Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

РАЗРАБОТКА ПРИЛОЖЕНИЯ NOTEAPP

Документация к лабораторной работе по дисциплине

«Новые технологии в программировании»

Студент гр. 589-2

\_\_\_\_\_\_\_\_\_ В. А. Кабаев

\_\_\_\_\_\_\_\_\_

Руководитель

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_ А. Е. Горяинов

(оценка) \_\_\_\_\_\_\_\_\_

Томск 2022 г

**Оглавление**

[1 Назначение приложения 3](#_Toc106455524)

[2 Группы пользователей и их функциональные возможности в приложении 4](#_Toc106455525)

[3 Стек технологий и системные требования 6](#_Toc106455526)

[4 Пользовательский интерфейс 7](#_Toc106455527)

[5 Архитектура приложения 8](#_Toc106455528)

[6 Диаграммы классов приложения 9](#_Toc106455529)

[7 Тестирование приложения 10](#_Toc106455530)

[8 Сборка установщика 12](#_Toc106455531)

[9 Описание модели ветвления 14](#_Toc106455532)

[10 Заключение 16](#_Toc106455533)

[Список литературы 17](#_Toc106455534)

# 1 Целевая задача приложения

Пользовательское приложение NoteApp, предназначено для ведения персональных записей и заметок. Приложение должно:

* 1. Обеспечивать стабильную работу приложения при порядке 200 заметок.
  2. Обеспечивать категоризацию заметок, навигацию по созданным заметкам.
  3. Предоставить инструменты для просмотра и редактирования заметок.
  4. Сохранять и восстанавливать заметки между сессиями приложения.
  5. Выполнять промежуточные сохранения заметок на машине пользователя на случай аварийного завершения программы, отключения компьютера и т.д. для защиты от потери данных.

Приложение-референс: десктоп-версия программы Evernote.

# 2 Группы пользователей и их функциональные возможности в приложении

Данное приложение разработано для пользователей ПК.

После запуска приложения перед пользователем появляется главное окно. Двухколоночная верстка главного окна содержит список всех созданных заметок в левой панели и отображает текущую выбранную заметку в правой панели. В списке заметок показаны названия заметок, в один момент времени может быть выбрана только одна заметка (далее – текущая заметка).

На панели со списком заметок внизу располагаются три кнопки в виде пиктограмм: Add Note («Создать новую заметку»), Edit Note («Редактировать текущую заметку»), Delete Note («Удалить текущую заметку»). При выборе заметки в списке, выбранная заметка отображается в правой панели. Главное окно не позволяет редактировать содержимое заметки – только просмотр.

При нажатии на кнопку Add Note и Edit Note появляется окно создания/редактирования заметки в диалоговом режиме. Для новой заметки окно изначально не заполнено (Установлены лишь название заметки по умолчанию, дата создания и дата редактирования). Для редактирования уже существующей заметки все поля должны быть предзаполнены данными текущей заметки.

При нажатии на кнопку OK окно создания заметки закрывается, в список заметок главного окна добавляется новая заметка. При редактировании текущей заметки, нажатие на кнопку OK должно обновить название заметки в списке заметок (если название текущей заметки было изменено), и обновить отображаемую заметку в правой панели приложения.

При нажатии кнопки Cancel создание/редактирование заметки отменяется (новая заметка не добавляется, исходная заметка остается без изменений).

В случае ввода пользователем некорректных данных (нарушение допустимой длины названия заметки), данная ситуация должна быть обработана соответствующим образом.

При нажатии на кнопку Remove Note главного окна текущая запись удаляется. Перед удалением должно появиться окно с запросом на разрешение записи: «Do you really want to remove this note: <Название текущей записи>». При нажатии на кнопку OK происходит удаление, при нажатии на кнопку Cancel удаление отменяется.

# 3 Стек технологий и системные требования

Язык программирования C#.NET Framework 4.5.

Тестирование производилось в ручном режиме путем создания заметок и реализации различных ситуаций.

