**НАЦИОНАЛНА ПРОГРАМА**

**„ОБУЧЕНИЕ ЗА ИТ КАРИЕРА“**

**Проектна разработка за**

**каталог за книги**

Документация

*Съставена от*

Димитър Чакъров, Иван Иванов

11 клас, МГ „Акад. Кирил Попов“

06.04.2019

**Съдържание**

1. [Увод 3](#_Toc5437791)
2. [Използвани технологии 3](#_Toc5437792)

[2.1. .Net Framework **Error! Bookmark not defined.**](#_Toc5437793)

[2.2. C# **Error! Bookmark not defined.**](#_Toc5437794)

[2.3. WindowsForms **Error! Bookmark not defined.**](#_Toc5437795)

[2.4. Entity Framework **Error! Bookmark not defined.**](#_Toc5437796)

1. [Софтуерната архитектура 4](#_Toc5437797)
2. [Data Layer 5](#_Toc5437798)
3. [Service Layer 5](#_Toc5437799)
4. [Presentation Layer 5](#_Toc5437800)
5. [Бъдещо развитие 5](#_Toc5437801)
6. [Заключение 6](#_Toc5437802)
7. [Използвана литература 6](#_Toc5437803)

# Увод

Настоящата разработка имплементира интерактивен каталог за книги[[1]](#footnote-1). Целта ни е изготвянето на програмен продукт, който служи за намиране, добавяне, изтриване и актуализиране на информацията за дадена книга в база данни. Добавена е функционалност за въвеждане на описанието на книга, както и маркирането на книга като *прочетена*, *харесана* или *любима.*

Следващата секция в документацията на разработката представлява разширено описание на използваните технологии. Третата секция излага архитектурата на каталога за книги. Следващите три секции показват съответно *Data Layer*, *Service Layer* и *Presentation Layer* на архитектурата на каталога за книги. Последните две секции са бъдещо развитие и заключение. Добавен е списък на използваната литература.

# Използвани технологии

Настоящата секция описва използваната технология.

**2.1. .NET Framework**

.NET Framework е софтуерна платформа, създадена и поддържана от Майкрософт. Предоставя програмен модел, библиотека от класове и среда за изпълнение на написан специално за нея програмен код. .NET приложенията се пишат на езици от високо ниво (C#, Visual C/C++) и се компилират до платформено-независим междинен език, наречен CIL (*Common Intermediate Language*).

**2.2. C#**

C# е обектно-ориентиран език за програмиране, разработен от Microsoft, като част от софтуерната платформа .NET Framework. Стремежът още при създаването на C# езика е бил да се създаде един прост, модерен, обектно-ориентиран език с общо предназначение. Основа за C# са C++ и Java. Той е проектиран да балансира мощността на C++ с възможностите за бързо разработване на Java.

**2.3. WindowsForms**

WindowsForms е графична библиотека от класове в състава на Microsoft .NET Framework, която предоставя платформа за писане на клиентски приложения за настолни компютри, лаптопи и таблети. Изработените с помощта на Windows форми приложения се задействат при настъпване на определено събитие или при определено действие от страна на потребителя, като например попълване на текстово поле или посочване и щракване на бутон.

**2.4. Entity Framework**

Entity Framework е отворена обектно-ориентирана mapping framework (ORM) за ADO.NET. Преди 6 версия е част от Microsoft .NET Framework, но след това е отделена от .NET framework. Entity Framework е множество от технологии, поддържащи работа и управление на информационно ориентирани приложения.

# Софтуерната архитектура

Софтуерният продукт използва .Net Framework v. 4.7.2 и Visual Studio 2017. Microsoft SQL Server Express 2017 е избраната система за управление на бази от данни, макар че моделът ѝ е изработен с метода Code-First на Entity Framework. Приложението е разделено логически на 3 слоя (layers):

* + - 1. Информационен слой
      2. Логически слой
      3. Презентационен слой

# Data Layer

Използвахме метода Code-First на Entity Framework за създаването на базата данни. Тя е локална и съдържа в себе си само една таблица, която съдържа цялата информация за всяка една книга. Използвахме Microsoft SQL Server 2017 за връзка между програмата и базата данни. Всеки ред от таблицата съдържа поле за име, автор, описание, коментар и дали е прочетена, харесана или любима дадена книга.

# Service Layer

Имлементирали сме ниво осъществяващо връзката между презентационния слой и информационния. В него са имплементирани всички функционалности за работа с данните от базата данни, така че да бъдат викани от потребителския интерфейс. Класът BookLogistics служа за wrapper на тези функционалности.

# Presentation Layer

Презентационното ниво се състои от осем функционалности.

* + - 1. Търсене на книга
      2. Добавяне на книга
      3. Промяна на информацията за книга
      4. Изтриване на книга
      5. Показване на списъка с всички книги
      6. Показване на списък с всички прочетени книги
      7. Показване на списък с всички харесани книги
      8. Показване на списък с всички любими книги

# Бъдещо развитие

Следните функционалности могат да се интегрират в софтуерния продукт, като ще ги степенуваме по трудност, започвайки от най-лесните:

* + - 1. Панел за маркиране на книга като прочетена, харесана, любима, като няма промяна на нищо друго
      2. Търсене на книги по име или автор, а не само по *id.*

# Заключение

Създадохме приложение, което служи като каталог за книги. Съдържа различни функционалности за работа с базата данни от книги.

# Използвана литература

* <https://stackoverflow.com/>

1. Имплементацията може да бъде открита на https://github.com/dimitarch/book-catalogue [↑](#footnote-ref-1)