

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 практической работе «ContactsApp»  
«Новые технологии в программировании»

Выполнил:  
студент гр. 589-3  
\_\_\_\_\_ Дерябин Н.А.  
« \_\_\_\_ » \_\_\_\_\_ 2022 г.

Проверил:  
Старший научный сотрудник  
\_\_\_\_\_ Горяинов А.Е.  
« \_\_\_\_ » \_\_\_\_\_ 2022 г.

## Содержание

Назначение приложения .....	3
Стек технологий разработки.....	3
Пользовательский интерфейс .....	4
Диаграмма пакетов приложения .....	7
Диаграммы классов приложения.....	8
Описание тестирования приложения .....	8
Описание сборки установщика .....	10
Описание модели ветвления в репозитории проекта .....	13
Заключение .....	19

## **Назначение приложения**

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

- 1) Обеспечивать стабильную работу приложения при порядке 200 контактов.
- 2) Обеспечивать поиск, навигацию по созданным контактам по фамилии и имени.
- 3) Предоставить инструменты для просмотра и редактирования контактов.
- 4) Сохранять и восстанавливать контакты между сессиями приложения.
- 5) Выполнять промежуточные сохранения контактов на машине пользователя на случай аварийного завершения программы, отключения компьютера и т.д. — для защиты от потери данных.

**Приложение-референс:** контакты мобильного телефона

## **Стек технологий разработки.**

Для разработки приложения был выбран язык программирования C#

Выбрана версия .NET Framework 4.7.2.

Для работы со строками и файлами была выбрана библиотека JSON.NET версии 13.0.1.

## Пользовательский интерфейс

После запуска приложения перед пользователем появляется главное окно (рис. 1). Двухколоночная верстка главного окна содержит список всех контактов в левой панели и отображает текущий выбранный контакт в правой панели. В списке контактов показаны фамилии контактов, в один момент времени может быть выбран только один контакт (далее – текущий контакт).

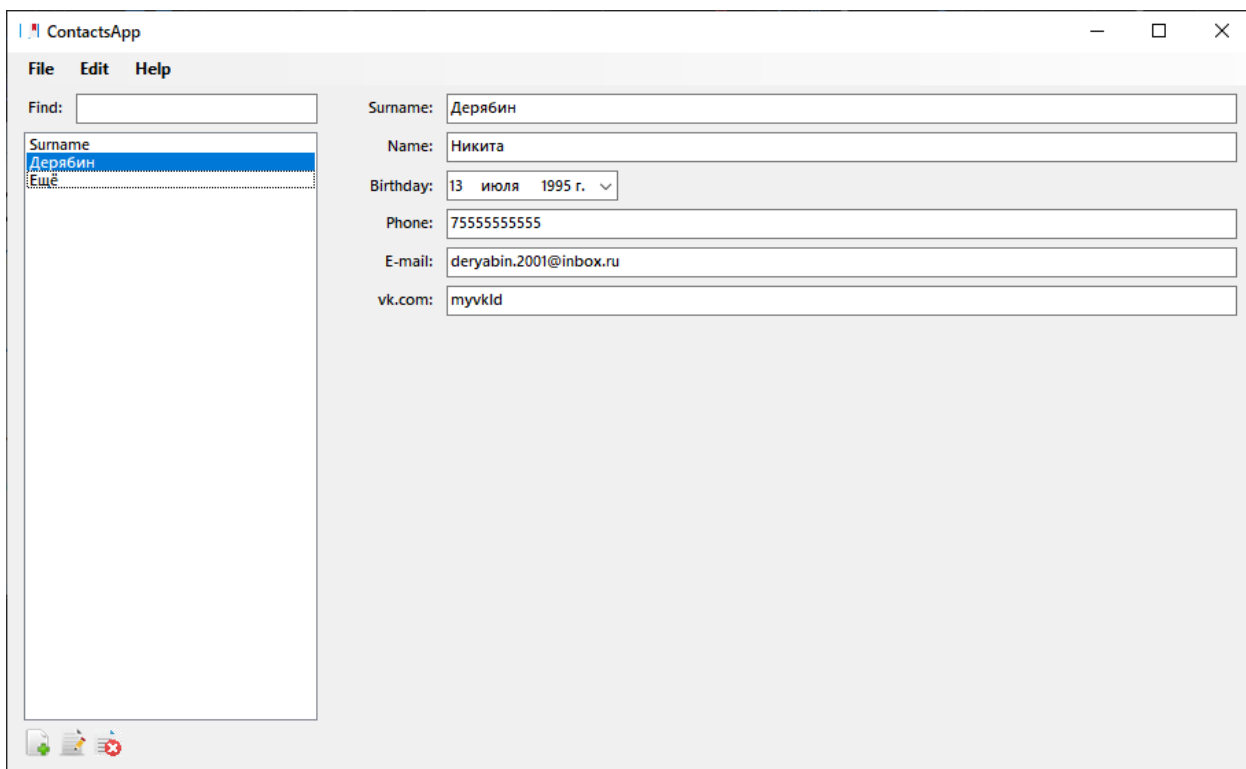


Рисунок 1 – Макет главного окна приложения ContactsApp

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.

При нажатии на кнопку Add Contact и Edit Contact появляется окно создания/редактирования контакта в диалоговом режиме (рис. 2). Для нового контакта окно изначально не заполнено (установлена лишь дата рождения по умолчанию). Для редактирования уже существующего контакта все поля должны быть заполнены данными текущего контакта.