Для выполнения сериализации и десериализации использовалась библиотека Newtonsoft.Json.

Для создания установщика использовался пакет InnoSetup.

Системные требования аналогичны требованиям ОС Windows 10.

# 4 Пользовательский интерфейс

# 

Рисунок 4.1 – Главное окно программы

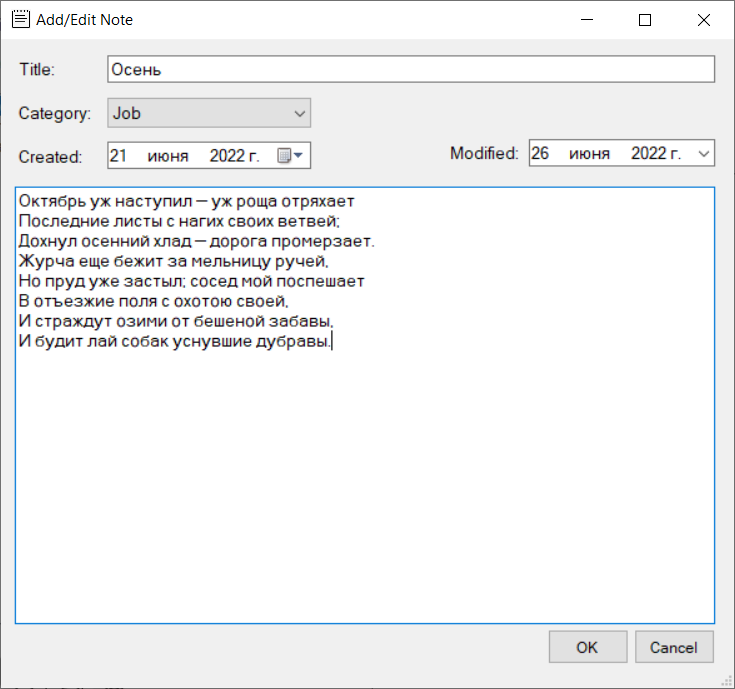


Рисунок 4.2 – Окно создания или редактирования заметки

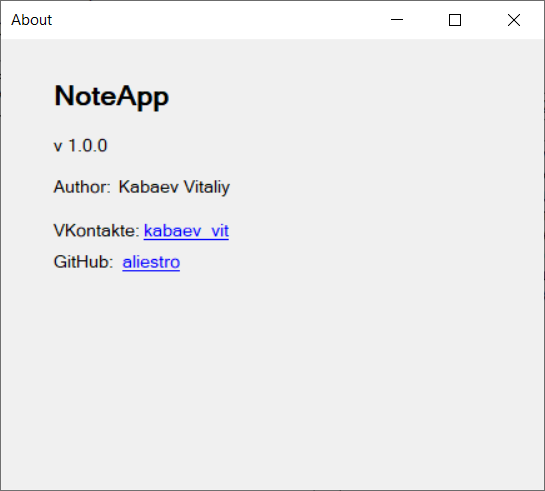


Рисунок 4.2 – Окно информации о авторе продукте

# 5 Архитектура приложения

Диаграмма пакетов приложения представлена на рисунке 5.1.

