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 КОНТАКТЫ «CONTACTSAPP»

Пояснительная записка к программе, разработанной по

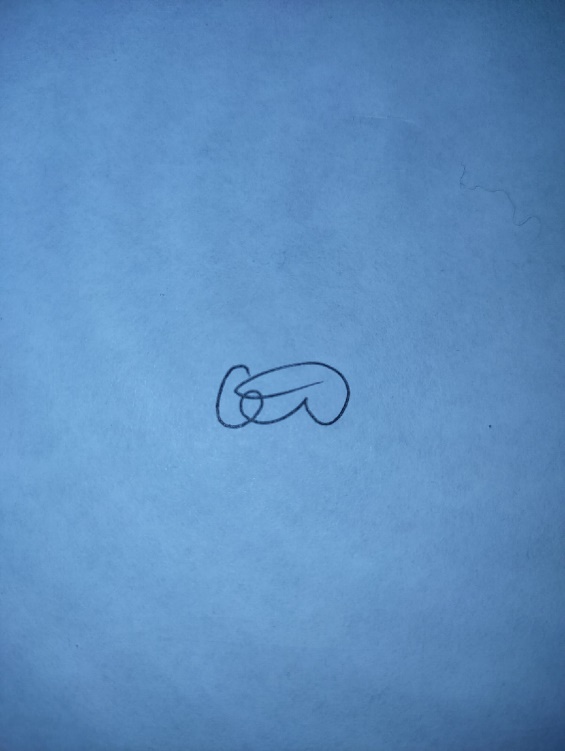
дисциплине «Системное программное обеспечение»

Проверил:

к.т.н., доцент каф. КСУП

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

Выполнил:

****Студент группы 539-1

\_\_\_\_\_\_\_ Сердюк И.В.

25.01.2022г.

Томск 2022

**Оглавление**

[1 Назначение приложения 3](#_Toc93878077)

[2 Группы пользователей и их функциональные возможности в приложении 4](#_Toc93878078)

[3 Стек технологий разработки 5](#_Toc93878079)

[4 Пользовательский интерфейс 6](#_Toc93878080)

[5 Диаграмма пакетов 9](#_Toc93878081)

[6 Диаграмма классов 10](#_Toc93878082)

[7 Описание тестирования приложения 11](#_Toc93878083)

[8 Сборка установщика 13](#_Toc93878084)

[9 Описание модели ветвления 14](#_Toc93878085)

[Список литературы 16](#_Toc93878086)

**1 Назначение приложения**

Пользовательское приложение ContactsApp, предназначено для ведения и хранения контактов. Приложение должно:

- Обеспечивать стабильную работу приложения при порядке 200 контактов.

- Обеспечивать поиск, навигацию по созданным контактам по фамилии и имени.

- Предоставить инструменты для просмотра и редактирования контактов.

- Сохранять и восстанавливать контакты между сессиями приложения.

- Выполнять промежуточные сохранения контактов 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**

В приложении предусмотрена одна роль пользователя. Он имеет следующий набор функционала:

1) Просмотр контактов.

2) Создание, редактирование и удаление контактов.

3) Поиск, навигацию по созданным контактам по фамилии и имени.

**3 Стек технологий разработки**

Для проекта был выбран язык программирования C# .NET Framework 4.7.2 - эти пункты являются обязательными требованиями учебного курса, изучаемого предмета.

Для сериализации и десериализации контактов использовалась библиотека JSON.NET от компании Newtonsoft версии 13.0.1.

Для тестирования приложения использовалась библиотека NUnit версии 3.13.1, так как она обеспечивает достаточную гибкость описания тестов, требуемую для данного проекта.

**4 Пользовательский интерфейс**

После запуска приложения перед пользователем появляется главное окно, (рисунок 3.1). Двухколоночная верстка главного окна содержит список всех контактов в левой панели и отображает данные текущего выбранного контакта на правой панели. В списке контактов показаны фамилии контактов, в один момент времени может быть выбран только один контакт.