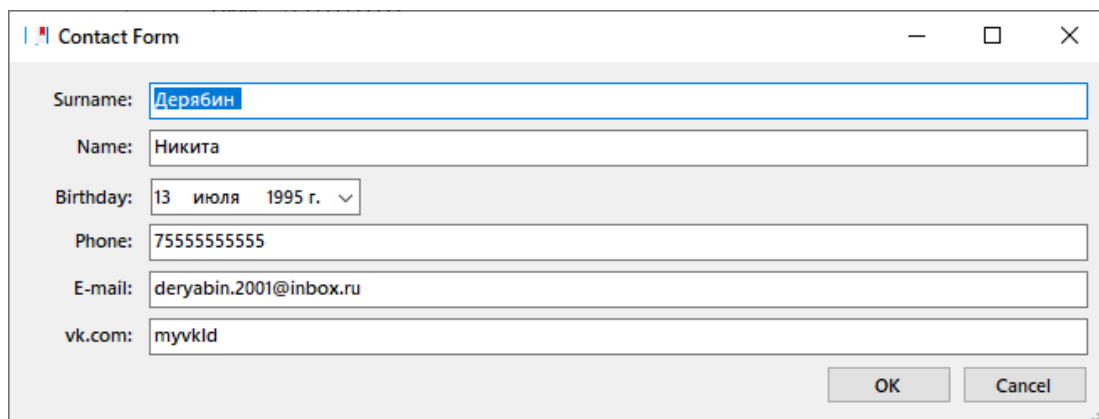
The image shows a standard Windows-style dialog box titled "Contact Form". It contains several input fields: "Surname:" with the text "Дерябин", "Name:" with "Никита", "Birthday:" with a date picker showing "13 июля 1995 г.", "Phone:" with "7555555555", "E-mail:" with "deryabin.2001@inbox.ru", and "vk.com:" with "myvkld". At the bottom right, there are two buttons labeled "OK" and "Cancel".

Рисунок 2 - Макет окна редактирования контакта в приложении  
ContactsApp

При нажатии на кнопку ОК окно создания контакта закрывается, в список контактов главного окна добавляется новый контакт. При редактировании текущей контакта, нажатие на кнопку ОК 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: «Do you really want to remove this contact?». При нажатии на кнопку ОК происходит удаление, при нажатии на кнопку Cancel удаление отменяется.

Меню главного окна содержит следующие пункты:

1. File:
  - 1.1.Exit (Выйти из приложения)
2. Edit:
  - 2.1.Add Contact (Создать новый контакт)
  - 2.2.Edit Contact (Редактировать текущий контакт)
  - 2.3.Remove Contact (Удалить текущий контакт)
3. Help:
  - 3.1.About (Вызвать окно «О программе» ) (см. рис. 3)

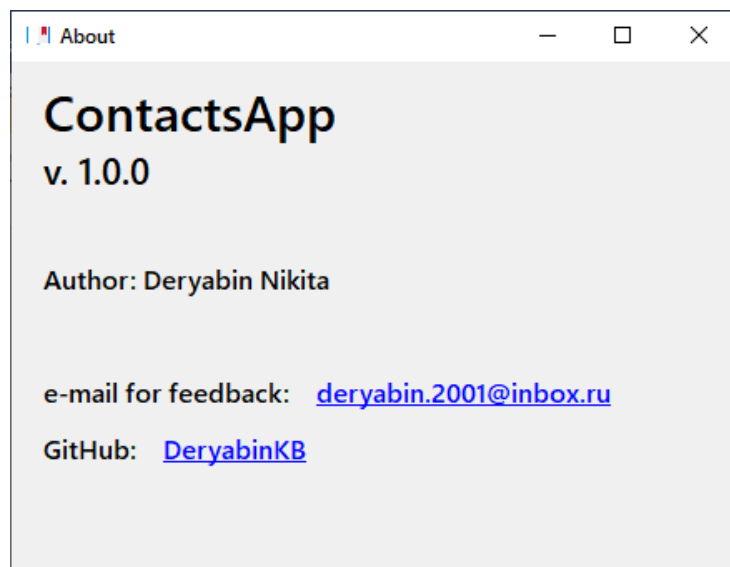


Рисунок 3 - Макет окна «About» приложения ContactsApp.

Таким образом, в программе должно быть реализовано три окна:

- Главное окно.
- Окно создания/редактирования контакта.
- Окно «About».

Верстка главного окна и окна создания/редактирования контакта должна быть адаптивной. Окно «About» имеет фиксированный размер.

Создание дополнительных элементов управления для уменьшения дублирования кода остается на усмотрение разработчика или руководителя.

