**УНИВЕРСИТЕТ ПО БИБЛИОТЕКОЗНАНИЕ И ИНФОРМАЦИОННИ ТЕХНОЛОГИИ**

**ФАКУЛТЕТ „ИНФОРМАЦИОННИ НАУКИ“**

**К У Р С О В А Р А Б О Т А**

по

ВИЗУАЛНИ СРЕДИ ЗА ПРОГРАМИРАНЕ

**ТЕМА**: **„Използване на визуални среди за програмиране за създаване на персонализирано приложение, представляващо пациентски картон във ветеринарна клиника“**

Изготвил: Преподавател:

Росица Ненова проф. д-р Иван Иванов

ФН 46231з

София

2020

Съдържание

[Увод 3](#_Toc57577436)

[I. Функции на приложението 4](#_Toc57577437)

[1. Информация за животното 4](#_Toc57577438)

[2. Имунизация 4](#_Toc57577439)

[3. Снимка 5](#_Toc57577440)

[4. Информация за собственик 5](#_Toc57577441)

[5. Преглед 5](#_Toc57577442)

[6. Общи положения 5](#_Toc57577443)

[II. Създаване на дизайна 5](#_Toc57577444)

[1. Създаване на ново *Windows* *приложение* 6](#_Toc57577445)

[2. Вмъкване на контролите *GroupBox* 6](#_Toc57577446)

[3. Вмъкване на текстовите полета 6](#_Toc57577447)

[4. Вмъкване на календари 7](#_Toc57577448)

[5. Вмъкване на етикети 7](#_Toc57577449)

[6. Вмъкване на радио-бутони 8](#_Toc57577450)

[7. Вмъкване на бутони 8](#_Toc57577451)

[8. Вмъкване на допълнителни контроли 9](#_Toc57577452)

[III. Създаване на функционалностите 10](#_Toc57577453)

[1. Метод за валидиране дали текстовите полета са празни 10](#_Toc57577454)

[2. Метод за валидиране на въведения телефонен номер на собственика 11](#_Toc57577455)

[3. Метод за деактивиране на календар и бутон за изчисляване на следваща имунизация 12](#_Toc57577456)

[4. Метод за активиране на календар и бутон 13](#_Toc57577457)

[5. Метод за изчисляване на дата за следваща имунизация 13](#_Toc57577458)

[6. Метод за показване на специално съобщение при избиране на опция „Не“ в поле „Кастриране“ 14](#_Toc57577459)

[7. Методи за деактивиране/активиране на *CheckBox* 15](#_Toc57577460)

[8. Метод за изчисляване на възраст 16](#_Toc57577461)

[9. Метод за качване на снимка към досието 16](#_Toc57577462)