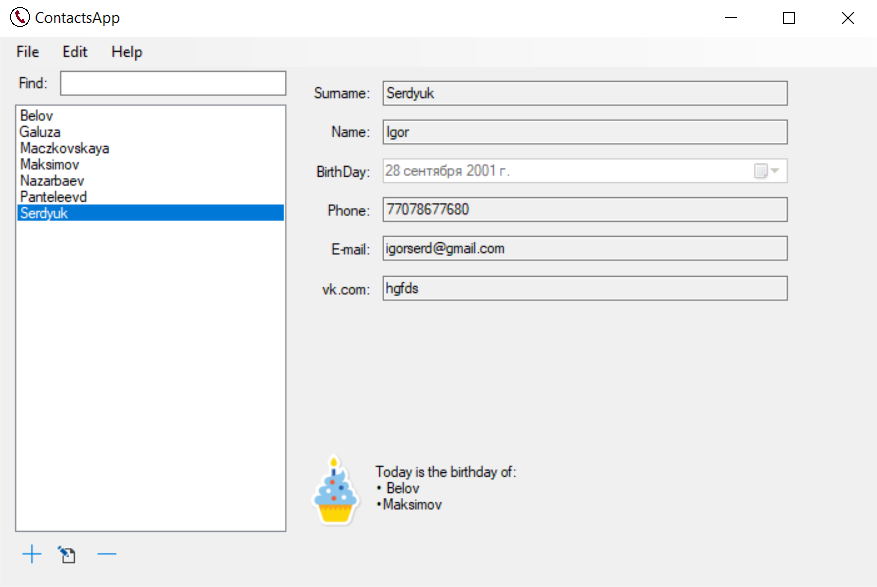


Рисунок 4.1 - Макет главного окна приложения ContactsApp

На панели со списком контактов внизу располагаются три кнопки в виде пиктограмм: Add Contact («Создать новый контакт»), Edit Contact («Редактировать текущий контакт»), Remove Contact («Удалить текущий контакт»).

При выборе контакта в списке, выбранный контакт отображается в правой панели. Главное окно не позволяет редактировать содержимое контакта – только просмотр. Также в правой панели под текущим контактом отображается информационная панель с сегодняшними именинниками (рисунок 3.1).

При нажатии на кнопку Add Contact и Edit Contact появляется окно создания/редактирования контакта в диалоговом режиме (рисунок 3.2). Для нового контакта окно изначально не заполнено (установлена лишь дата рождения по умолчанию). Для редактирования уже существующего контакта все поля должны быть предзаполнены данными текущего контакта.

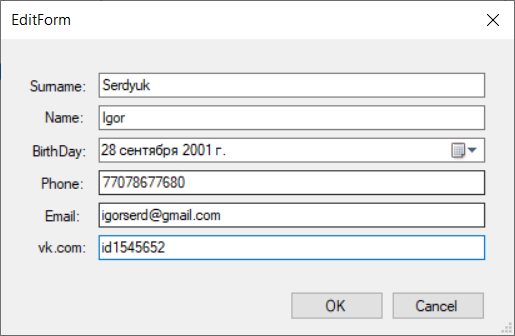


Рисунок 4.2 - Макет окна редактирования контакта в приложении ContactsApp

При нажатии на кнопку OK окно создания контакта закрывается, в список контактов главного окна добавляется новый контакт. При редактировании текущей контакта, нажатие на кнопку OK должно обновить фамилию контакта в списке контактов (если фамилия текущего контакта была изменена/исправлена), и обновить отображаемый контакт в правой панели приложения. При нажатии кнопки Cancel создание/редактирование контакта отменяется (новый контакт не добавляется, исходный контакт остается без изменений.

В случае ввода пользователем некорректных данных (нарушение допустимой длины фамилии, имени, указание невозможной даты рождения или неправильного номера телефона), данная ситуация должна быть обработана соответствующим образом.