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

### Диаграмма пакетов приложения

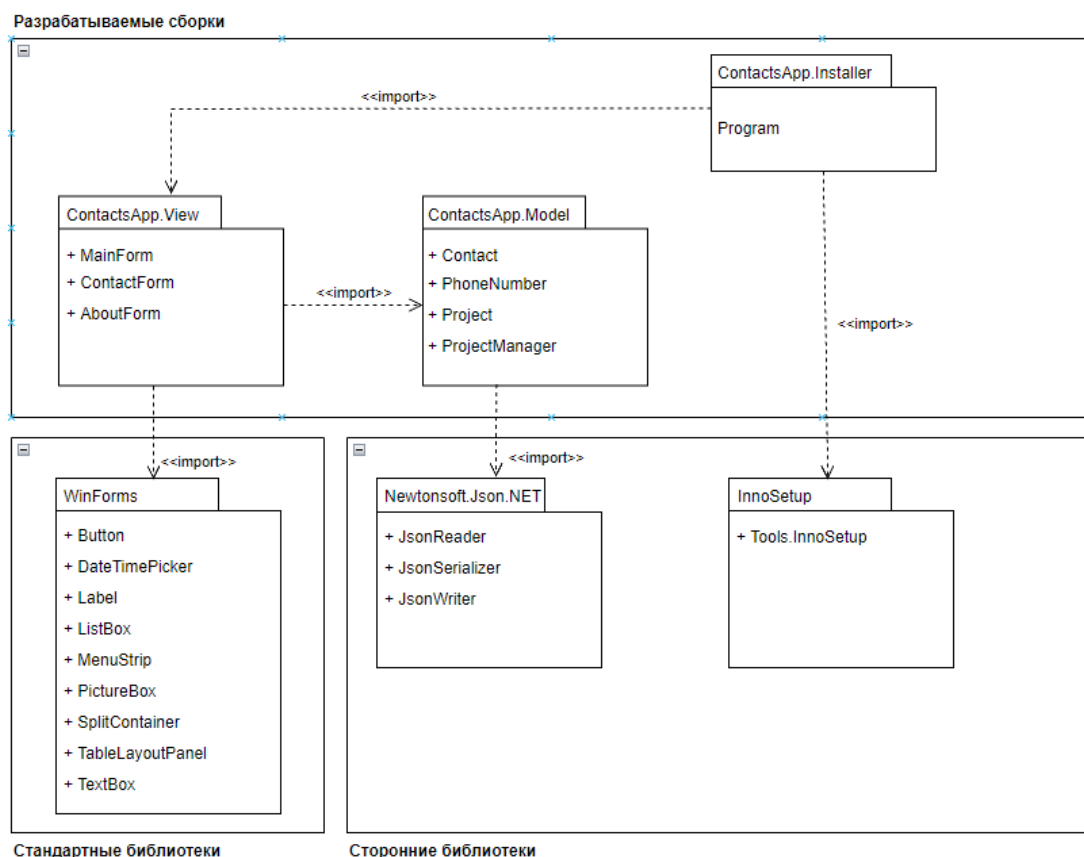


Рисунок 4 – Диаграмма пакетов приложения

## Диаграммы классов приложения

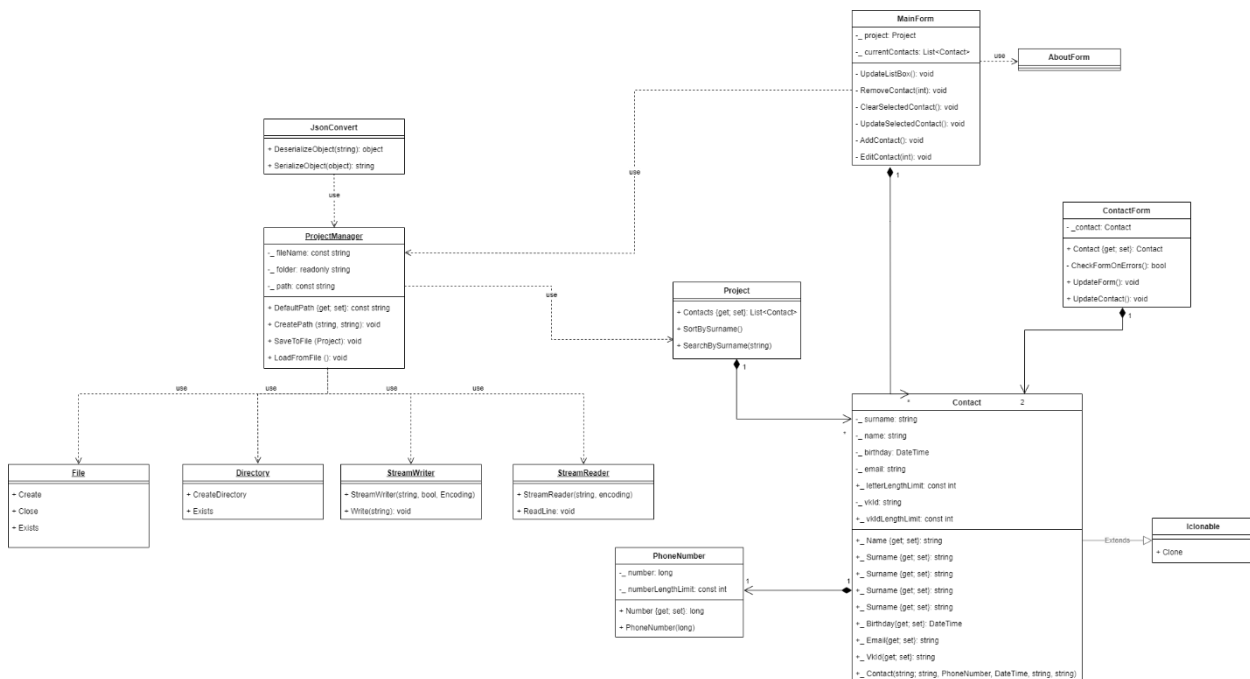


Рисунок 5 – Диаграмма классов приложения

Тестирование классов пользовательского интерфейса проводилось 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 контактов.
3. Создать в приложении три контакта.
4. Переключиться между контактами, чтобы убедиться, что смена текущей заметки происходит корректно.
5. Выбрать второй контакт и нажать кнопку редактирования. Должно открыться окно редактирования контакта.



