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)

Пояснительная записка к приложению NoteApp

По дисциплине

«Новые технологии в программировании» (НТвП)

Выполнил: Студент гр. 587–2

Волков Л. Н.

«\_\_» 2020 г.

Принял:

к.т.н., доцент каф. КСУП

Горяинов А.Е.

«\_\_» 2020 г.

Томск 2020

Содержание

[1 Назначение приложения. 3](#_Toc61538119)

[2 Группы пользователей и их функциональные возможности в приложении 3](#_Toc61538120)

[3 Стек технологий разработки. Системные требования 3](#_Toc61538121)

[4 Пользовательский интерфейс 4](#_Toc61538122)

[5 Диаграммы пакетов приложения 8](#_Toc61538123)

[6 Диаграммы классов приложения 9](#_Toc61538124)

[7 Описание тестирования приложения 10](#_Toc61538125)

[8 Описание сборки установщика 12](#_Toc61538126)

[9 Описание модели ветвления в репозитории проекта 14](#_Toc61538127)

# 1 Назначение приложения.

Пользовательское приложение NoteApp, предназначено для ведения персональных записей и заметок

# 2 Группы пользователей и их функциональные возможности в приложении

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 рассчитано только на один тип пользователей: пользователей ПК.

# 3 Стек технологий разработки. Системные требования

Для реализации проекта был выбран язык программирования C# на платформе .NET Framework 4.7.2, набор библиотек Windows Forms для создания десктоп-приложений для операционной системы Windows и среда разработки Visual Studio 2019. Системные требования к приложению ограничены требованиями к .NET Framework.

Для выполнения сериализации и десериализации использовалась библиотека Newtonsoft.Json версии 12.0.3

Для проведения юнит-тестирования приложения использовалась библиотека NUnit версии 3.12.0.

Для создания сценария сборки установочного пакета и компиляции установщика используется программа Inno Setup версии 6.1.2.

# 4 Пользовательский интерфейс

В приложении, для осуществления взаимодействия с ним, было реализовано три окна: Главное окно, окно создания/редактирования заметки и окно «About». Верстка главного окна и окна создания/редактирования заметки должна быть адаптивной. Окно «About» имеет фиксированный размер.

После запуска приложения перед пользователем появляется главное окно (рисунок 4.1). 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.

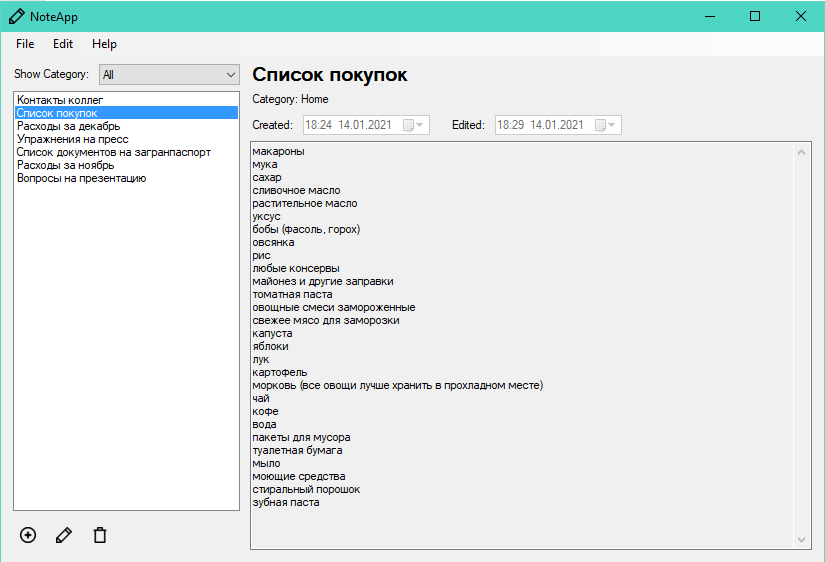


Рисунок 4.1 – Интерфейс главного окна приложения

На панели со списком заметок внизу располагаются три кнопки в виде пиктограмм: Add Note («Создать новую заметку»), Edit Note («Редактировать текущую заметку»), Remove 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 (рисунок 4.2). Для новой заметки окно изначально не заполнено (Установлены лишь дата создания и дата редактирования). Для редактирования уже существующей заметки все поля должны быть предзаполнены данными текущей заметки.