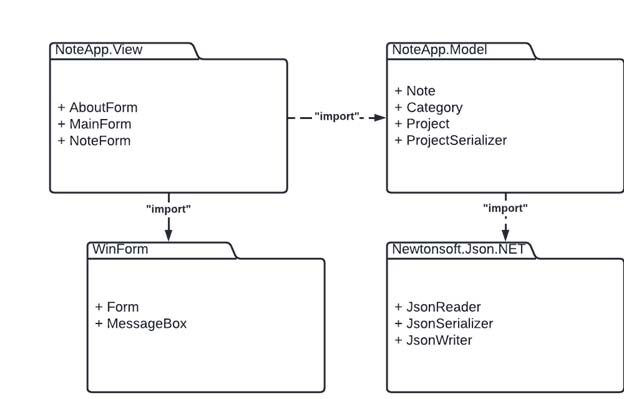
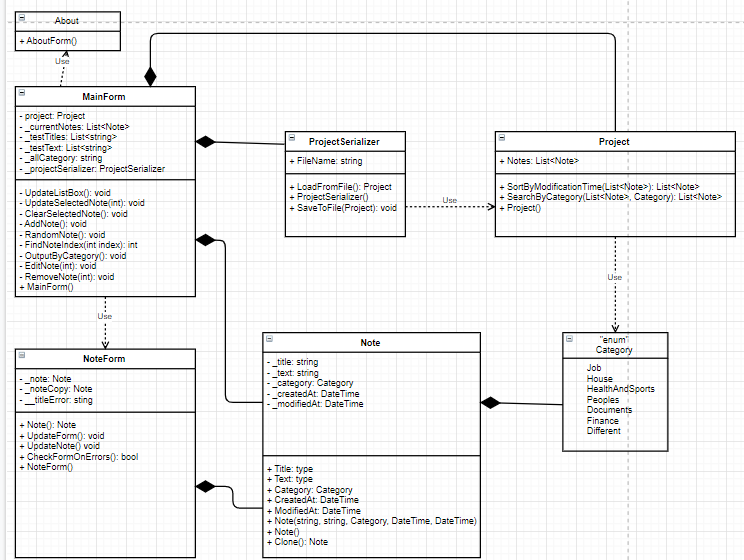


Рисунок 5.1 – Диаграмма пакетов приложения

# 6 Диаграммы классов приложения

Рисунок 6.1 – Диаграммы классов приложения

На рисунке 6.1 показана диаграмма классов приложения.



**7 Тестирование приложения**

Тестирование приложения было проведено вручную. В качестве общего системного теста, проверяющего работоспособность всего приложения, было проведено приемочное тестирование. Для приемочного тестирования необходимо выполнить следующую последовательность действий:

1. Установить приложение на компьютер с помощью собранного установочного пакета.

2. Запустить приложение. Окно программы должно быть пустым – в приложении не должно быть заметок.

3. Создать в приложении три заметки разных категорий.

4. Переключиться между заметками, чтобы убедиться, что смена текущей заметки происходит корректно.

5. Переключить отображаемую категорию заметок – в списке заметок должны остаться только заметки целевой категории. Снова отобразить все категории заметок – список заметок должен восстановиться.

6. Выбрать вторую заметку и нажать кнопку редактирования. Должно открыться окно редактирования заметки.

7. Ввести название заметки более 50 символов. Элемент управления с названием заметки должен указать на некорректное значение.

8. Ввести название заметки менее 50 символов. Элемент управления с названием заметки должен стать корректным.

9. Поменять текст заметки. Нажать «OK». Отредактированная заметка должна подняться в списке заметок на первую позицию, время изменения заметки должно поменяться, отображаемый текст заметки также должен измениться.

10. Выбрать вторую заметку и нажать кнопку редактирования. Должно открыться окно редактирования. Изменить название заметки, ее текст и категорию. Нажать «Cancel». Исходная заметка должна остаться без изменений.

11. Удалить третью заметку.

12. Закрыть приложение. Должно произойти сохранение заметок в целевой файл.

13. Запустить приложение. В программе должны восстановиться заметки, созданные в предыдущую сессию.

# 8 Сборка установщика

Для того, чтобы создания сценарий сборки инсталлятора, использовалась расширение Microsoft Visual Studio Installer Projects. После этого к решению проекта добавить новый проект (Setup Project) и называем его NoteAppSetup. После этого в папке Application Folder добавляем выходной элемент проекта. Нажимаем правой кнопкой мыши по созданному файлу и Create Shortcut ( называем его NoteApp) перемещаем этот файл в папку User’s Programs Menu/App, а также делаем такую же копию в Users Desktop. Как туда перейти можно увидеть на рисунке 8.1.



