

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 ЗАПИСНАЯ КНИЖКА «NOTEAPP»

Пояснительная записка к лабораторной работе по дисциплине
«Новые технологии в программировании» (НТВП)

Выполнил:

Студент гр. 588-1

_____ К.В. Рыжнев

«__» _____ 2021 г.

Проверил:

к.т.н., доцент каф. КСУП

_____ А. Е. Горяинов

«__» _____ 2021 г.

Содержание

1 Назначение приложения.....	3
2 Группы и роли пользователей.....	4
3 Стек технологий и системные требования	5
4 Пользовательский интерфейс.....	6
5 Диаграмма пакетов.....	11
6 Диаграмма классов	12
7 Тестирование приложения	13
8 Сборка установщика	15
9 Описание модели ветвления	18

1 Назначение приложения

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

Приложение должно иметь следующую функциональность:

1. Обеспечивать стабильную работу приложения при 200 заметок.
2. Обеспечивать сортировку по категории, а также навигацию по созданным заметкам.
3. Обеспечивать возможность создания и редактирования заметок.
4. Сохранять и восстанавливать заметки после открытия и закрытия приложения.
5. Выполнять промежуточные сохранения заметок на жесткий диск, чтобы предотвратить потерю данных при аварийном завершении приложения, т.е. для защиты от потери данных.

Приложение референс: десктоп-версия программы Evernote.

2 Группы и роли пользователей

Роль пользователя в приложении единственная. Пользователь имеет следующий набор функционала:

- 1) создание заметки;
- 2) просмотр заметок;
- 3) редактирование заметки;
- 4) удаление заметок;
- 5) просмотр информации о программе.

Помимо этого, пользователь может отфильтровать созданные заметки по одной из установленных категорий: «Все», «Работа», «Дом», «Здоровье и Спорт», «Люди», «Документы», «Финансы», «Разное».

3 Стек технологий и системные требования

Стек технологий:

- Язык программирования: C#;
- Среда разработки: Visual Studio 2019 версия 16.8.0 с применением Microsoft.NET Framework 4.0.;
- Для реализации сериализация и десериализация используется библиотека Newtonsoft.Json 12.0.3;
- Тестирование с помощью библиотек: NUnit версия 2.6.4, NUnit3TestAdapter версия 2.6.4;

4 Пользовательский интерфейс

После запуска приложения перед пользователем появляется главное окно (рис. 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. Заметки в списке отображаются в порядке их последнего редактирования/добавления.