6. Ввести фамилию контакта более 50 символов. Элемент управления с фамилией контакта должен указать на некорректное значение.

7. Ввести фамилию контакта менее 50 символов. Элемент управления с фамилией контакта должен стать корректным.

8. Поменять фамилию контакта. Нажать «ОК». Данные контакта должны обновиться в главном окне.

9. Выбрать другой контакт и нажать кнопку редактирования. Должно открыться окно редактирования. Изменить одно или несколько полей контакта. Нажать «Cancel». Исходная контакт должен остаться без изменений.

10. Удалить первый контакт.

11. Заккрыть приложение. Должно произойти сохранение контактов в целевой файл.

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

## Описание сборки установщика

```
; Script generated by the Inno Setup Script Wizard.  
; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO  
SETUP SCRIPT FILES!
```

```
#define MyAppName "ContactsApp"  
#define MyAppVersion "1.0.0"  
#define MyAppPublisher "Deryabin Nikita, Inc."  
#define MyAppExeName "ContactsApp.View.exe"  
#define MyAppAssocName MyAppName + " File"  
#define MyAppAssocExt ".myp"  
#define MyAppAssocKey StringChange(MyAppAssocName, " ",  
"" ) + MyAppAssocExt
```

### [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={{0CEAF1EE-C781-44AD-969D-9A19B901F3D5}  
AppName={#MyAppName}  
AppVersion={#MyAppVersion}  
;AppVerName={#MyAppName} {#MyAppVersion}  
AppPublisher={#MyAppPublisher}  
DefaultDirName={autopf}\{#MyAppName}  
ChangesAssociations=yes  
DefaultGroupName={#MyAppName}  
AllowNoIcons=yes  
; Uncomment the following line to run in non  
administrative install mode (install for current user  
only.)  
;PrivilegesRequired=lowest  
OutputDir=C:\ContactsAppSetup  
OutputBaseFilename=ContactsAppSetup  
SetupIconFile=C:\Users\ADMIN\source\repos\ContactsApp\src  
\ContactsApp\ContactsApp.View\BookmarkIcon.ico  
Compression=lzma  
SolidCompression=yes  
WizardStyle=modern
```

### [Languages]

Name: "english"; MessagesFile: "compiler:Default.isl"  
Name: "russian"; MessagesFile:  
"compiler:Languages\Russian.isl"

#### [Tasks]

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

#### [Files]

Source:  
"C:\Users\ADMIN\source\repos\ContactsApp\src\ContactsApp\  
ContactsApp.View\bin\Release\{#MyAppExeName}"; DestDir:  
"{app}"; Flags: ignoreversion  
Source:  
"C:\Users\ADMIN\source\repos\ContactsApp\src\ContactsApp\  
ContactsApp.View\bin\Release\ContactsApp.Model.dll";  
DestDir: "{app}"; Flags: ignoreversion  
Source:  
"C:\Users\ADMIN\source\repos\ContactsApp\src\ContactsApp\  
ContactsApp.View\bin\Release\Newtonsoft.Json.dll";  
DestDir: "{app}"; Flags: ignoreversion  
; NOTE: Don't use "Flags: ignoreversion" on any shared  
system files

#### [Registry]

Root: HKA; Subkey:  
"Software\Classes\{#MyAppAssocExt}\OpenWithProgids";  
ValueType: string; ValueName: "{#MyAppAssocKey}";  
ValueData: ""; Flags: uninsdeletevalue  
Root: HKA; Subkey: "Software\Classes\{#MyAppAssocKey}";  
ValueType: string; ValueName: ""; ValueData:  
"{#MyAppAssocName}"; Flags: uninsdeletekey  
Root: HKA; Subkey:  
"Software\Classes\{#MyAppAssocKey}\DefaultIcon";  
ValueType: string; ValueName: ""; ValueData:  
"{app}\{#MyAppExeName},0"  
Root: HKA; Subkey:  
"Software\Classes\{#MyAppAssocKey}\shell\open\command";  
ValueType: string; ValueName: ""; ValueData:  
""""{app}\{#MyAppExeName}"" ""%1""  
Root: HKA; Subkey:  
"Software\Classes\Applications\{#MyAppExeName}\SupportedT

```
ypes"; ValueType: string; ValueName: ".myp"; ValueData:  
""
```

#### [Icons]