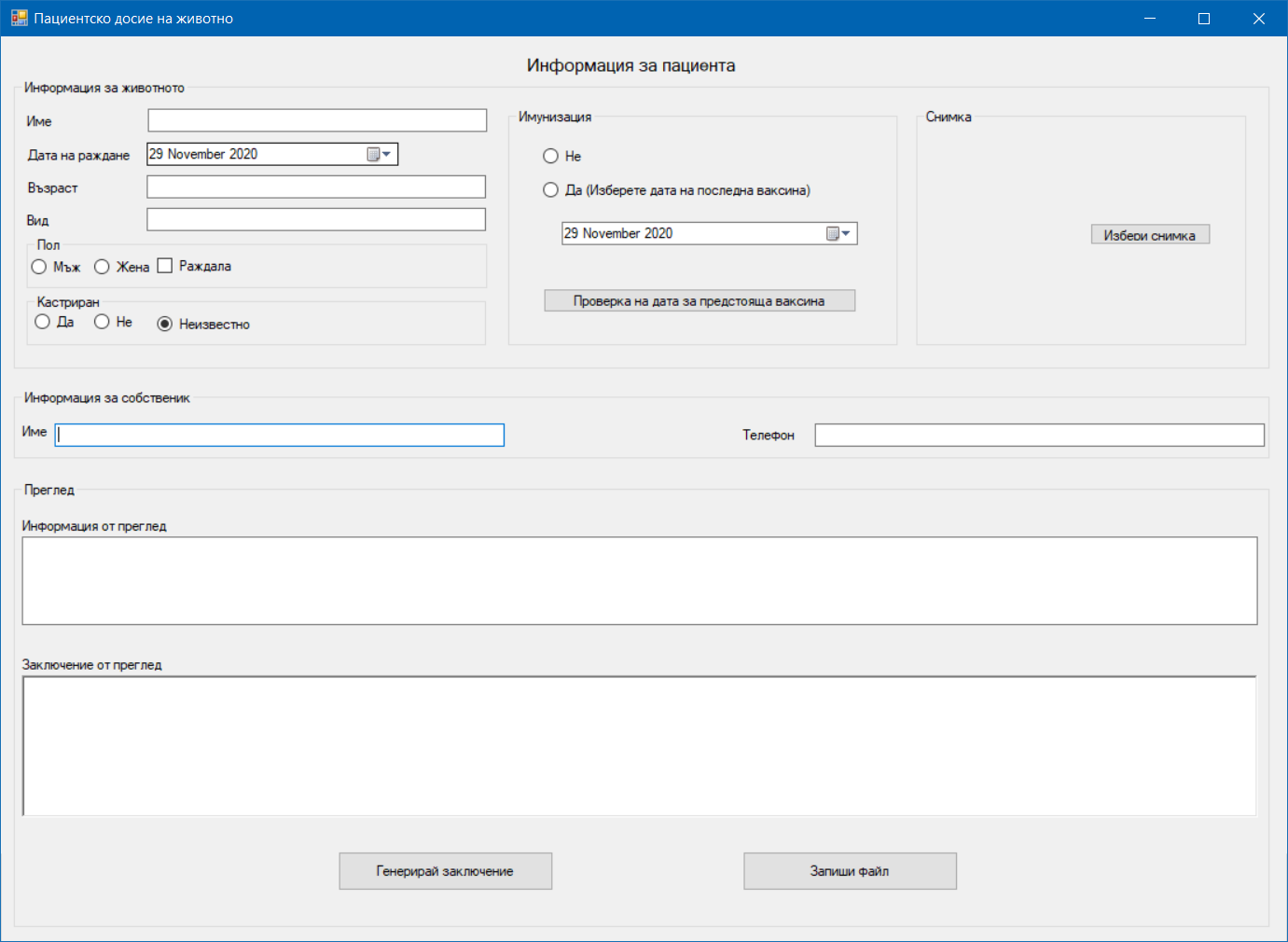
[10. Метод за попълване на текстово поле „Заключение от преглед“ 16](#_Toc57577463)

[11. Метод за генериране на файл с текста от текстово поле „Заключение от преглед“ 17](#_Toc57577464)

[Библиография 19](#_Toc57577465)

# Увод

В настоящата курсова работа ще разгледаме последователните стъпки и използваните контроли и код за създаване на приложение за въвеждане на данни при преглед на животно във ветеринарна клиника, генериране на заключение от прегледа, както и за запазване на въведените данни в изходен файл в .rtf формат. Потребителският интерфейс е показан по-долу на Фигура 1.



Фигура Изглед на краен потребителски интерфейс

## Функции на приложението

За да функционира коректно приложението за въвеждане на данни на животно-пациент при преглед във ветеринарна клиника правилно, в него се визуализират полета, които трябва да бъдат попълнени от потребителя.

### Информация за животното

Приложението дава възможност на потребителя да въвежда в свободен текст данни като име и вид на животното-пациент.

Датата на раждане се избира от вграден календар, а възрастта се изчислява на база да въведената дата на раждане при кликване върху полето, в която трябва да се изобрази тя.