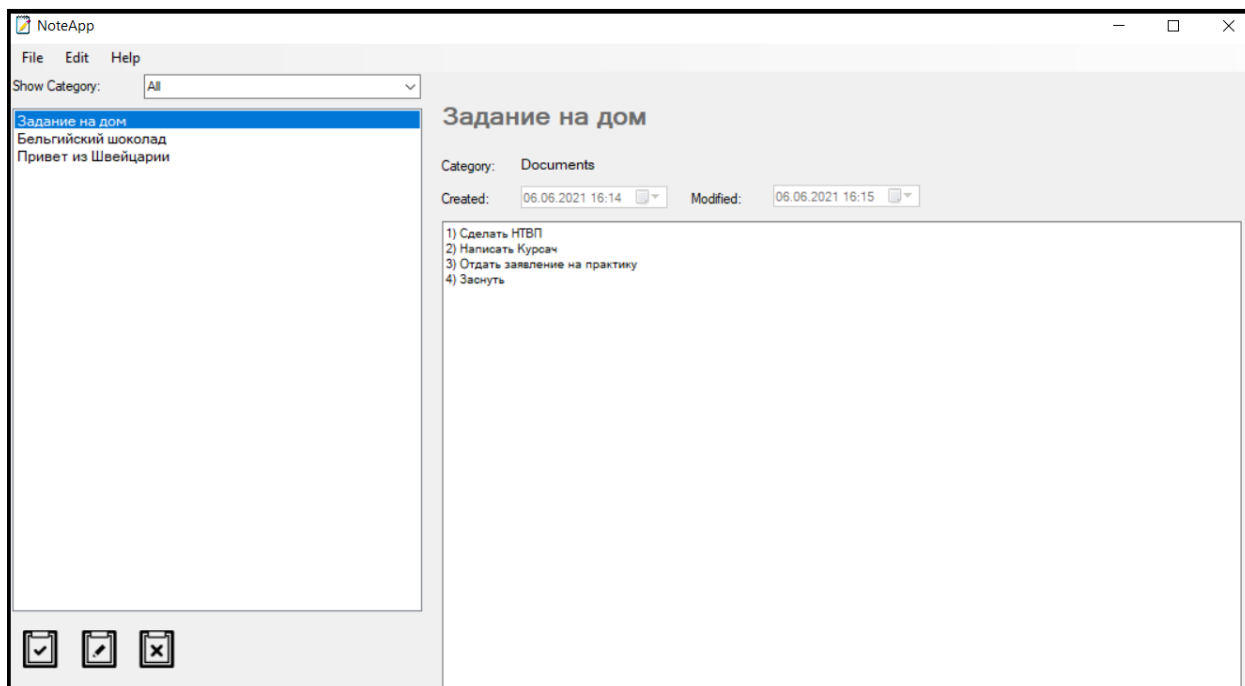


Рисунок 4.1 – Главное окно программы

На панели со списком заметок сверху расположен выпадающий список категорий заметок. При выборе пользователем определенной категории, в списке заметок должны отобразиться только заметки выбранной категории. При выборе пункта «All» отобразить все заметки.

На панели со списком заметок внизу располагаются три кнопки: 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 (рис. 4.2). Для новой заметки установлены только название заметки по умолчанию, дата создания и дата редактирования. Для редактирования уже существующей заметки все поля должны быть заполнены данными текущей заметки.