При нажатии на кнопку Remove Contact главного окна текущий контакт удаляется. Перед удалением должно появиться окно с запросом на разрешение записи: «"Are you sure you want to delete (фамилия контакта) from your contact list?". 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) (рисунок 3.3)

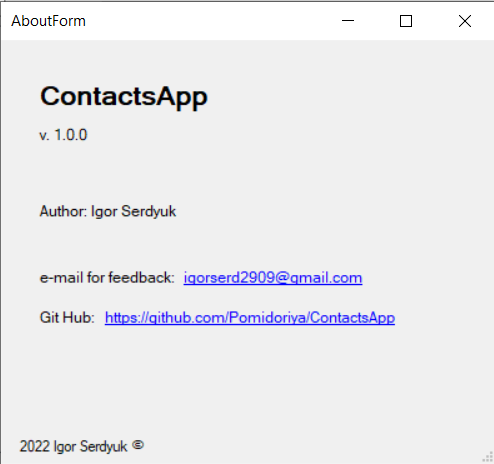
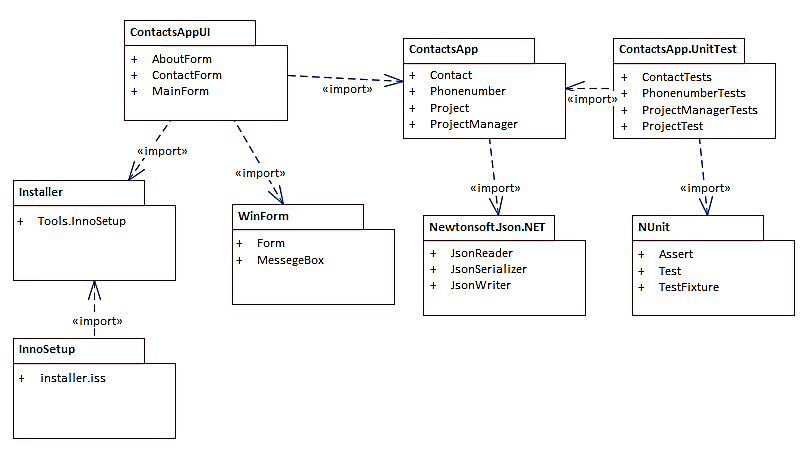
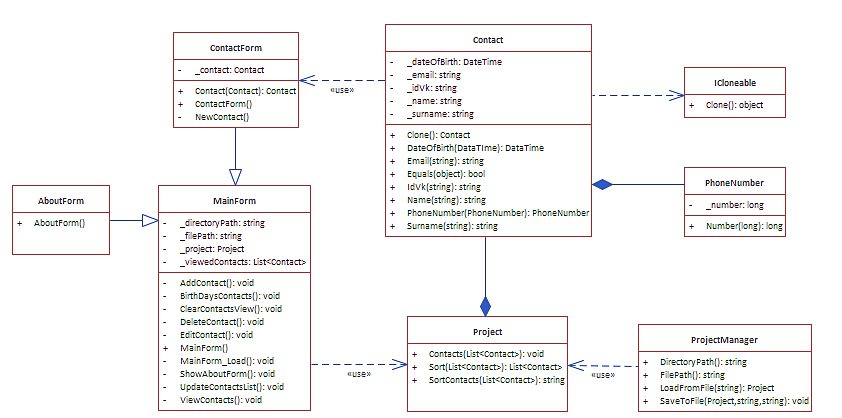


Рисунок 4.3 - Макет окна «About» приложения ContactsApp

**5 Диаграмма пакетов**



**6 Диаграмма классов**



**7 Описание тестирования приложения**

Для написания юнит-тестов использовалась библиотека NUnit.

Были написаны тесты, покрывающие все классы бизнес-логики приложения. Полный список юнит-тестов представлен на рисунке 7.1.