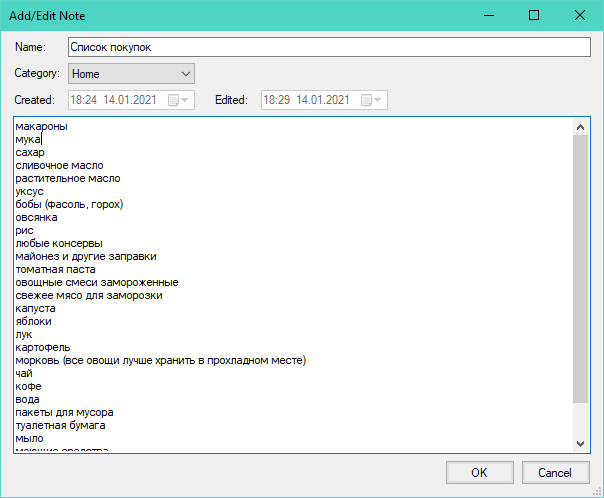


Рисунок 4.2– Интерфейс окна редактирования заметки

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 попытке ввода пользователем некорректных данных – превышение длинны названия заметки (название заметки ограниченно 50 символами), программа не позволит далее вводить данные и подсветит текстовое поле красным цветом.

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

Меню главного окна содержит следующие пункты:

* File:
* Exit (Выйти из приложения – Alt+F4)
* Edit:
* Add Contact (Создать новый контакт)
* Edit Contact (Редактировать текущий контакт)
* Remove Contact (Удалить текущий контакт)
* Help
* About (Вызвать окно «О программе» - F1)

При выборе пункта About, откроется окно содержащее информацию о приложении и его разработчике.

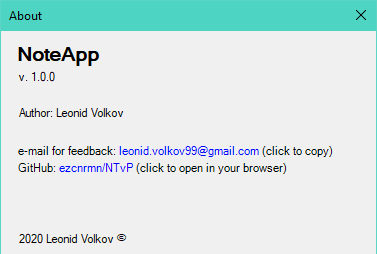


Рисунок 73 – Интерфейс окна «About»

Загрузка заметок осуществляется при запуске программы до вывода главного окна пользователю, сохранение заметок в файл должно выполняться в случаях: а) создания новой заметки; б) удаления заметки; в) закрытии приложения.

# 5 Диаграммы пакетов приложения

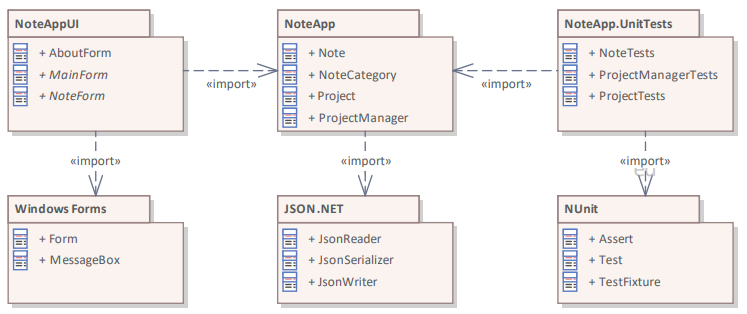


Рисунок 5.1 – Диаграмма пакетов приложения

# 6 Диаграммы классов приложения

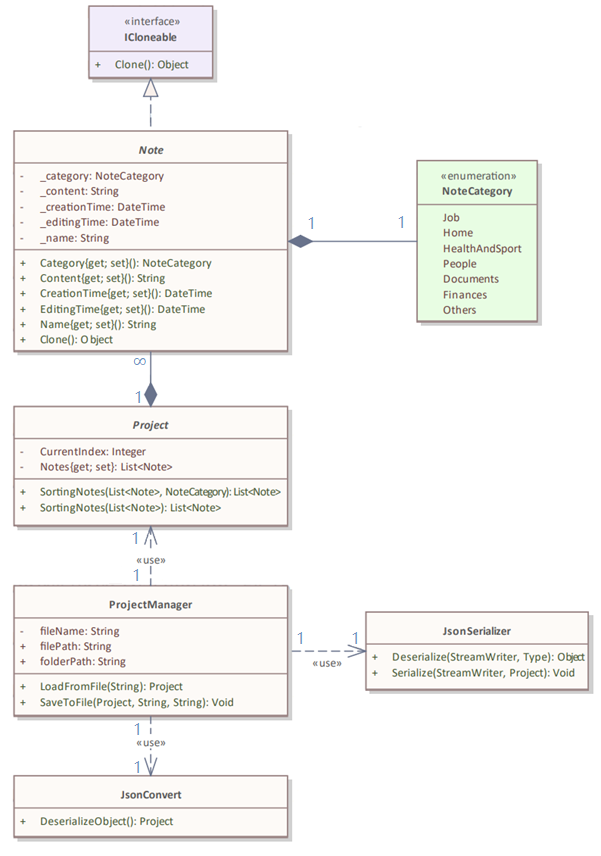


Рисунок 6.1 – Диаграмма классов приложения

# 7 Описание тестирования приложения

При работе над данным проектом проводилось юнит-тестирование. Для него были написаны 14 юнит-тестов, которые покрывают 91% бизнес-логики приложения.