```
Name: "{group}\{#MyAppName}"; Filename:  
"{app}\{#MyAppExeName}"  
Name: "{group}\{cm:UninstallProgram,{#MyAppName}}";  
Filename: "{uninstallexe}"  
Name: "{autodesktop}\{#MyAppName}"; Filename:  
"{app}\{#MyAppExeName}"; Tasks: desktopicon
```

#### [Run]

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

## Описание модели ветвления в репозитории проекта

Ссылка на репозиторий <https://github.com/DeryabinKB/ContactsApp>  
Центральный репозиторий содержит две главные ветки, существующие всё время.

- master
- develop

Ветвь master создаётся при инициализации репозитория, что должно быть знакомо каждому пользователю Git. Параллельно ей также мы создаём ветку для разработки под названием develop.

Origin/master – главная ветка. Исходный код в ней должен находиться в состоянии production-ready в любой произвольный момент времени.

Ветвь origin/develop – главная ветвь для разработки. Хранящийся в ней код в любой момент времени должен содержать самые последние изданные изменения, необходимые для следующего релиза. Эту ветку также можно назвать «интеграционной». Она служит источником для сборки автоматических ночных билдов.

Когда исходный код в ветви разработки (develop) достигает стабильного состояния и готов к релизу, все изменения должны быть определённым способом влиты в главную ветвь (master) и помечены тегом с номером релиза. Ниже мы рассмотрим этот процесс в деталях.

Помимо главных ветвей master и develop, наша модель разработки содержит некоторое количество типов вспомогательных ветвей, которые используются для распараллеливания разработки между членами команды, для упрощения внедрения нового функционала (features), для подготовки релизов и для быстрого исправления проблем в производственной версии приложения. Эти ветви всегда имеют ограниченный срок жизни. Каждая из них в конечном итоге рано или поздно удаляется.