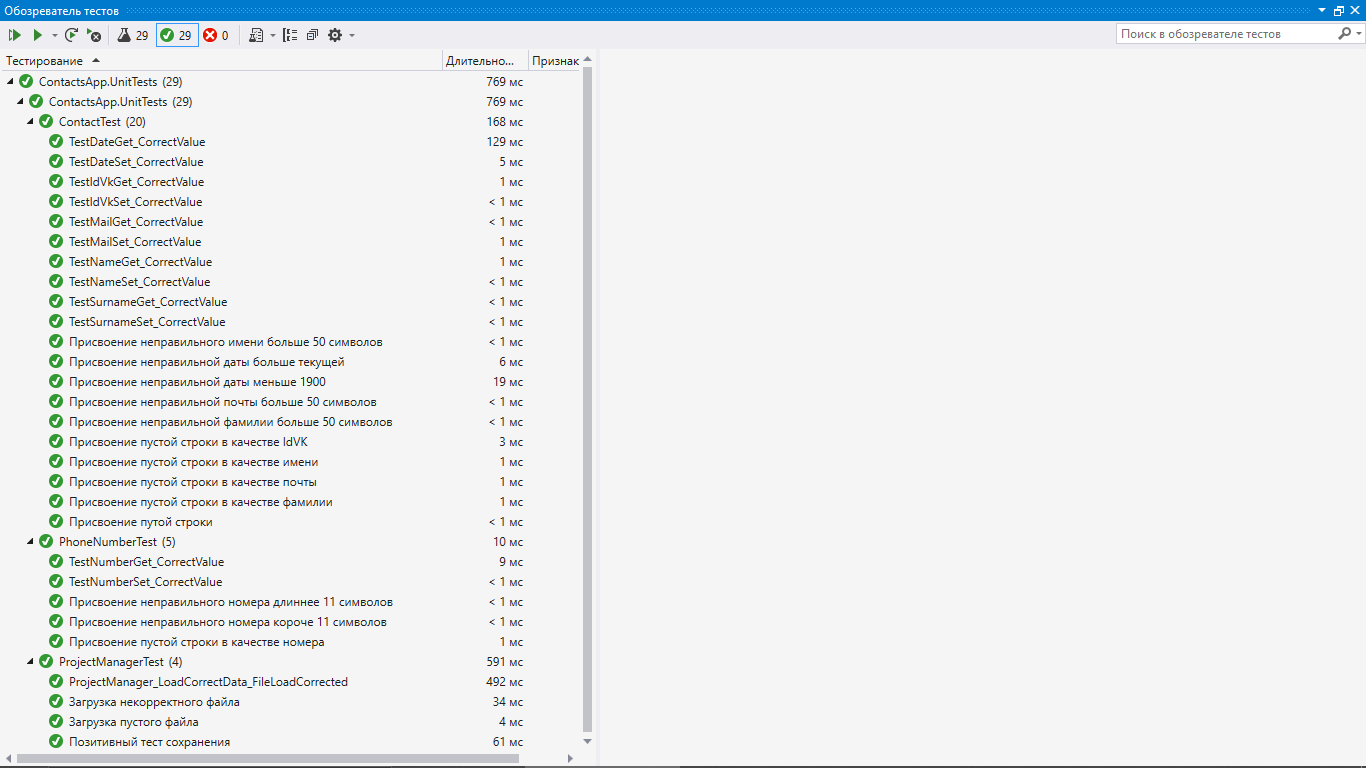


Рисунок 7.1 - Полный список юнит-тестов.

Для приёмочного тестирования проводилась следующая последовательность действий:

1.Установка приложения на компьютер с помощью собранного установочного пакета.

2. Запуск приложения. Окно программы должно быть пустым.

3. Создать три контакта в приложении с разными фамилиями.

4. Переключиться между контактами, убедиться, что смена текущего контакта в правой панели происходит корректно.

5. Ввести в поиск подстроку для поиска контактов - в списке контактов должны остаться только контакты, содержащие подстроку.

6. Введите в поиск подстроку, которой нет в фамилиях и именах контактов - список контактов должен быть пустым. Удалить подстроку поиска - список контактов должен восстановиться.

7. Выбрать любой контакт и нажать кнопку редактирования. Должно открыться окно редактирования контакта.

8. Ввести фамилию более 50 символов. Убедиться, что элемент управления с фамилией указывает на некорректное значение.

9. Ввести фамилию менее 50 символов. Убедиться, что элемент управления с фамилией становиться корректным.