Результаты работы тестов и покрытие представлены на рисунках 7.1 и 7.2

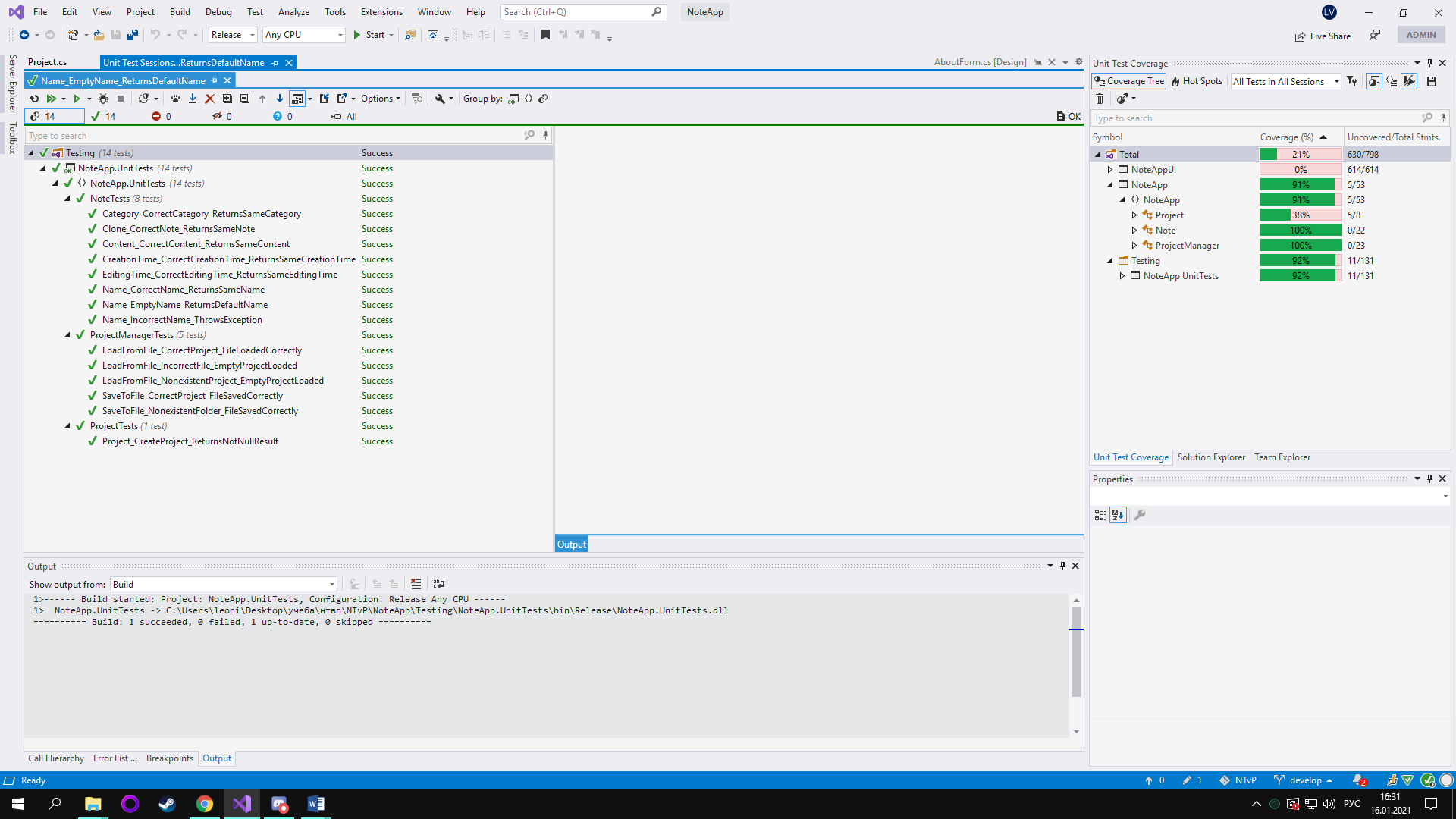


Рисунок 7.1 – Полный список юнит-тестов

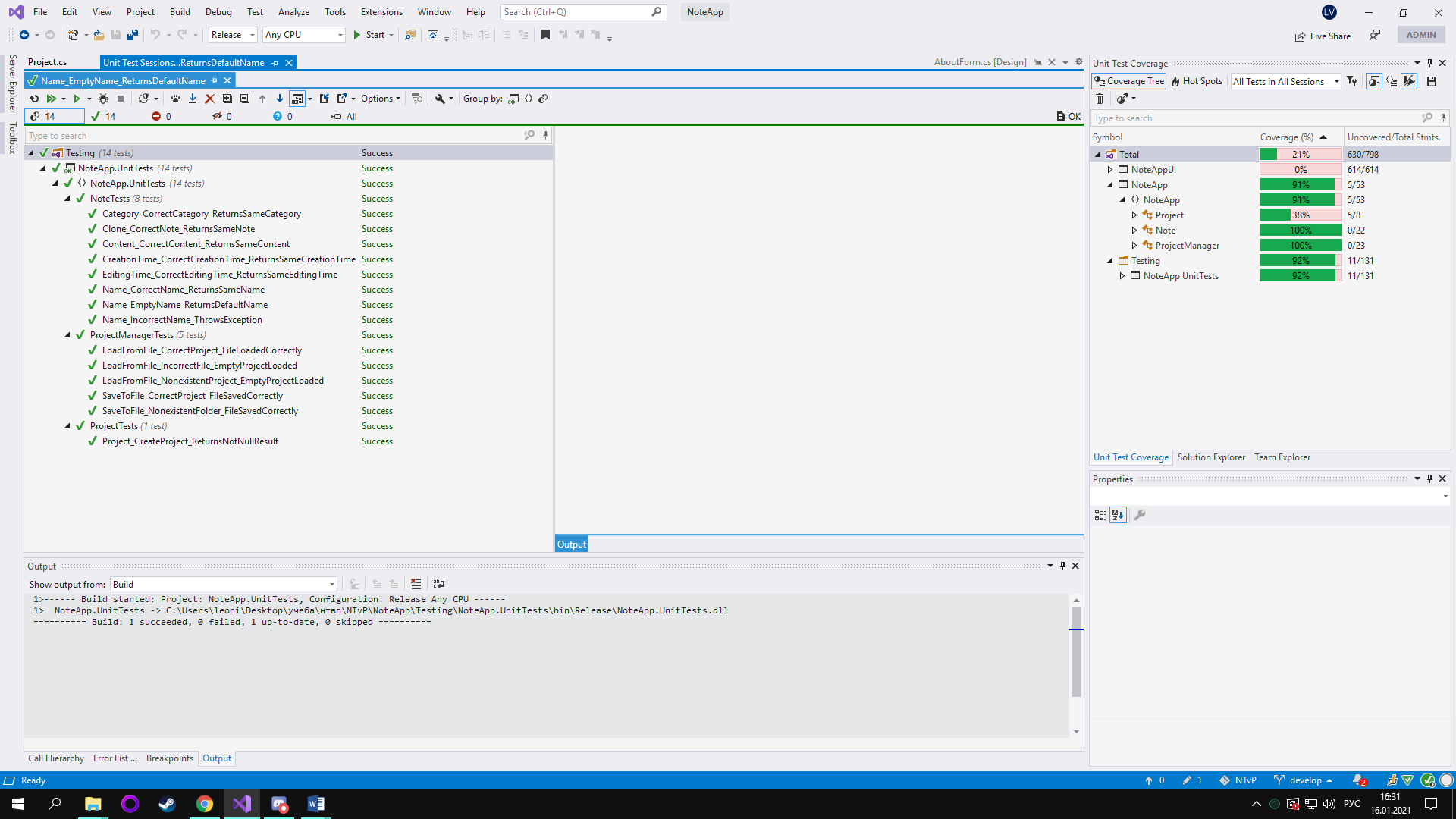


Рисунок 7.2 – Покрытие кода юнит-тестов

Кроме юнит-тестирования также проводилось приемочное тестирование. Для успешного прохождения приёмочного тестирования, необходимо выполнение следующей последовательности действий:

1. Установите приложение на компьютер с помощью собранного установочного пакета.
2. Запустите приложение. Окно программы должно быть пустым – в приложении не должно быть заметок.
3. Создайте три заметки в приложении разных категорий.
4. Переключитесь между заметками, показав, что смена текущей заметки происходит корректно.
5. Переключите отображаемую категорию заметок – в списке заметок должны остаться только заметки целевой категории. Снова отобразите все категории заметок – список заметок должен восстановиться.
6. Выберите вторую заметку и нажмите кнопку редактирования. Должно открыться окно редактирования заметки.
7. Введите название заметки более 50 символов. Элемент управления с названием заметки должен указать на некорректное значение.
8. Введите название заметки менее 50 символов. Элемент управления с названием должен стать корректным.
9. Поменяйте текст заметки. Нажмите «OK». Отредактированная заметка должна подняться в списке заметок на первую позицию, время изменения заметки должно поменяться, отображаемый текст заметки также должен измениться.
10. Выберите вторую заметку и нажмите кнопку редактирования. Должно открыться окно редактирования. Измените название заметки, её текст и категорию. Нажмите «Cancel». Исходная заметка должна остаться без изменений.
11. Удалите третью заметку.
12. Закройте приложение. Должно произойти сохранение заметок в целевой файл.
13. Запустите приложение. В программе должны восстановиться заметки, созданные в предыдущую сессию.
14. Дайте руководителю провести исследовательское тестирование программы.

# 8 Описание сборки установщика

Сборка установочного пакета осуществляется в автоматическом режиме, для этого необходимо скомпилировать класс Installer. Для создания сценария сборки, использовалась программа InnoSetup.

Исходный код сценария для автоматизации сборки представлен ниже. В нем указывается название установочного пакета, название программы, имя автора, пути установки и др. Необходимо указать файлы, требуемые для работы приложения: \*exe и \*dll.

В проекте NoteAppUI

Pre-build:

rd /s /q "$(SolutionDir)Installer\InstallScripts\Release"

rd /s /q "$(SolutionDir)Installer\InstallScripts\Installers"

Post-build:

md "$(SolutionDir)Installer\InstallScripts"

md "$(SolutionDir)Installer\InstallScripts\Release"

md "$(SolutionDir)Installer\InstallScripts\Installers"

xcopy "$(ProjectDir)$(OutDir)\*.dll" "$(SolutionDir)Installer\InstallScripts\Release\"

xcopy "$(ProjectDir)$(OutDir)\*.exe" "$(SolutionDir)Installer\InstallScripts\Release\"

ren "$(SolutionDir)Installer\InstallScripts\Release\NoteAppUI.exe" NoteApp.exe

В проекте Installer

Post-build:

"$(SolutionDir)packages\Tools.InnoSetup.6.1.2\tools\ISCC.exe"

"$(SolutionDir)Installer\InstallScripts\InstallScript.iss"

Код сценария установщика:

; Script generated by the Inno Setup Script Wizard.

; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES!

#define MyAppName "NoteApp"

#define MyAppVersion "1.0.0"

#define MyAppPublisher "Leonid Volkov"

#define MyAppExeName "NoteApp.exe"

[Setup]

; NOTE: The value of AppId uniquely identifies this application. Do not use the same AppId value in installers for other applications.

; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)

AppId={{3A605011-A436-4201-A3B3-526CA689A8B1}

AppName={#MyAppName}

AppVersion={#MyAppVersion}

;AppVerName={#MyAppName} {#MyAppVersion}

AppPublisher={#MyAppPublisher}

DefaultDirName={autopf}\{#MyAppName}

DisableProgramGroupPage=yes

; Uncomment the following line to run in non administrative install mode (install for current user only.)

;PrivilegesRequired=lowest

OutputBaseFilename=NoteAppSetup

OutputDir="Installers"

Compression=lzma

SolidCompression=yes

WizardStyle=modern

[Languages]

Name: "english"; MessagesFile: "compiler:Default.isl"

[Tasks]

Name: "desktopicon"; Description: "{cm:CreateDesktopIcon}"; GroupDescription: "{cm:AdditionalIcons}"; Flags: unchecked

[Files]

Source: "Release\\*.exe"; DestDir: "{app}"; Flags: ignoreversion

Source: "Release\\*.dll"; DestDir: "{app}"; Flags: ignoreversion

; NOTE: Don't use "Flags: ignoreversion" on any shared system files

[Icons]

Name: "{autoprograms}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"

Name: "{autodesktop}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"; Tasks: desktopicon

[Run]

Filename: "{app}\{#MyAppExeName}"; Description: "{cm:LaunchProgram,{#StringChange(MyAppName, '&', '&&')}}"; Flags: nowait postinstall skipifsilent

# 9 Описание модели ветвления в репозитории проекта

При работе над данным проектом, была использована система версионного контроля Git и сервис GitHub. Ссылка на репозиторий - https://github.com/ezcnrmn/NTvP.

