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 «Записная книжка NoteApp»

Выполнил:

студент гр. 586-1

\_\_\_\_\_\_\_\_\_С.Д. Дубинин

«\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

Проверил:

доцент каф. КСУП

\_\_\_\_\_\_\_\_\_А.Е. Горяинов

«\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

Томск 2019

Содержание

[1 Назначение приложения 3](#_Toc27695646)

[2 Группы пользователей и их функциональные возможности в приложении 3](#_Toc27695647)

[3 Стек технологий разработки. Системные требования 4](#_Toc27695648)

[3.1 Стек технологий разработки 4](#_Toc27695649)

[3.2 Системные требования 4](#_Toc27695650)

[4 Пользовательский интерфейс 4](#_Toc27695651)

[5 Диаграмма пакетов приложения 8](#_Toc27695652)

[6 Диаграмма классов приложения 9](#_Toc27695653)

[7 Описание тестирования приложения 9](#_Toc27695654)

[8 Описание сборки установщика 11](#_Toc27695655)

[9 Описание модели ветвления 11](#_Toc27695656)

1 Назначение 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.

2 Группы пользователей и их функциональные возможности в приложении

Данное приложение предназначено для использования неограниченной целевой аудиторией в личных целях.

Приложение предоставляет следующие функциональные возможности:

* добавление текстовой заметки, при добавлении необходимо ввести название и текст заметки, выбрать из предложенного списка категорию заметки, дата создания заметки добавится автоматически;
* редактирование текстовой заметки с автоматической фиксацией времени последнего изменения заметки;
* удаление текстовой заметки;
* просмотр текстовой заметки;
* сортировка заметок по определенной выбранной категории;
* запоминание последней просмотренной пользователем заметки при закрытии приложения.

3 Стек технологий разработки. Системные требования

3.1 Стек технологий разработки

Язык программирования: C#, используемая версия .Net Framework: 4.7.03056.

Среда разработки: Microsoft Visual Studio Community 2019 версия 16.2.5.

Библиотека, используемая для сериализации и десериализации: Newtonsoft.Json 12.0.3.

Библиотеки, используемые для модульного тестирования:

* NUnit версия 3.12.0;
* NUnit3TestAdapter версия 3.15.1.

3.2 Системные требования

Операционная система: Microsoft Windows 10 (x64).

Процессор: Intel Core i3-7020U CPU 2.30 GHz.

Оперативная память: 6 GB.

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 (далее – текущая заметка).

