реализованы все запланированные функции: создание, редактирование и удаление задач, категоризация по статусу, сохранение данных между сеансами, а также дополнительные возможности — напоминания, возможность отмечать выполненные задания, создание и загрузка резервных копий.

Завершающим этапом стало комплексное тестирование всех функциональных модулей, включая проверку базовых и дополнительных возможностей и соответствия требованиям технического задания. Результаты тестирования подтвердили готовность приложения к эксплуатации.

Таким образом, программа учебной практики ПМ.05 в период с 15.01.2025 по 19.06.2025 была выполнена мной в полном объеме.

При выполнении работ, предусмотренных заданием на практику, в организации ФГБОУ ВО «Вятский государственный университет» мной, Зараменских Ильёй Алексеевичем, соблюдались правила внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологические правила и гигиенические нормативы. Пройден вводный инструктаж и инструктаж на рабочем месте. Замечаний не имел.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Менеджмент – Википедия [Электронный ресурс]. – URL: <https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D0%BD%D0%B5%D0%B4%D0%B6%D0%BC%D0%B5%D0%BD%D1%82> (дата обращения: 28.05.2025)
2. Почему Python такой популярный [Электронный ресурс]. – URL: <https://tproger.ru/articles/pochemu-python-takoj-populyarnyj-napisano-chelovekom> (дата обращения: 28.05.2025)
3. Зачем нужен Git и GitBash? [Электронный ресурс]. – URL: <https://qna.habr.com/q/913527> (дата обращения: 21.05.2025)
4. Qt Documentation [Электронный ресурс]. – URL: <https://doc.qt.io/> (дата обращения: 11.05.2025)
5. Разработка на Pyside6 [Электронный ресурс]. – URL: <https://habr.com/ru/articles/799037/> (дата обращения: 11.05.25)
6. Документация по Python [Электронный ресурс]. – URL: <https://education.yandex.ru/handbook/python> (дата обращения: 06.05.25)
7. Приложение с GUI на Python [Электронный ресурс]. – URL: [https://www.youtube.com/watch?v=au01t\\_WOe\\_Q](https://www.youtube.com/watch?v=au01t_WOe_Q) (дата обращения: 13.05.25)