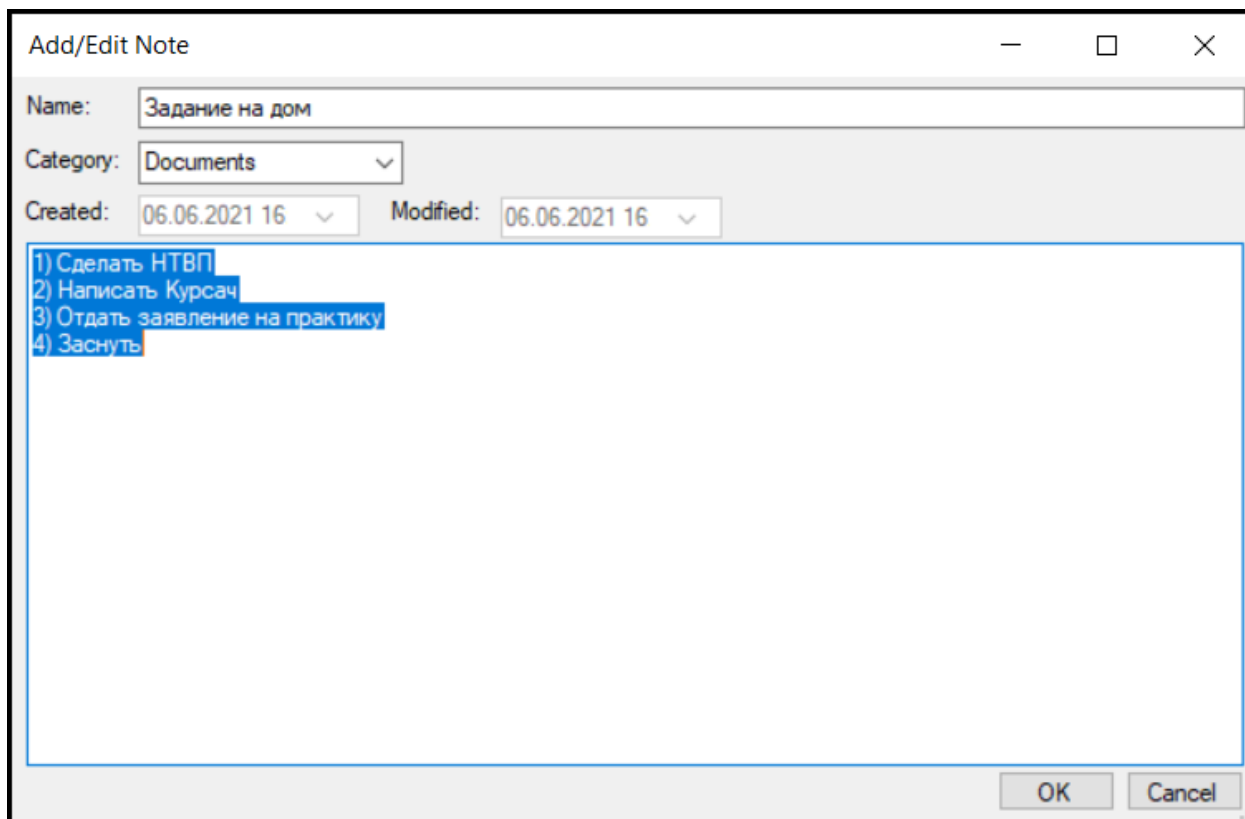


Рисунок 4.2 – Окно создания и редактирования заметки

При нажатии на кнопку ОК окно создания заметки закрывается, в список заметок главного окна добавляется новая заметка. При редактировании текущей заметки, нажатие на кнопку ОК должно обновить название заметки в списке заметок, и обновить отображаемую заметку в правой панели приложения. При нажатии кнопки Cancel создание/редактирование заметки отменяется и исходная заметка остается без изменений.

В случае ввода пользователем некорректных данных (нарушение допустимой длины названия заметки), поле окрашивается в красный цвет (Рисунок 4.3).