Для разработки было сделано в репозитории две ветки: master – главная ветвь, код которой проверен, протестирован и готовый к сборке, а также developer – ветвь для разработки.

На рисунке 9.1 показана история фиксаций ветки develop.

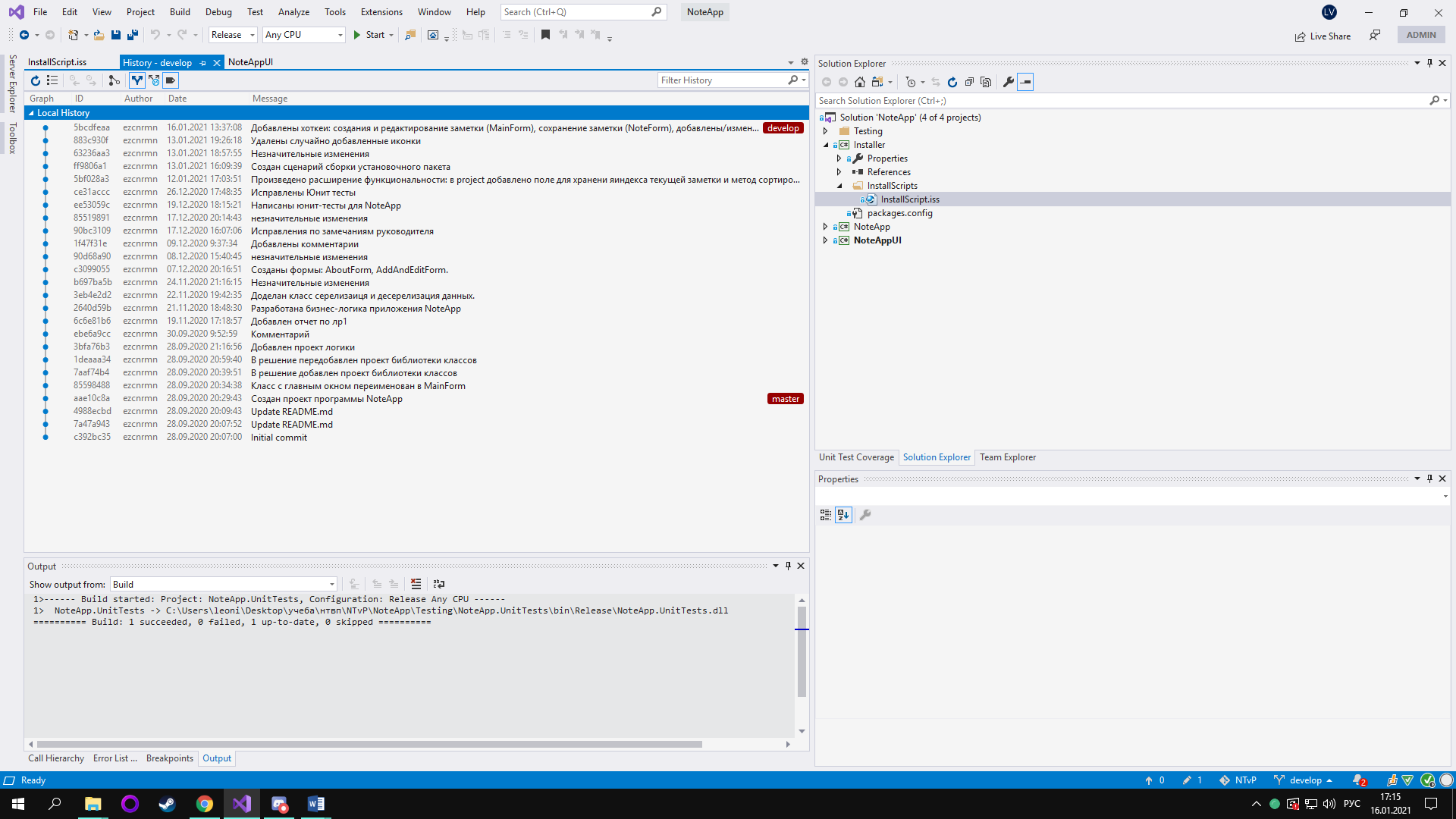


Рисунок 9.1 – История фиксаций ветки develop