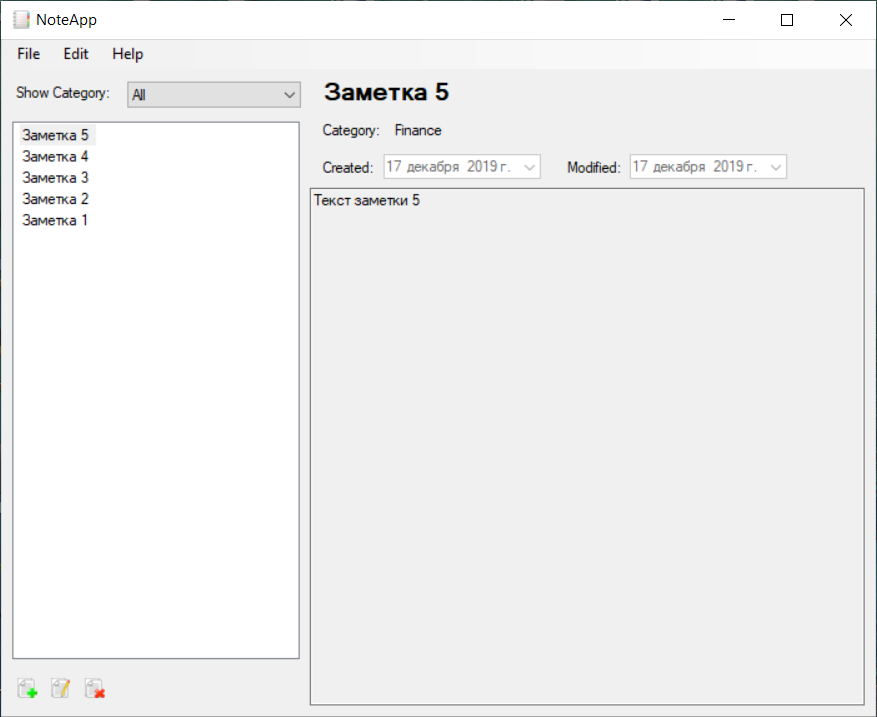


Рисунок 4.1 – Главное окно приложения NoteApp

В левой панели главного окна над списком заметок расположен выпадающий список категорий заметок. При выборе пользователем определенной категории, в списке заметок отображаются только заметки выбранной категории. При выборе пункта «All» отображаются все заметки.

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 (рис. 4.2). 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 предзаполнены данными текущей заметки.

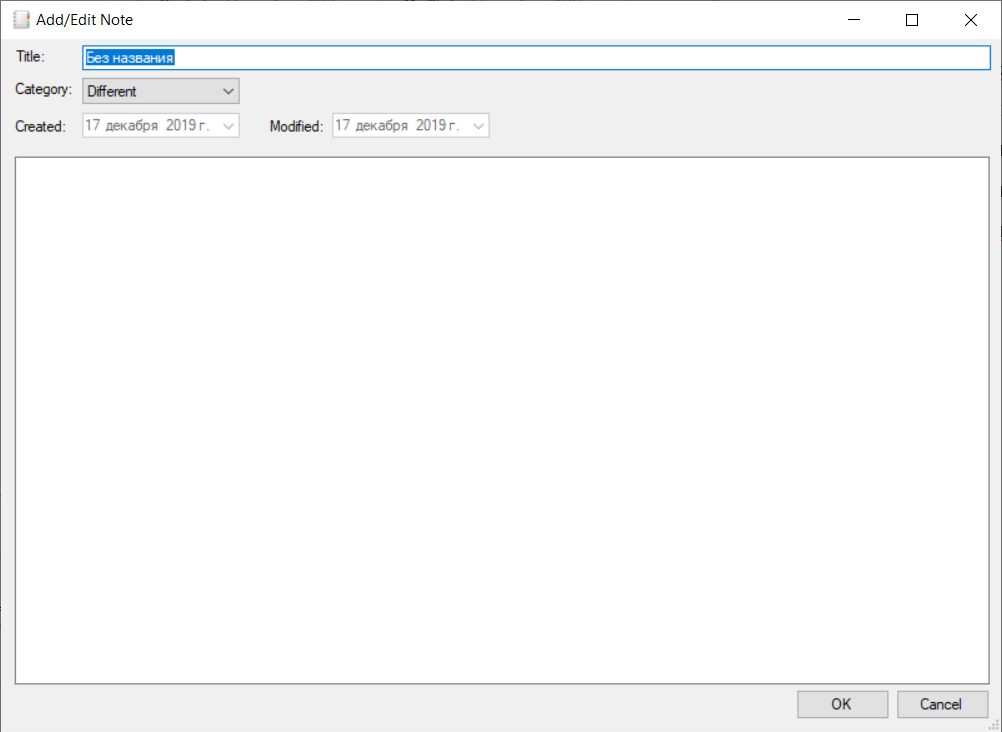


Рисунок 4.2 – Окно добавления/редактирования заметки приложения NoteApp

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 обновляет название заметки в списке заметок (если название текущей заметки было изменено), и обновляет 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), поле окрашивается красным цветом и при наведении курсора мыши на данное поле отображается всплывающая подсказка о некорректности введенных данных.