Полът се избира чрез радио-бутони, като при избиране на пол „Жена“ се дава възможност на потребителя да избере дали животното е раждало или не, а при избиране на пол „Мъж“ тази опция се деактивира.

Потребителският интерфейс дава възможност да се въведе информация дали животното е кастрирано, като при избор на опция „Не“ се визуализира прозорец със специално съобщение, отправено към потребителя. По подразбиране стои избрана опцията „Неизвестно“.

### Имунизация

Потребителят въвежда и информация за направена имунизация. В случай, че е избрана опция „Да“, то потребителят получава в допълнителен прозорец специално съобщение, което съдържа инструкции за последващи действия от страна на потребителя. В полето за информация относно имунизацията има специален бутон, който изчислява кога трябва да бъде следващата ваксинация на пациента. В случай, че е избрана опция „Не“, календарът и специалният бутон се деактивират, тъй като нямат връзка с така въведената информация.

### Снимка

Приложението дава възможност да се качи снимка на пациента. Това става чрез бутон, създаден специално за тази цел, който отваря диалогов прозорец, чрез който се избира снимка от файловата система на компютъра на потребителя.

### Информация за собственик

Приложението дава възможност да се въведе информация за собственика на животното-пациент. Прави се проверка дали телефонният номер е въведен в коректен за България формат. В случай, че не е въведен правилно, потребителят получава съобщение за това, заедно с инструкции какъв е правилният формат.

### Преглед

В тази секция приложението има две текстови полета. В първото поле се въвежда информация за текущия преглед и е в свободен текст.

С помощта на бутон „Генерирай заключение“ във второто поле се генерира подробна информация за животното-пациент и прегледа, както и препоръки при някои особени случаи. Препоръките зависят от въведените данни от полета „Информация за животното“ и „Имунизация“.

С помощта на бутон „Запиши файл“ се отваря диалогов прозорец, който позволява на потребителя да запише данните от текстово поле „Заключение от преглед“ в .rtf файл.

### Общи положения

Приложението помага на потребителя, като оцветява в червено полетата, в които не е въведена информация или е въведена грешна такава. Това са всички текстови полета, които трябва да бъдат попълнени от потребителя.

## Създаване на дизайна

Създаването на формата със съпътстващите я компоненти се изпълнява в няколко стъпки.

### Създаване на ново *Windows* *приложение*

Създава се ново *Windows* *приложение* с наименование VSP\_46231z\_10. Размерите на формата е препоръчително да са „1487, 1000“. Минималният размер и максималният размер получават същите стойности. Стойността на свойството *Name* е „Пациентско досие на животното“.

### Вмъкване на контролите *GroupBox*

Вмъкват се седем контроли от типа *GroupBox*. В Таблица 1 са показани стойностите, които трябва да се присвоят на *Name* и *Text* свойствата.

|  |  |  |
| --- | --- | --- |
| GroupBox | Name | Text |
| 1 | groupBoxAnimalData | Информация за животното |
| 2 | groupBoxSex | Пол |
| 3 | groupBoxNeutered | Кастриран |
| 4 | groupBoxVax | Имунизация |
| 5 | groupBoxAvatar | Снимка |
| 6 | groupBoxOwnerInfo | Информация за собственик |
| 7 | groupBoxCheck | Преглед |

Таблица Свойства на GroupBox контролите

### Вмъкване на текстовите полета

Вмъкват се шест текстови полета от типа *TextBox*. В Таблица 2 са показани стойностите, които трябва да се присвоят на Name свойствата.

|  |  |
| --- | --- |
| TextBox | Name |
| 1 | textBoxName |
| 2 | textBoxAge |
| 3 | textBoxType |
| 4 | textBoxOwnerName |
| 5 | textBoxOwnerPhone |
| 6 | textBoxCheckInfo |

Таблица Свойства на TextBox контролите

Добавя се и едно поле от типа *RichTextBox* със стойност *RichTextBoxFinal* на свойството *Name*.