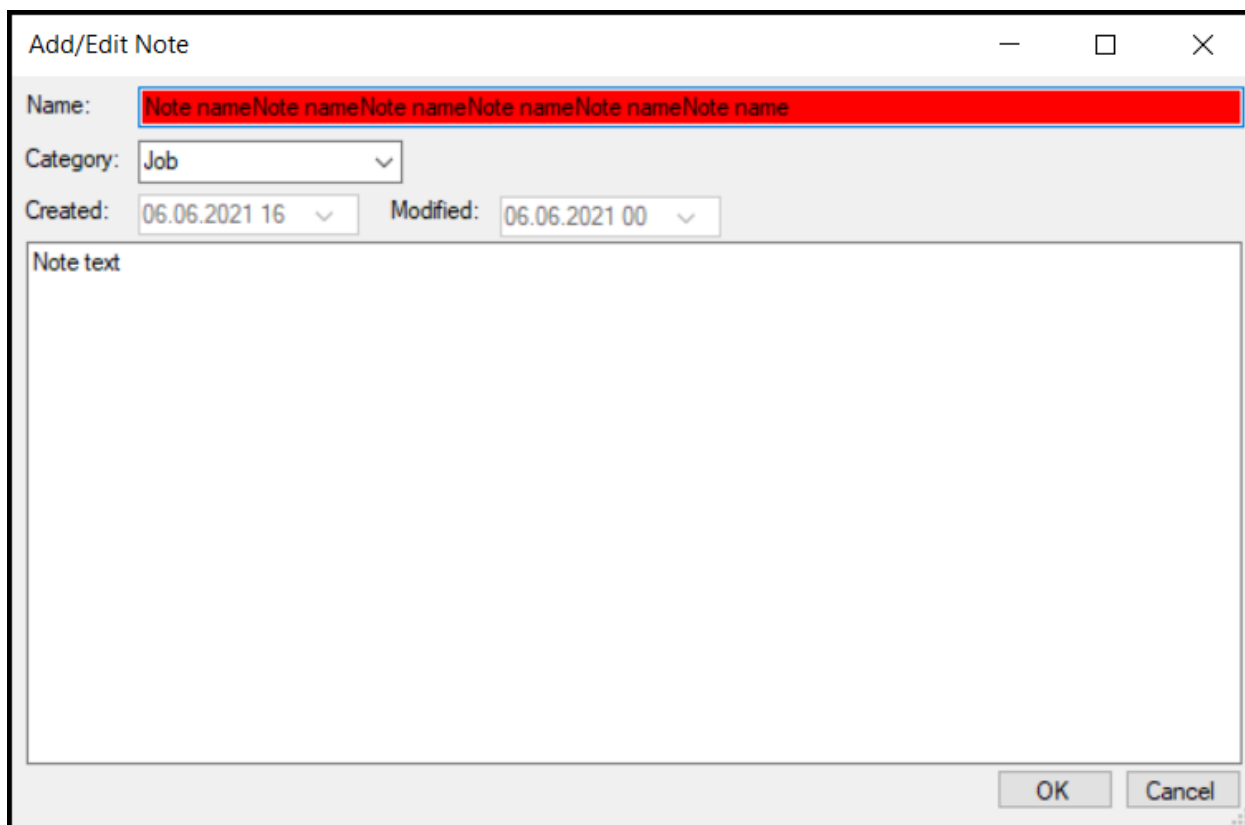


Рисунок 4.3 – Демонстрация превышения длины имени

Также имеется возможность удалить выбранную заметку, перед удалением появиться окно с уточняющим сообщением о желании удалить (рис. 4.4). При нажатии на кнопку ОК происходит удаление, при нажатии на кнопку Cancel удаление отменяется. Также удаление можно произвести при нажатии кнопки “Delete” на клавиатуре.

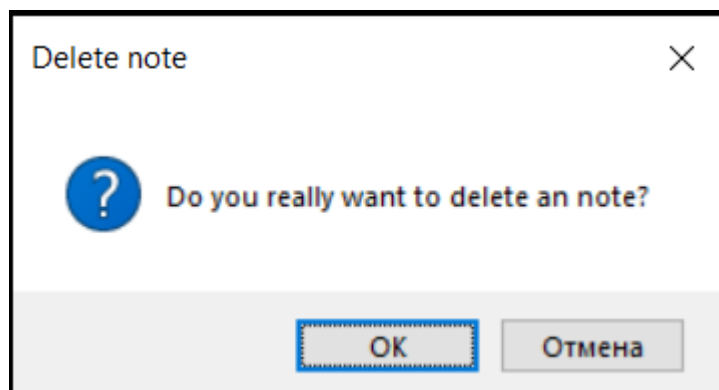


Рисунок 4.4 – Сообщение об удалении заметки

Так же пользователь может посмотреть окно «О программе» (рисунок 4.5).