Используются следующие типы ветвей:

- Ветви функциональностей (Feature branches)
- Ветви релизов (Release branches)
- Ветви исправлений (Hotfix branches)

На рисунках 6-10 показана история фиксаций ветки develop.

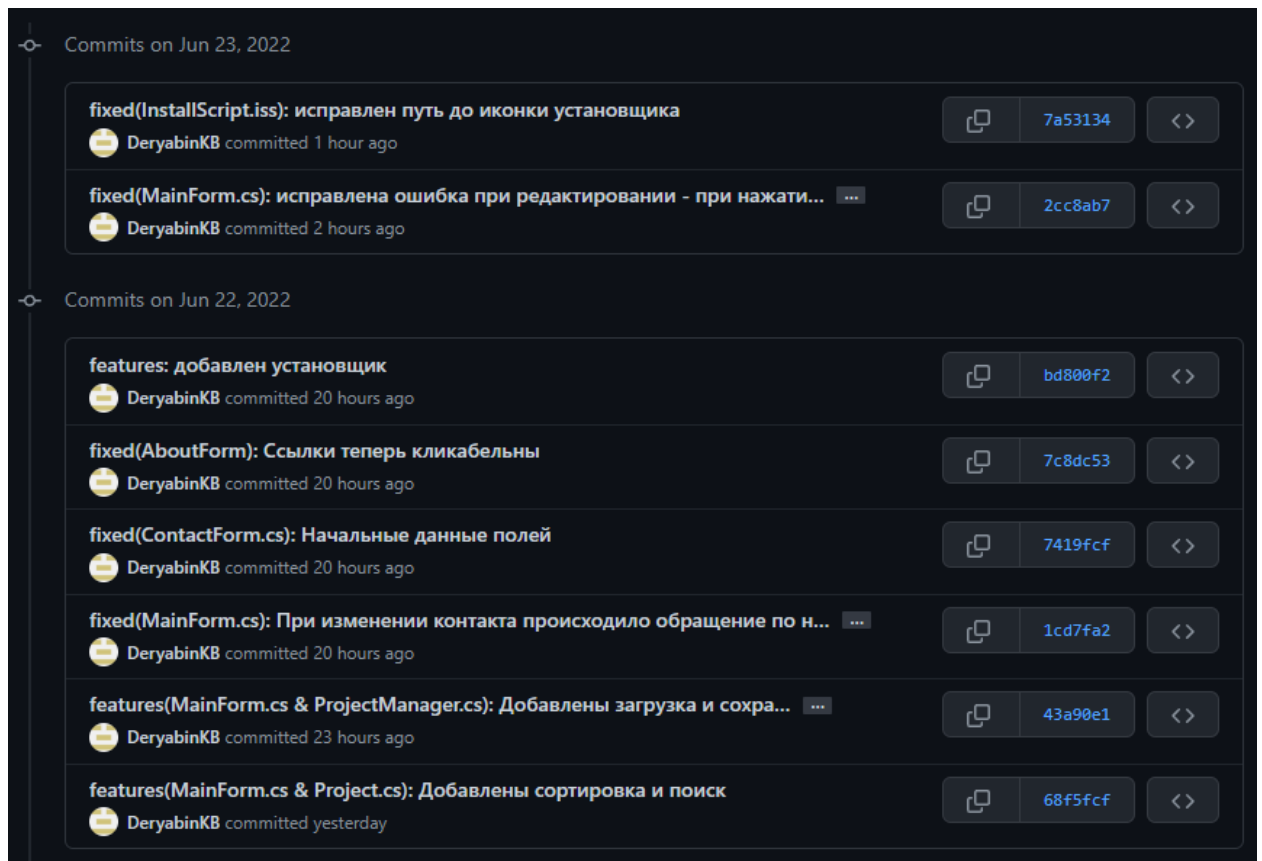


Рисунок 6 – Коммиты в ветку develop

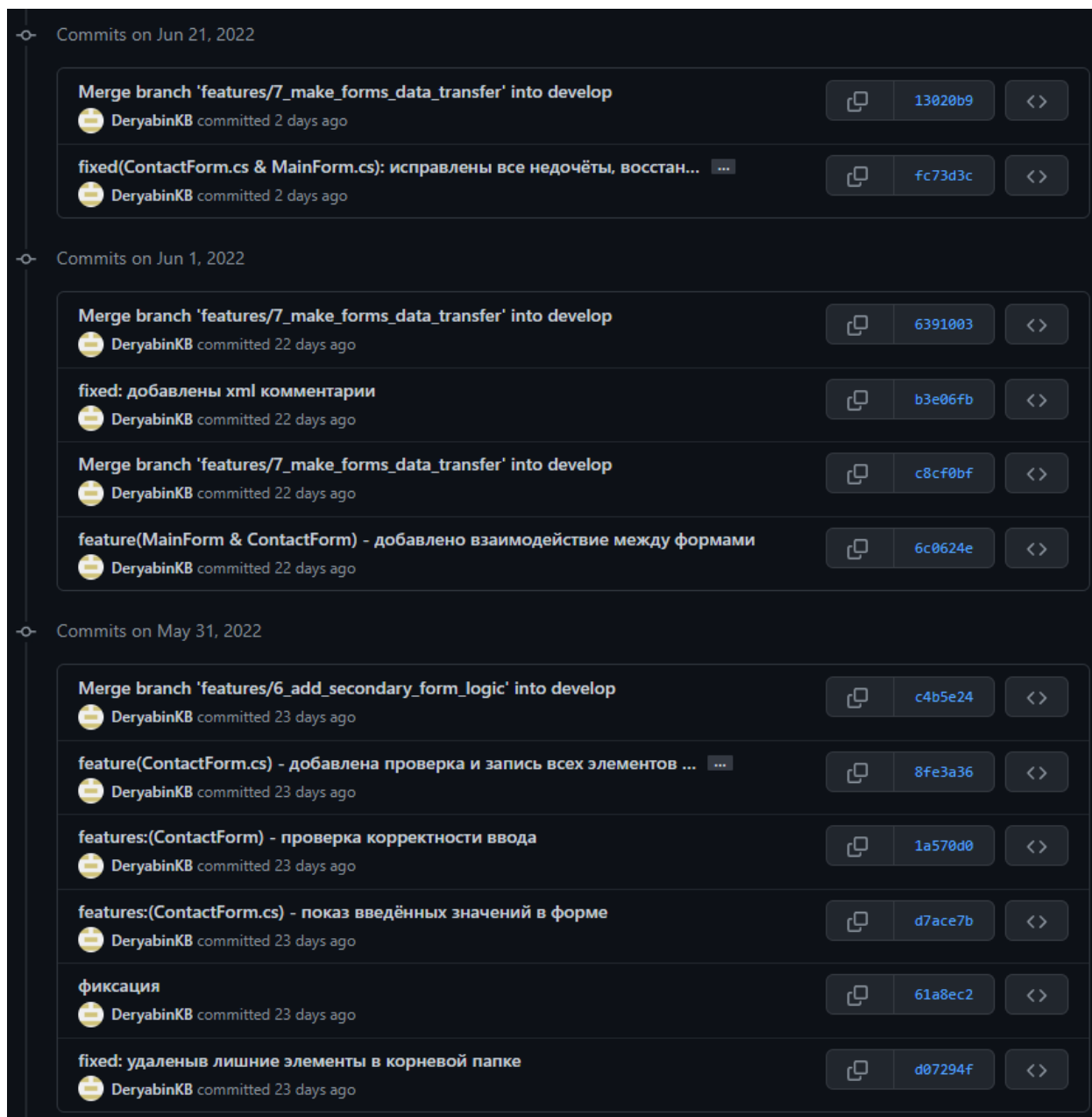


Рисунок 7 – Коммиты в ветку develop

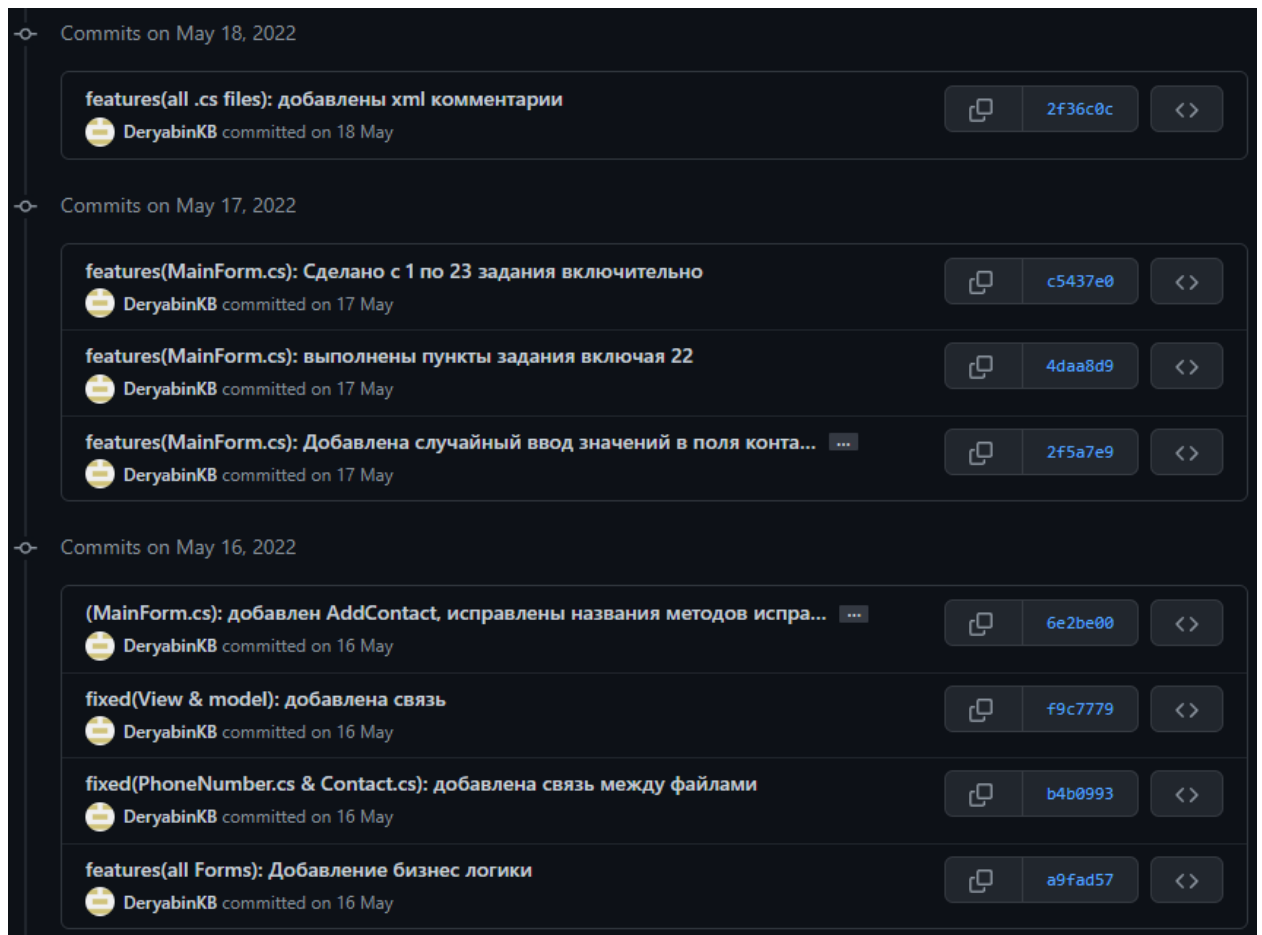


Рисунок 8 – Коммиты в ветку develop



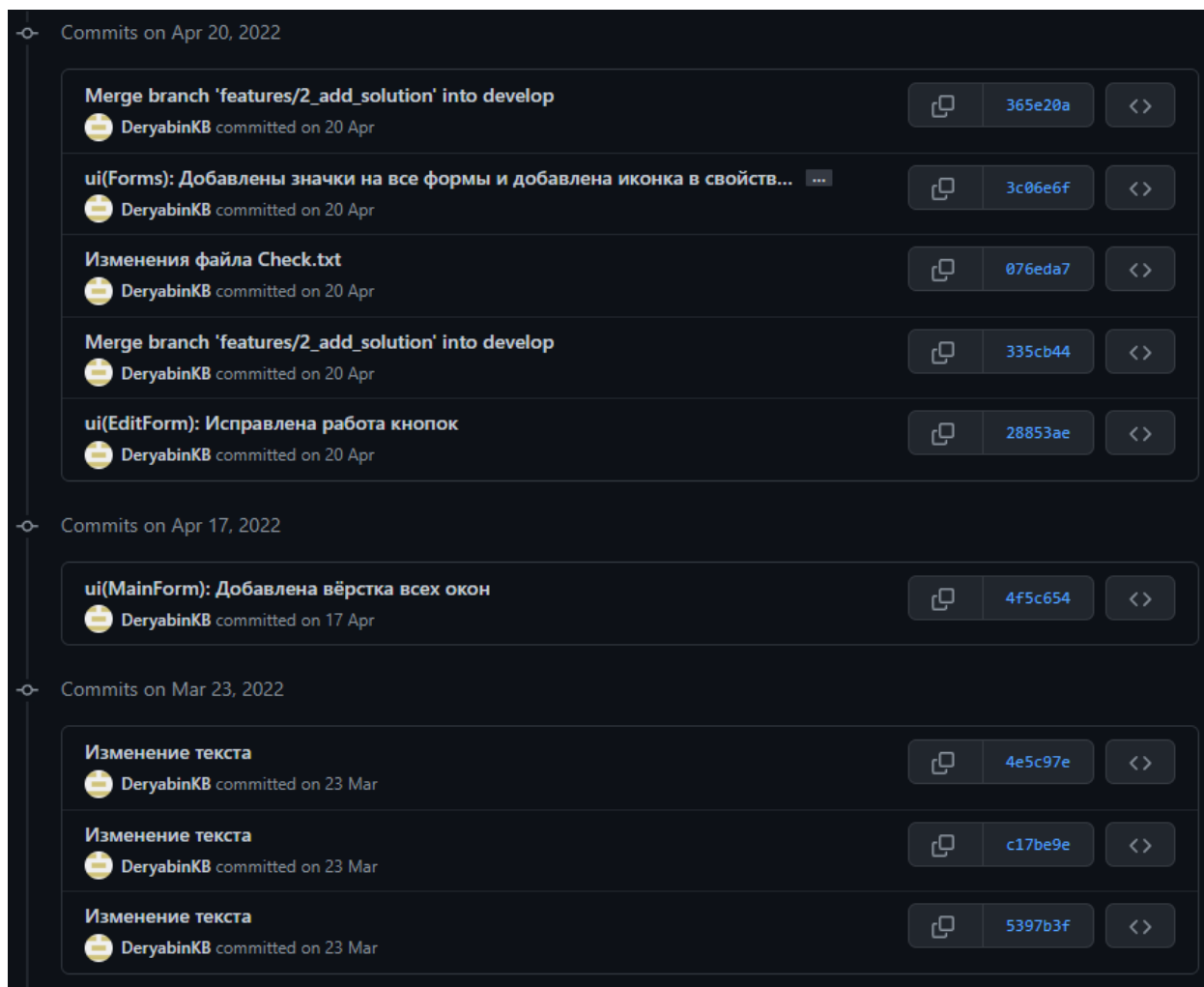


Рисунок 9 – Коммиты в ветку develop

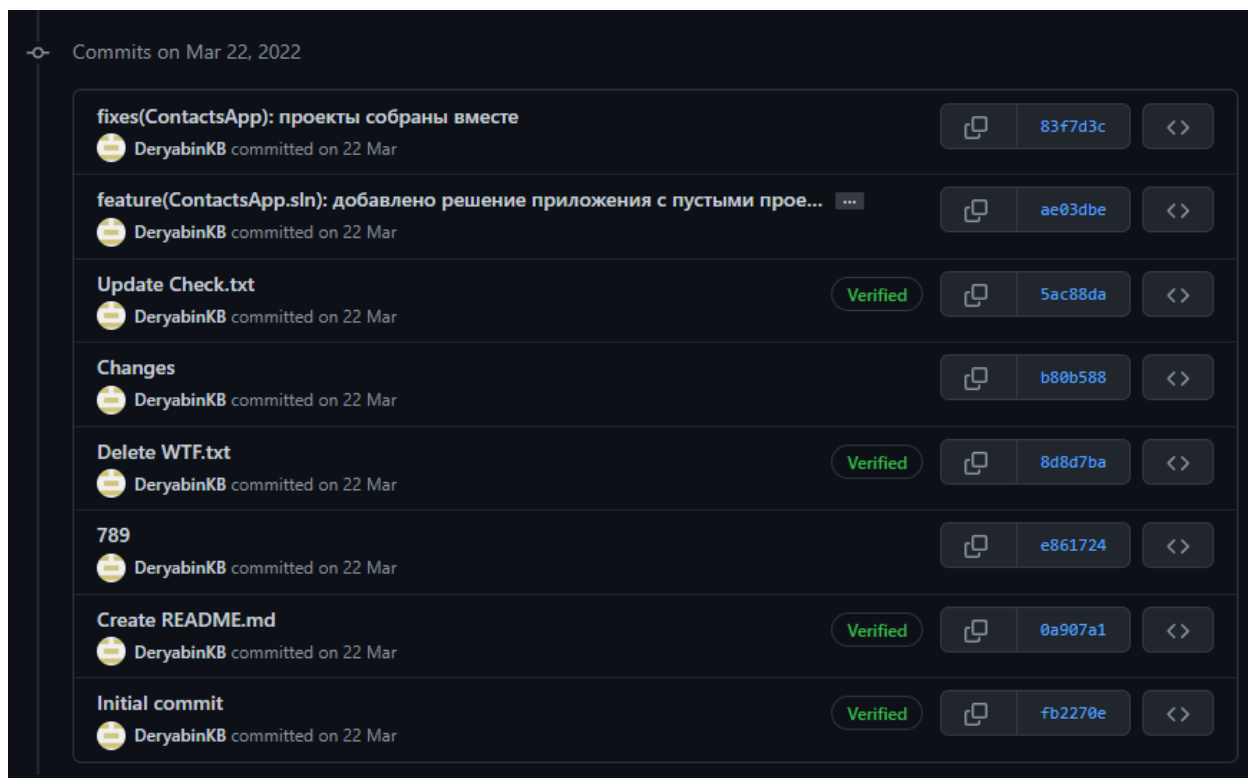


Рисунок 10 – Коммиты в ветку develop

## **Заключение**

Поставленная задача была выполнена в полном новом объеме (9 из 9) согласно представленного ТЗ, так же был реализован установщик для нашего приложения.

Поставленная задача была выполнена в новый поставленный срок. Для выполнения всех задач потребовалось около 30-40ч. Основное время ушло на изучение и повторение синтаксиса языка программирования С#. На написание документации у меня ушло приблизительно 3-3,5ч.

Благодаря четко сформулированному ТЗ проблем с реализацией логики приложения не было. Версионный контроль проблем не вызвал из-за знаний английского языка на базовом уровне и наличия сотен обучающих видео на всевозможных языках.

Основная проблема – из-за наличия нескольких не самых приятных предметов, которые сжирали всё время, к программированию получалось возвращаться только за день-два перед самой лабораторной (данной высказывание характерно для временного промежутка от середины апреля до середины июня). До середины апреля само собой были видеомонтаж, компьютерные игры и прокрастинация.

Общее заключение: на протяжении разработки приложения было усвоено много материала. Эта информация может пригодиться для улучшения моих навыков в сфере разработки приложений.