### Вмъкване на календари

Вмъкват се две контроли от тип DateTimePicker. В Таблица 3 са показани стойностите на свойството Name.

|  |  |
| --- | --- |
| DateTimePicker | Name |
| 1 | dateTimePickerBirth |
| 2 | dateTimePickerVax |

Таблица Свойства на DateTimePicker контролите

### Вмъкване на етикети

Към вмъкнатите до момента контроли трябва да се добавят и контроли от типа *Label*, за да се указва на потребителя в кое поле каква информация трябва да се въведе. За целта се добавят осем контроли *Label*, като на Таблица 4 са показани стойностите на свойствата им *Text* и *Name*.

|  |  |  |
| --- | --- | --- |
| Label | Name | Text |
| 1 | labelName | Име |
| 2 | labelBirthDate | Дата на раждане |
| 3 | labelAge | Възраст |
| 4 | labelType | Вид |
| 5 | labelOwnerName | Име |
| 6 | labelOwnerPhone | Телефон |
| 7 | labelCheckInfo | Информация от преглед |
| 8 | labelFinalInfo | Заключение от преглед |

Таблица Свойства на Label контролите

### Вмъкване на радио-бутони

За някои данни относно животното-пациент е нужно да се избират определени стойности. Това ще стане с помощта на контролите *RadioButton*. По-долу в Таблица 5 са показани стойностите на свойствата им, както и в кой GroupBox контрол трябва да бъдат поставени.

|  |  |  |  |
| --- | --- | --- | --- |
| RadioButton | Name | Text | GroupBox |
| 1 | radioButtonSexMale | Мъж | Sex |
| 2 | radioButtonSexFemale | Жена | Sex |
| 3 | radioButtonNeuteredYes | Да | Neutered |
| 4 | radioButtonNeuteredNo | Не | Neutered |
| 5 | radioButtonNeuteredUnknown | Неизвестно | Neutered |
| 6 | radioButtonVaxNo | Не | Vax |
| 7 | radioButtonVaxYes | Да | Vax |

Таблица Свойства на RadioButton контролите

Радио-бутонът с име „radioButtonNeuteredUnknown“ се настройва да е избран по подразбиране чрез стойност „True“ на свойството *Checked*.

### Вмъкване на бутони

За да се използват повечето функционалности на приложението е нужно да се вмъкнат четири бутона, чиито свойства са посочени в Таблица 6, както и към кой *GroupBox* контрол принадлежат.

|  |  |  |  |
| --- | --- | --- | --- |
| Button | Name | Text | GroupBox |
| 1 | buttonCheckVaxDate | Проверка на дата за предстояща ваксина | Vax |
| 2 | buttonChoseAvatar | Избери снимка | Avatar |
| 3 | buttonGenerateFinalInfo | Запиши файл | Check |
| 4 | buttonSave | Генерирай заключение | Check |

Таблица Свойства на Button контролите

### Вмъкване на допълнителни контроли

За да получим завършен вид на приложението е нужно да добавим един контрол от тип *CheckBox*, един от тип *OpenFileDialog*, един от тип *SaveFileDialog* и един от тип *PictureBox*.

Свойството на контрола *CheckBox Name* получава стойност „checkBoxSexGivenBirth“, а свойството *Text* - „Раждала“.

Свойството на контрола *OpenFileDialog* Name получава стойност „OpenAvatarDialog“, a свойството *Filter* – „Image files(\*.jpg;\*.gif)|\*jpg;\*gif|All Files(\*.\*)|\*.\*“. Този контрол се използва, за да се отвори диалогов прозорец, от който да се избере снимка, която да се прикачи към пациентското досие на животното. Чрез свойството Filter и неговата стойност се указва на приложението да показва само файлове с разширените .jpg и .gif или при избор от страна на потребителя да показва всички файлове.