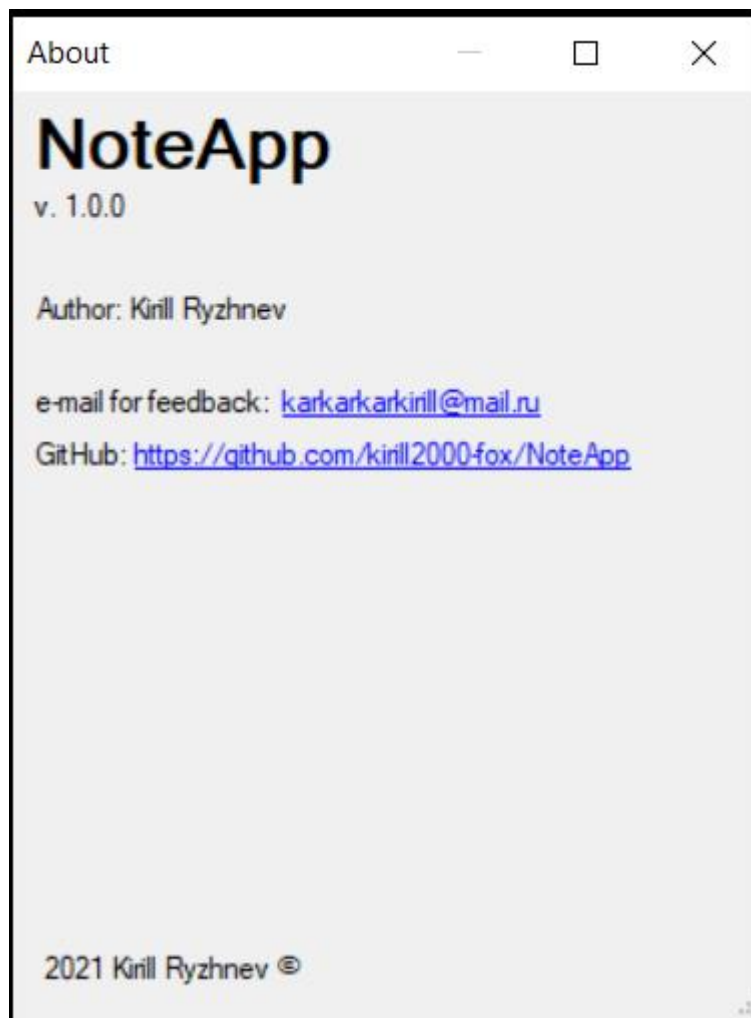


Рисунок 4.5 – окно «О программе»

Таким образом, в программе реализовано три окна:

- главное окно;
- окно создания/редактирования заметки;
- окно «About».

Верстка главного окна и окна создания/редактирования заметки адаптивны. Окно «About» имеет фиксированный размер.

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

При закрытии приложения запоминается последняя просматриваемая заметка и выбранная категория, что позволяет при повторном открытии

приложения быть в той же категории и видеть последнюю просматриваемую заметку.

Меню главного окна содержит следующие пункты:

- *File*
 - *Exit* (Выйти из приложения)
- *Edit*
 - *Add Note* (Создать новую заметку)
 - *Edit Note* (Редактировать текущую заметку)
 - *Remove Note* (Удалить текущую заметку)
- *Help*
 - *About* (Вызвать окно “О программе”)

5 Диаграмма пакетов

Диаграмма пакетов приложения представлена на рисунке 5.1.