При нажатии на кнопку Remove Note главного окна, либо на кнопку Delete на клавиатуре текущая запись удаляется. Перед удалением появляется окно с запросом на разрешение записи: «Do you really want to remove this note: <Название текущей заметки>?». 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)
* Edit:
  + Add Note (Создать новую заметку)
  + Edit Note (Редактировать текущую заметку)
  + Remove Note (Удалить текущую заметку)
* Help:
  + About (Вызвать окно «О программе») (рис. 4.3)

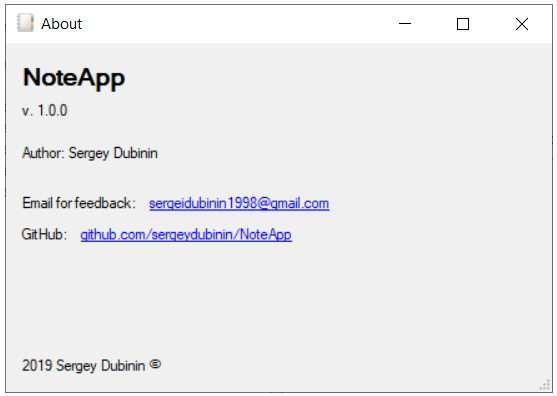


Рисунок 4.3 – Окно «About» приложения NoteApp

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.

5 Диаграмма пакетов приложения

На рисунке 5.1 представлена диаграмма пакетов приложения NoteApp.

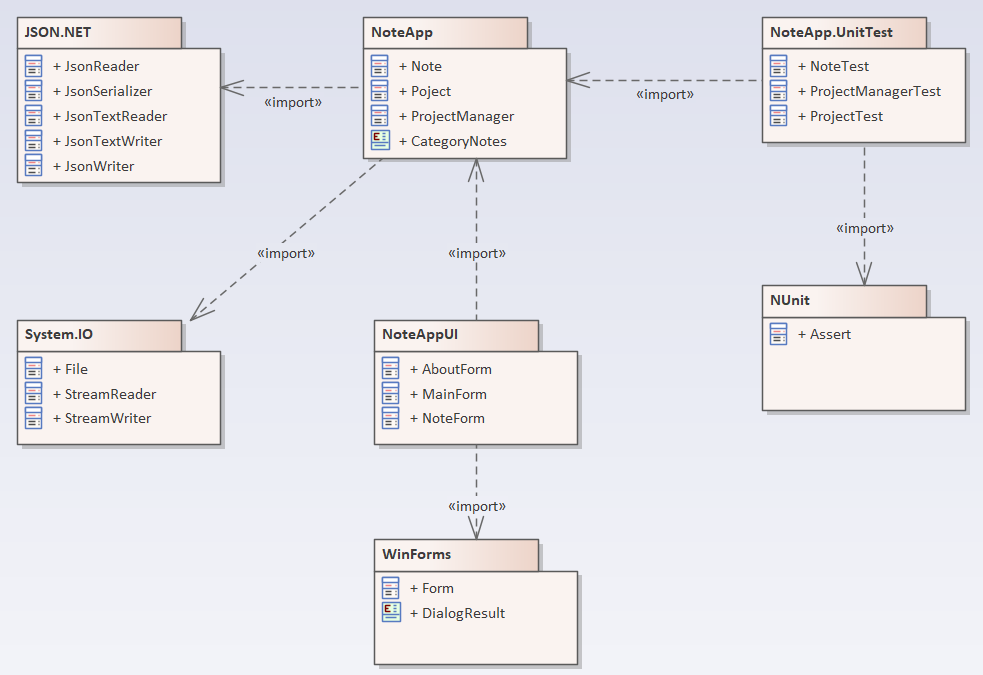
****

Рисунок 5.1 **–** Диаграмма пакетов приложения NoteApp

Пакет NoteApp содержит в себе классы бизнес-логики программы. Пакет NoteAppUI хранит классы пользовательского интерфейса. Пакет NoteApp.UnitTest содержит автоматизированные тесты для бизнес-логики программы и использует стороннюю библиотеку NUnit, предназначенную для упрощения написания юнит-тестов. Пакет WinForms содержит готовые компоненты для создания пользовательского интерфейса. Пакет JSON.NET используется для сохранения/загрузки данных программы на жестком диске. Пакет System.IO содержит типы, позволяющие осуществлять чтение и запись в файлы и потоки данных, а также типы для базовой поддержки файлов и папок.

6 Диаграмма классов приложения

На рисунке 6.1 приведена диаграмма классов приложения NoteApp.