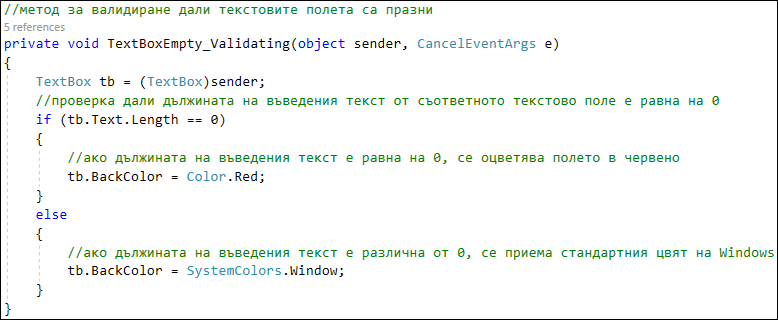
Свойството на *SaveFileDialog* *Name* получава стойност „saveFileDialogInfo“. Чрез този контрол на по-късен етап ще се програмира функционалността на приложението да запазва информацията от досието в .rtf файл.

Свойството на *PictureBox* *Name* получава стойност „pictureBoxAvatar“. В този контрол ще се изборазява качената снимка на животното. Тъй като на потребителя не е указано в какъв размер ще е снимката, чрез стойността „Zoom“ на свойството *SizeMode* контролираме размера на изображението и то ще се вижда според размера на *PictureBox* контрола.

## Създаване на функционалностите

### Метод за валидиране дали текстовите полета са празни

Чрез методът по-долу на Фигура 2 се валидира дали текстовите полета са празни. Ако са празни, т.е. дължината на текста в тях е равна на 0 (нула), то те се оцветяват в червен цвят. Съответно, ако не са празни, фоновият им цвят е стандартният за Windows.



Фигура Метод за валидиране дали текстовите полета са празни

Този метод се задава като стойност за събитие *Validating* на контролите от тип *TextBox*, с изключение на контрола с име „textBoxOwnerPhone“.

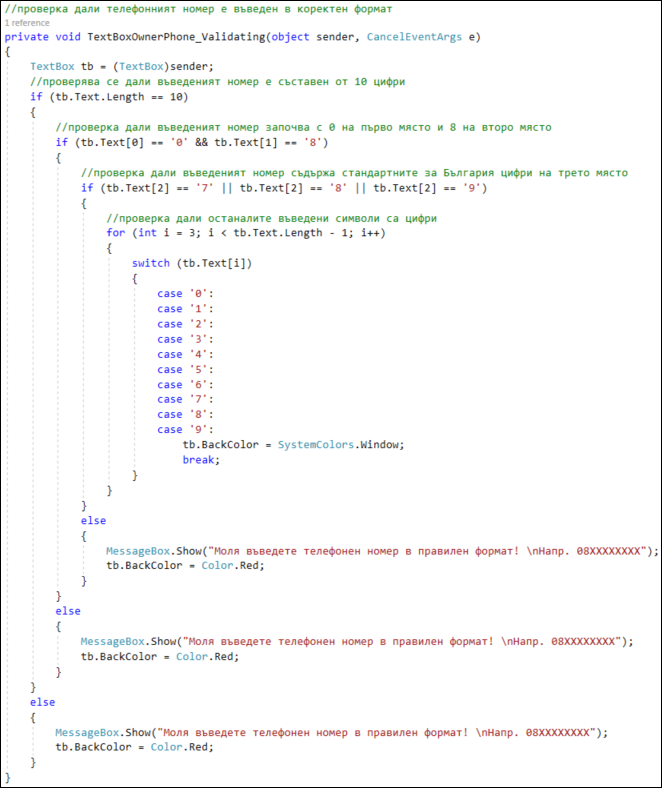
На Фигура 3 се изобразява функционалността на този метод чрез две от полетата, като едното съдържа текст, а другото – не.



Фигура Функционалност на метода TextBoxEmpty\_Validating()

### Метод за валидиране на въведения телефонен номер на собственика

Чрез метода, показан на Фигура 4 се валидира дали въведения телефонен номер на собственика отговаря на стандартния формат за България.