Рисунок 8.1 – Файловая система

После этого в обозревателе решений нажимаем правой кнопкой по NoteAppSetup и собираем его. На рисунке 8.2 показана сборка.

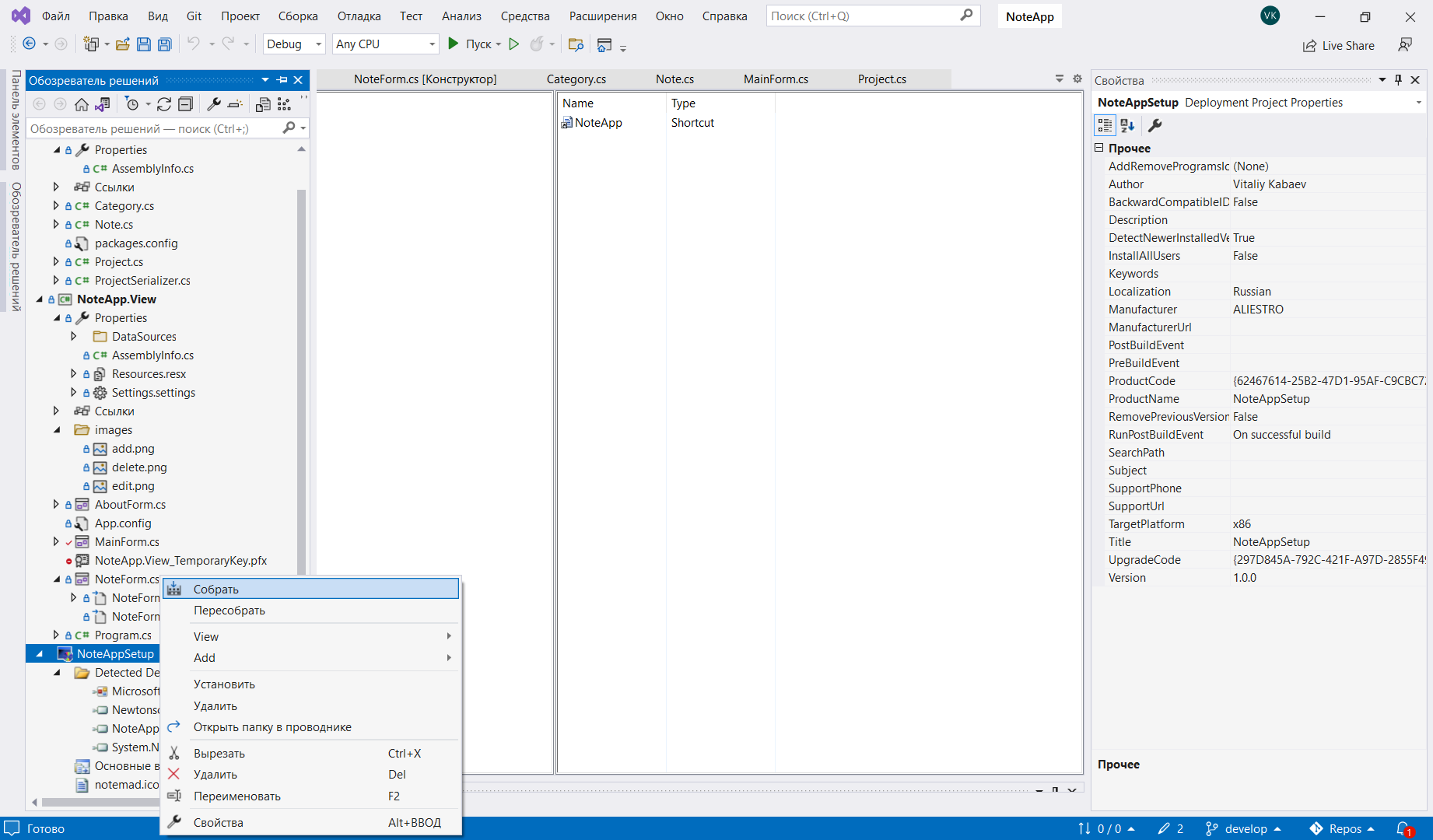


Рисунок 8.1 – Сборка установщика

Теперь мы можем перейти в папку NoteAppSetup проводнике и увидеть, зайти в папку Debug и увидеть два файла сам установщик приложения NoteApp и пакет установки. Мы можем видеть их на рисунке 8.3.