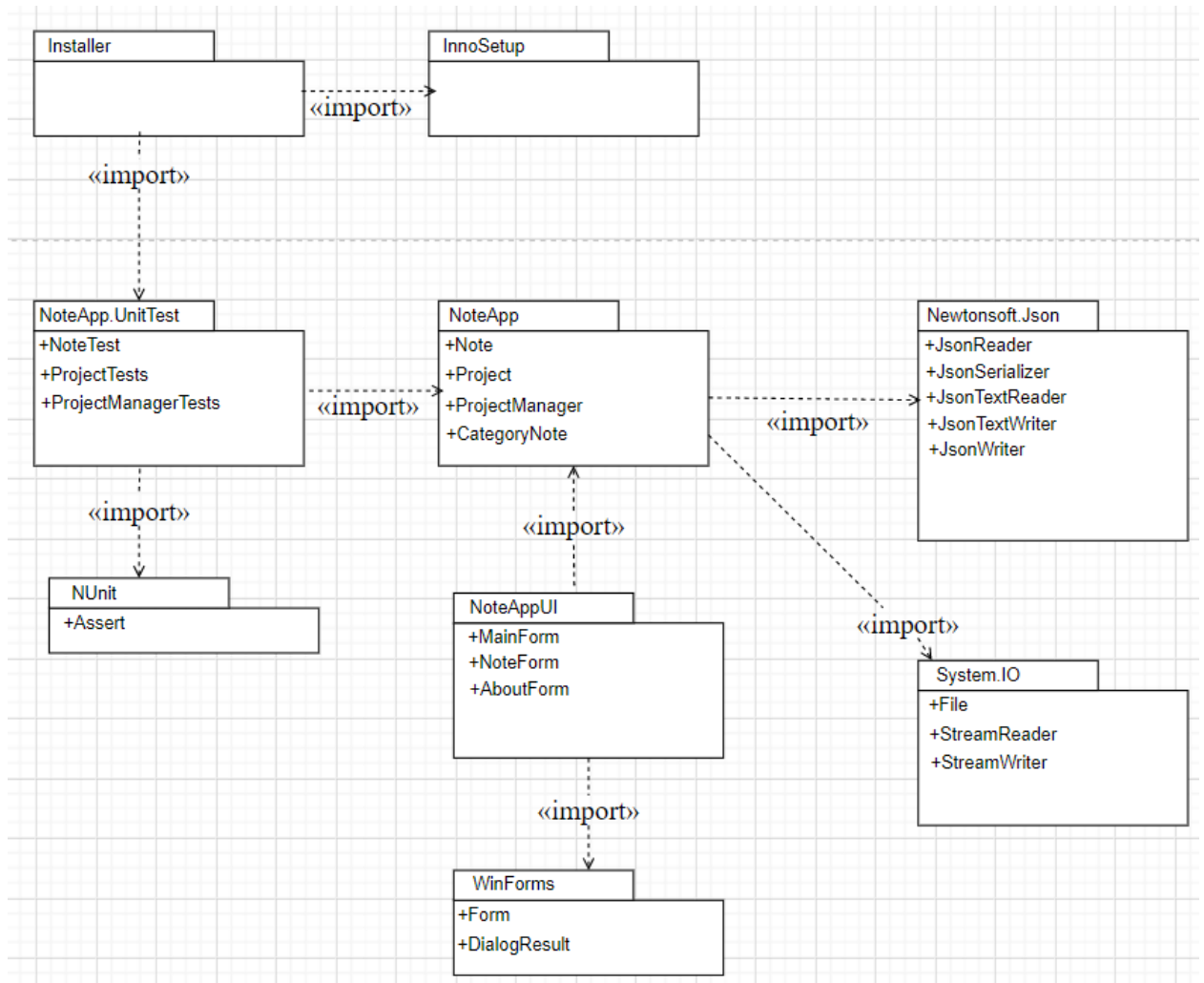


Рисунок 5.1 – Диаграмма пакетов приложения

6 Диаграмма классов

Диаграмма классов приложения представлена на рисунке 6.1.