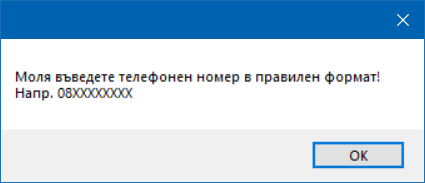


Фигура Метод за валидиране на въведен телефонен номер

Първо се прави проверка дали въведеният текст е с дължина 10 символа. Ако проверката е успешна, се прави втора такава, която валидира дали първия и втория символ са съоветно 0 и 8. Ако и тази проверка е успешна, се валидира третия символ. Ако той отговаря на стойност 7, 8 или 9, то се прави последна проверка, която валидира дали останалите 7 символа са цифри от 0 до 9.

В случай, че всички проверки са успешни, то приложението приема така въведения телефонен номер.

В случай, че някоя от проверките не е успешна, потребителят получава съобщение, че трябва телефонният номер да е въведен в коректен формат. Съобщението се вижда на Фигура 5.

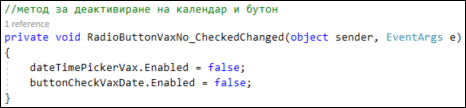


Фигура Съобщение за грешка

Този метод се задава като стойност на събитие *Validating* на контрола с име „textBoxOwnerPhone“.

### Метод за деактивиране на календар и бутон за изчисляване на следваща имунизация

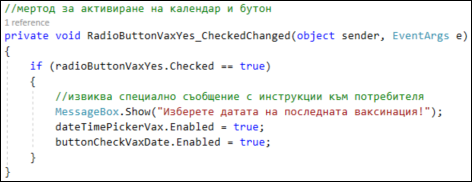
Функционалността на приложението позволява при избор на опция „Не“ от поле „Имунизация“ да се деактивират календара и бутона за изчисляване на следваща дата за ваксина. Това става чрез метода, показан на Фигура 6. Този метод се задава като стойност на събитието *CheckedChange* на радио-бутона с име „radioButtonVaxNo“.



Фигура Метод за деактивиране на календар и бутон

### Метод за активиране на календар и бутон

Аналогично на предната точка и посочения в нея метод, с показания на Фигура 7 метод се активират отново календарът и бутонът за изчисляване на следващата имунизация. Разликата с предходния метод е, че при този има допълнителна функция. Тя се изразява в това, че се визуализира нов прозорец, който съдържа специално съобщение към потребителя.



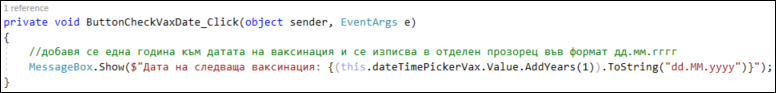
Фигура Метод за активиране на календар и бутон и извеждане на съобщение

Методът се задава като стойност на събитието CheckedChange на радио-бутона с име „radioButtonVaxYes“.

### Метод за изчисляване на дата за следваща имунизация

Чрез този метод, показан на Фигура 8, се изчислява датата за следваща имунизация, в случай, че е избрана опция „Да“ от полето „Имунизация“. Задава се като стойност на събитието *Click* на бутона с име „buttonCheckVaxDate“.

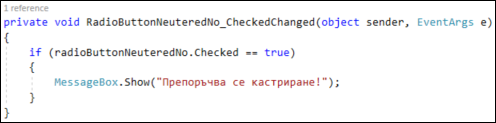
Изчисляването на датата за следваща имунизация става след като се добави една година към избраната дата от календата от контрола *DateTimePicker* с име „dateTimePickerVax“. След това датата се форматира в стринг с формат „дд.мм.гггг“, тъй като настройките по подразбиране на Windows ще покажат и часовете, а за целта на приложението от тях няма нужда.