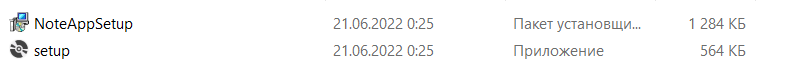


Рисунок 8.3 – Расположение установщика

# 9 Описание модели ветвления

Во время разработки программы для обеспечения версионного контроля использовался сервис GitHub.

Ссылка на репозиторий: https://github.com/aliestro/NoteApp

Для разработки использовалось девять ветвей: develop – главная ветвь, содержащая стабильную и отлаженную версию программы, а также ветви features – ветви, для разработки и тестирования нового функционала. На рисунке 9.1 показаны ветви в репозитории на GitHub.

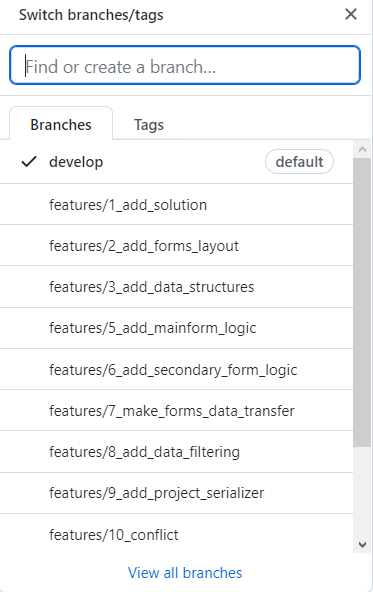


Рисунок 9.1 – Ветви репозитория

На рисунке 9.2 показана история фиксаций ветки develop.