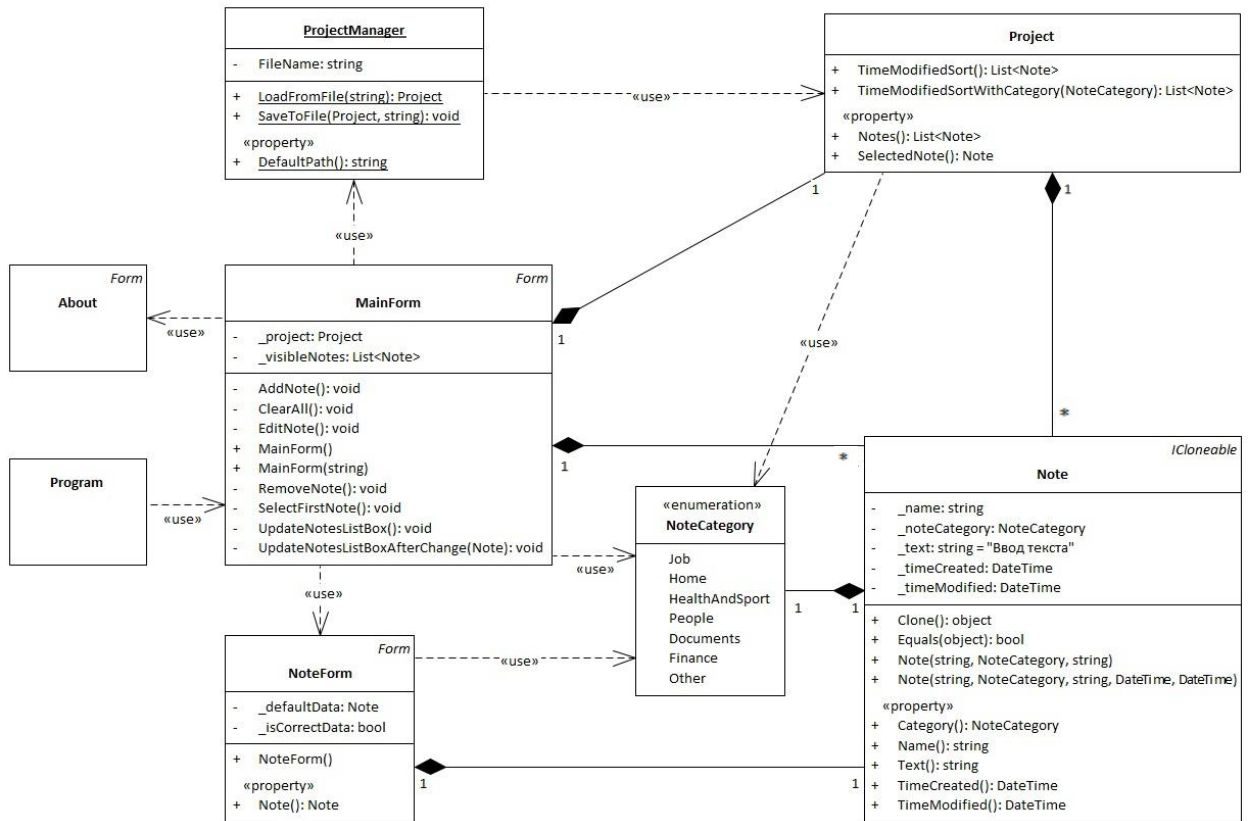
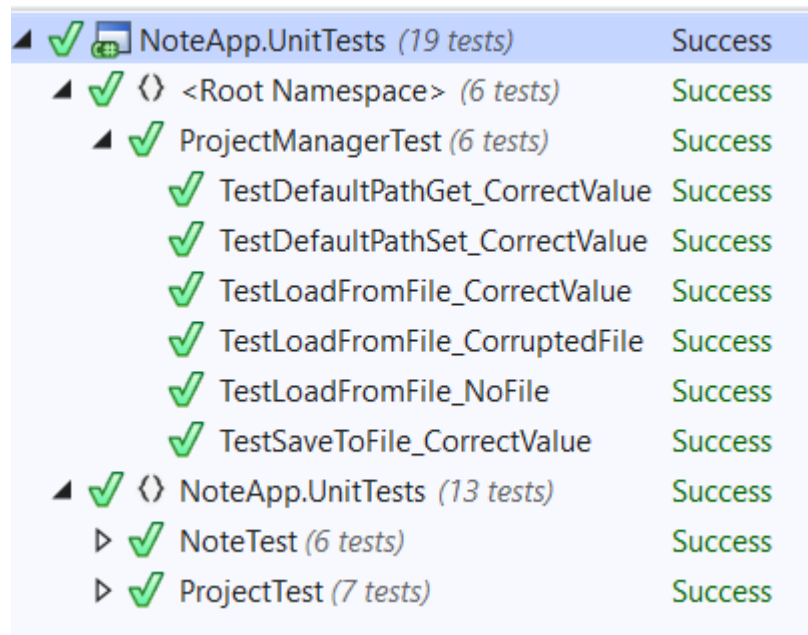


Рисунок 6.1 – Диаграмма классов приложения

7 Тестирование приложения

Тестирование приложения производилось как вручную, так и при помощи юнит-тестов.

Для написания юнит-тестов использовалась библиотека NUnit. Тесты проверяют как на позитивные, так и на негативные варианты работы классов. Список юнит-тестов представлен на рисунке 7.1.