Фигура 8 Метод за изчисляване и показване на дата за следваща имунизация

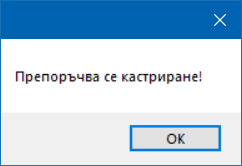
### Метод за показване на специално съобщение при избиране на опция „Не“ в поле „Кастриране“

Приложението има функционалност, която показва специално съобщение на потребителя, в случай, че е избрана опция „Не“ от поле „Кастриране“. На Фигура 9 е показан кода за тази функция, като програмата проверява дали е отбелязан радио-бутона „Не“.



Фигура 9 Метод за изписване на съобщение в нов прозорец

На фигура 10 е показано съобщението, което потребителят ще получи при избор на опция „Не“ в поле „Кастриране“.



Фигура 10 Специално съобщение при избор на опция „Не“ в поле „Кастриране“

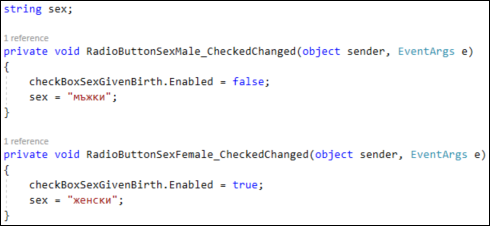
Методът се задава като стойност на събитието *CheckedChanged* на радио-бутона с име „radioButtonVaxNo“.

### Методи за деактивиране/активиране на *CheckBox*

Приложението има специална функционалност, която при избиране на опция „Мъж“ от поле „Пол“, деактивира *CheckBox* с текст „Раждала“, а при избиране на опция „Жена“ от същото поле, *CheckBox*-ът се активира отново.

Методът съдържа проста проверка дали радио-бутонът с име „radioButtonSexMale“ е избран. В този случай *CheckBox*-ът се деактивира. Вторият метод съдържа проста проверка дали радио-бутонът с име „radioButtonSexFemale“ е избраният. В този случай *CheckBox*-ът отново се активира.

Двата метода се задават като стойност на събитието *CheckedChanged* на двата радио-бутона, съответно с имена „radioButtonSexMale“ и „radioButtonSexFemale“.

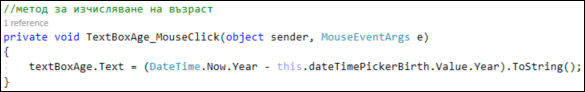


Фигура 11 Метод за деактивиране/активиране на CheckBox

На Фигура 11 е описан кода на тези два метода.

### Метод за изчисляване на възраст

Методът за изчисляване на възраст е описан на Фигура 12. Задава се като стойност на събитието *MouseClick* на текстовото поле с име „textBoxAge“.

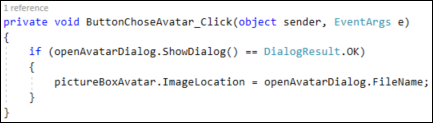


Фигура 12 Метод за изчисляване на възраст

Структурата на метода следва логиката, при която от годината на текущата дата се вади годината на раждане, а след това се преобразува в string.

### Метод за качване на снимка към досието

Методът, чрез който се качва снимка към досието, е описан на Фигура 13. Задава се като стойност на събитието Click на бутона с име „buttonChoseAvatar“.