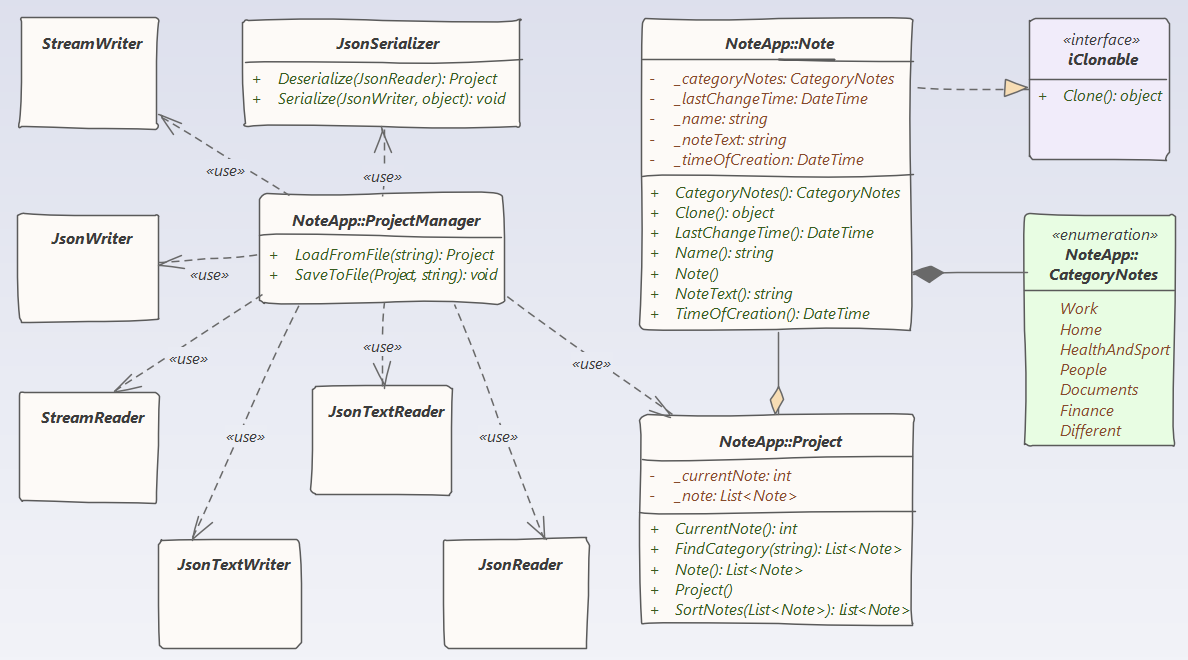
****

Рисунок 6.1 – Диаграмма классовприложения NoteApp

Класс Note хранит данные о заметке и реализует интерфейс IClonable для клонирования заметки. Класс Project содержит список всех заметок и индекс текущей заметки, а также методы для сортировки заметок по дате последнего изменения и по категориям. Класс ProjectManager состоит из методов для сохранения/загрузки списка заметок с жесткого диска, данная возможность реализуется с помощью классов SteramWriter, JsonWriter, SteramReader, JsonTextWriter, JsonTextReader, JsonReader.

7 Описание тестирования приложения

При разработке приложения производилось юнит-тестирование бизнес-логики программы с помощью автоматизированных тестов со 100% покрытием кода для этого использовалась сторонняя библиотека NUnit. Всего было написано 17 юнит-тестов. Функциональность и работоспособность приложения тестировалась вручную.

Также производилось приемочное тестирование по следующему алгоритму:

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 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 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, 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.

8 Описание сборки установщика

Для создания сценария сборки установочного пакета и компиляции установщика использовалась программа InnoSetup. С помощью средств Visual Studio была автоматизирована сборка установщика путем автоматического запуска сценария InnoSetup. В процессе сборки установщика обфускация кода не используется, функции защиты и лицензирования не внедряются.

9 Описание модели ветвления

При разработке приложения использовалась система версионного контроля Git. Ссылка на репозиторий: https://github.com/sergeydubinin/NoteApp

В репозитории были созданы следующие ветки:

master – главная и начальная ветка репозитория;

develop – основная ветка разработки.