▲ ✓  NoteApp.UnitTests (19 tests)	Success
▲ ✓ {} <Root Namespace> (6 tests)	Success
▲ ✓ ProjectManagerTest (6 tests)	Success
✓ TestDefaultPathGet_CorrectValue	Success
✓ TestDefaultPathSet_CorrectValue	Success
✓ TestLoadFromFile_CorrectValue	Success
✓ TestLoadFromFile_CorruptedFile	Success
✓ TestLoadFromFile_NoFile	Success
✓ TestSaveToFile_CorrectValue	Success
▲ ✓ {} NoteApp.UnitTests (13 tests)	Success
▷ ✓ NoteTest (6 tests)	Success
▷ ✓ ProjectTest (7 tests)	Success

Рисунок 7.1 - Список юнит-тестов.

Классы пользовательского интерфейса тестировались вручную.

В качестве общего системного теста было проведено приемочное тестирование, проверяющее работоспособность всего приложения. Шаги приемочного тестирования:

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 три заметки в приложении разных категорий.
4. Переключиться между заметками, показав, что смена текущей заметки происходит корректно.

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

6. Выбрать вторую заметку и нажмите 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 должен стать корректным.

9. Поменять текст заметки. Нажмите «ОК». 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 нажмите кнопку редактирования. Должно открыться окно редактирования. Изменить название заметки, её 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.

14. Дать руководителю провести исследовательское тестирование программы.

8 Сборка установщика

Для создания сценария сборки инсталлятора, использовалась программа InnoSetup. Программа InnoSetup позволяет создавать сценарии для сборки установочных пакетов и компилировать установщики. Сценарии сохраняются в формате *.iss (inno setup script). Далее представлен сценарий сборки установочного пакета.

```
; Script generated by the Inno Setup Script Wizard.
```

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

```
#define MyAppName "NoteApp"
```

```
#define MyAppVersion "1.5"
```

```
#define MyAppPublisher "Kirill Ryzhnev"
```

```
#define MyAppExeName "NoteAppUI.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={{20D3D2E0-8395-4DBF-B4AD-6E06201FC0C2}}
```

```
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
```

```
OutputDir=Installers
```

```
OutputBaseFilename=setup
```

```
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\NoteAppUI.exe" ; DestDir: "{app}"; Flags: ignoreversion

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

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

Source: "Notepad_22522.ico"; DestDir: "{app}"; Flags: ignoreversion

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

[Icons]

Name: "{autoprogams}\{#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

Рисунок 8.1 – скриншот сценария сборки установочного пакета

В установочный пакет помещаются файлы, необходимые для работы приложения. Также помещаться иконки, вспомогательные файлы примеров и т.д. – любые файлы, которые необходимы для работы приложения у пользователя. В установочный пакет не помещаются файлы исходного кода – только скомпилированное приложение.

Для автоматизации сборки установочного пакета используется запуск сценария сборки установщика в среде Visual Studio. Для этого сделаны дополнительные настройки в решении. Добавлены команды, выполняющиеся после каждой успешной компиляции.


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

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

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

```
xcopy "$(ProjectDir)$(OutDir)*.exe"
```

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

```
xcopy "$(ProjectDir)$(OutDir)*.dll" "
```

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

где:

1. «rmdir» - удаление директории;
2. «md» - создание директории;
3. «xcopy» - копирование из одной директории в другую.

В результате выполнения сценария сборки установочного пакета был получен установочный файл “setup.exe”, в результате выполнения которого происходит установка приложения “NoteApp” на машину пользователя.

9 Описание модели ветвления

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

GitHub.com – это веб-сервис, позволяющий хранить промежуточные версии вашего исходного кода. Таким образом, в случае написания непоправимых ошибок в исходном коде или его потери, сервис позволит восстановить проект. Также является обязательным инструментом для организации командной работы, используемым во многих IT-компаниях. Предоставляет бесплатную лицензию для малых проектов.

Ссылка на репозиторий: <https://github.com/kirill2000-fox/NoteApp.git>.

Для разработки использовались две ветки: master – главная ветвь, содержащая стабильную и отлаженную версию программы, а также develop – основная ветвь для разработки и тестирования нового функционала. Ветка develop создается из ветки master сразу же при создании репозитория и существует всё время разработки проекта.