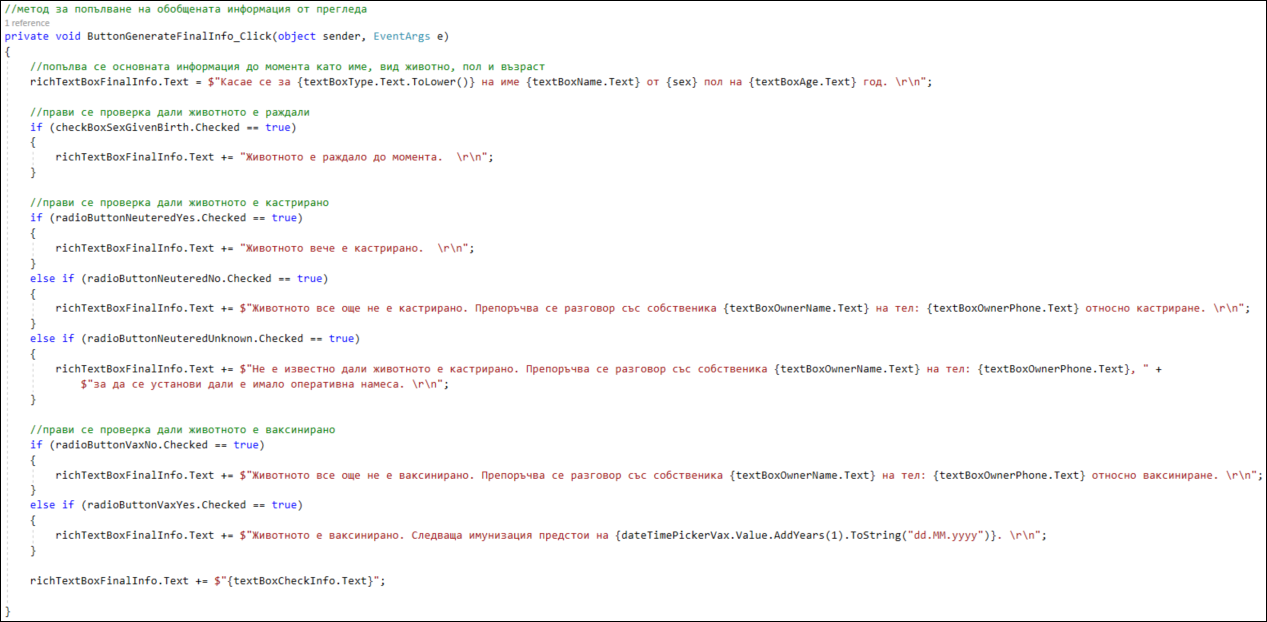
Фигура 13 Метод за качване на снимка към пациентското досие

Методът отваря диалогов прозорец, чрез който се навигира през файловата система на компютъра на потребителя. След като се избере файл за аватар, съответната снимка се визуализира в *pictureBox* контрола.

### Метод за попълване на текстово поле „Заключение от преглед“

Чрез този метод се попълва текстовото поле „Заключение от преглед“. Задава се като стойност на събитието Click на бутон с име „buttonGenerateInfo“. На Фигура 14 е представен целият код от този метод.

Правят се различни проверки за въведената информация, за да се изведе коректно обобщена информация. Методът добавя и текста, който е въведен от потребител при попълване на текстовото поле „textBoxCheckInfo“.



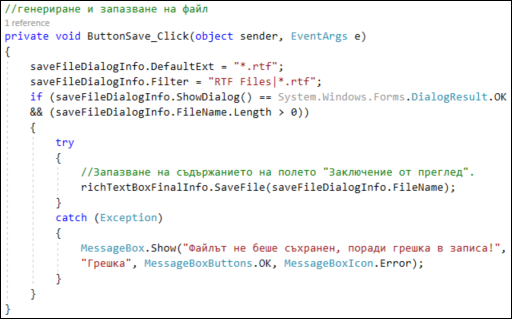
Фигура 14 Метод за попълване на текстово поле „Заключение от преглед“

### Метод за генериране на файл с текста от текстово поле „Заключение от преглед“

На Фигура 15 е показан кодът, който предоставя тази функционалност. Задава се като стойност на събитието Click на бутона с име „buttonSave“.

Кодът първо указва на метода, че файловете ще се запазват в .rtf формат. След което се извиква диалоговият прозорец, чрез който ще се навигира във файловата система на компютъра на потребителя.

Ако кодът не може да се изпълни, приложението ще покаже съобщение за грешка.



Фигура 15 Метод за генериране и запазване на файл

# Библиография

Визуални среди за програмиране, проф. д-р Иван Иванов, Фараго, София, 2012

www.stackoverflow.com