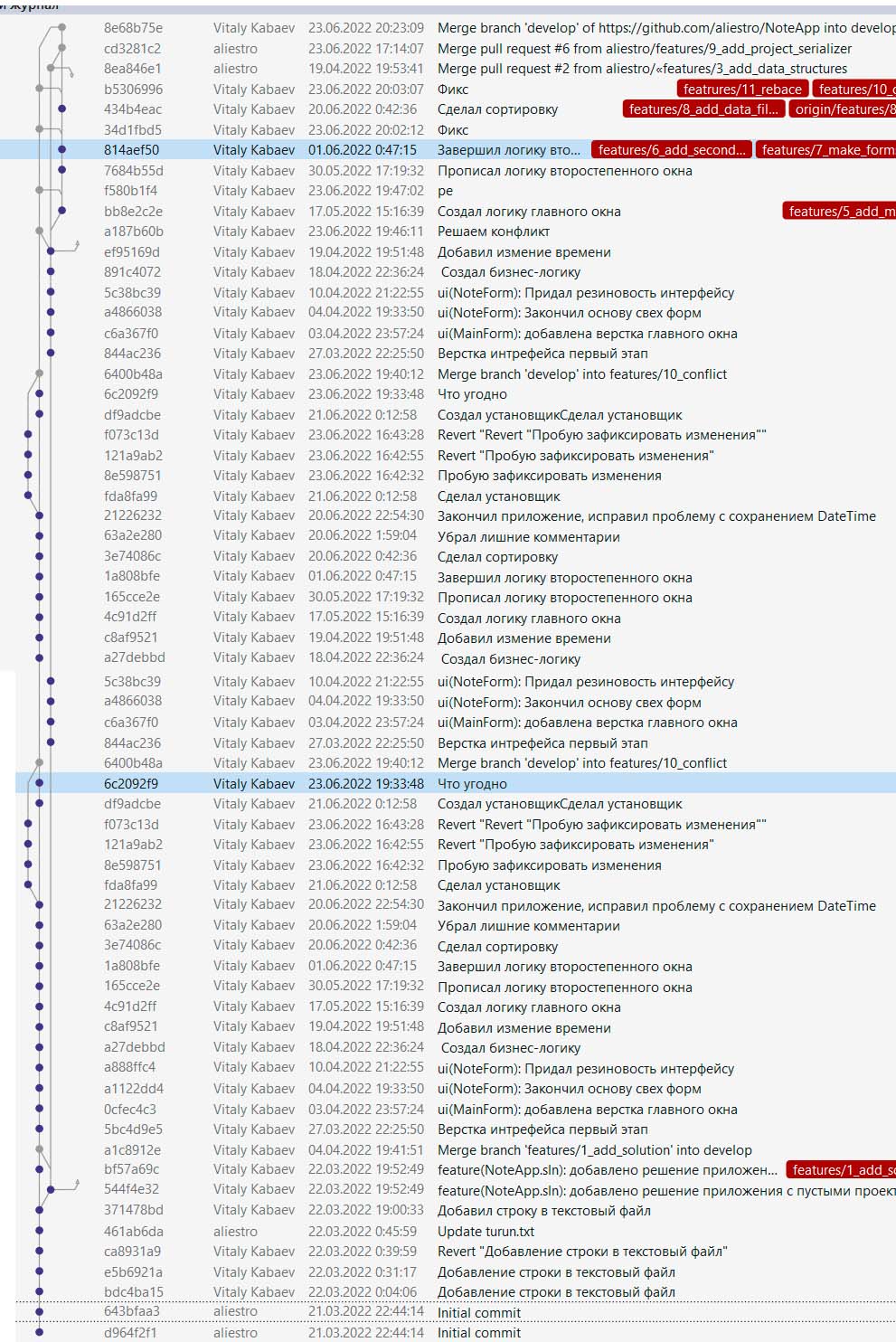


Рисунок 9.2 – История фиксации ветки develop

# 

# 10 Заключение

Поставленная задача была выполнена в полном объёме согласно представленному ТЗ.

Поставленная задача была выполнена в срок. Для выполнения всех задач было потрачено 50-60 часов. Большую часть времени заняла разработка бизнес-логики проекта и на создание функциональности интерфейса приложения. На написание документации у меня ушло 4 часа. Работу удавалось выполнять самостоятельно, искав необходимую информацию в интернете и советовавшись с одногруппниками.

ТЗ было написано достаточно подробно и понятно для выполнения поставленной задачи, в случае возникновения вопроса можно было обратиться к преподавателю и получить помощь и доходчивые объяснения. Все темы в заданиях были раскрыты должным образом.

Проблем с версионным контролем возникали решились путем ввода некоторых команд в консоли git-cmd.

Проблемы не касающиеся разработки: преподаватель не ставил в приоритет работу в сфере обучения студентов, был занят основной работой team lead сеньор разработчиком в другой компании. С другой стороны понятно и интересно подавал информацию.

Общее заключение: на протяжении разработки приложения был получен полезный опыт в сфере разработки приложений, улучшились навыки работы с win form, в целом с языком C#, а также приобрел навыки и знания работы с github и системой версионного контроля. Свою продуктивность в течение семестра оцениваю положительно.

# Список литературы

1. Команды Windows [Электронный ресурс]. – URL: [https://docs.microsoft.com/ru-ru/windows-server/administration/windows-](https://docs.microsoft.com/ru-ru/windows-server/administration/windows-commands/windows-commands) [commands/windows-commands](https://docs.microsoft.com/ru-ru/windows-server/administration/windows-commands/windows-commands)
2. GitHub [Электронный ресурс]. – URL: https://github.com