10. Показать, что защита от некорректных значений также работает и для других полей.

11. Поменять фамилию контакта, отличную от исходной. Поменять номер телефона. Нажать «OK». Отредактированный контакт должен переместиться в списке контактов согласно алфавиту, отображаемые данные текущего контакта в правой панели также должны измениться.

12. Выбрать любой контакт и нажать кнопку редактирования. Должно открыться окно редактирования. Изменить фамилию контакта, номер телефона и e-mail. Нажать «Cancel». Исходный контакт должен остаться без изменений.

13. Удалить третий контакт.

14. Закрыть приложение. Должно произойти сохранение контактов в целевой файл.

15. Запустить приложение. В программе должны восстановиться контакты, созданные в предыдущую сессию.

16. Исследовательское тестирование программы руководителем.

**8 Сборка установщика**

Сборка проекта осуществляется в автоматическом режиме. Для создания сценарий сборки инсталлятора, использовалась программа InnoSetup. Сборка установочного файла осуществляется автоматически во время сборки программы. Установщик располагается в директории с исходным проектом. Сценарий сборки для проекта ContactsAppUI перед сборкой проекта

rd /S /Q "$(SolutionDir)InstallScripts\Release"

rd /S /Q "$(SolutionDir)InstallScripts\Installers"

Сценарий события после сборки проекта ContactsAppUI:

md "$(SolutionDir)InstallScripts"

md "$(SolutionDir)InstallScripts\Release"

md "$(SolutionDir)InstallScripts\Installers"

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

"$(SolutionDir)InstallScripts\Release\"

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

"$(SolutionDir)InstallScripts\Release\"

где:

«md» - создание директории;

«rd /S /Q» - удаление директории;

«xcopy» - копирование из одной директории в другую.

Для проекта Installer сценарий события после сборки имеет вид:

"$(SolutionDir)packages\Tools.InnoSetup.6.2.0\tools\ISCC.exe" "$(SolutionDir)InstallScripts\installer.iss"

**9 Описание модели ветвления**

Во время разработки программы для обеспечения версионного контроля использовался сервис GitHub. Ссылка на репозиторий:

<https://github.com/Pomidoriya/ContactsApp>

Для разработки использовалось две ветки: master - главная ветвь, содержащая стабильную и отлаженную версию программы, и develop - ветвь, для разработки и тестирования нового функционала.

На рисунке 9.1– показана история фиксаций ветки develop.

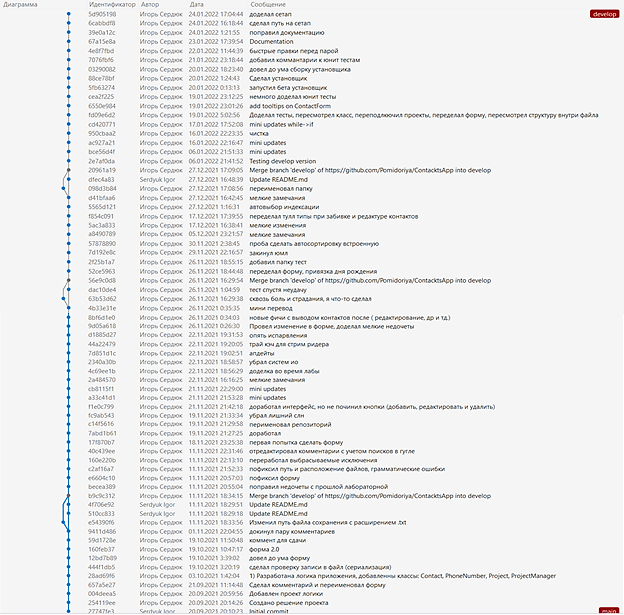


Рисунок 9.1 – Полная история фиксаций ветки develop

# **Список литературы**

Команды Windows [Электронный ресурс]. – URL: <https://docs.microsoft.com/ru-ru/windows-server/administration/windows-commands/windows-commands>

GitHub [Электронный ресурс]. – URL: https://github.com