



Пловдивски университет „Паисий Хилендарски”  
Факултет по математика и информатика

---

# Курсова работа

По дисциплина „Дизайн на приложения за мобилни устройства“

На тема: „Филмова база данни“

Изготвил: Александър Йончев

Специалност: Софтуерни

технологии и дизайн

Факултетен номер: 1701681058

Проверил:

/ доц. д-р Н. Касъклиев/

## Съдържание

1.	Увод .....	3
1.	Използвани технологии и библиотеки .....	3
2.	Изисквания към приложението .....	3
2.	Потребителски инструктаж .....	3
3.	Имплементация.....	6
1.	NavPage.xaml .....	7
2.	MovieListPage.xaml .....	7
3.	MovieDetailsPage.xaml .....	7
4.	MovieCreateEditPage.xaml .....	8
5.	GenresPage.xaml .....	8
4.	Заключение .....	8
5.	Библиография .....	8

## 1. Увод

Приложението представлява база данни с филми. В него потребителят може да добавя филми, да редактира, да разглежда, да търси и да филтрира по жанр.

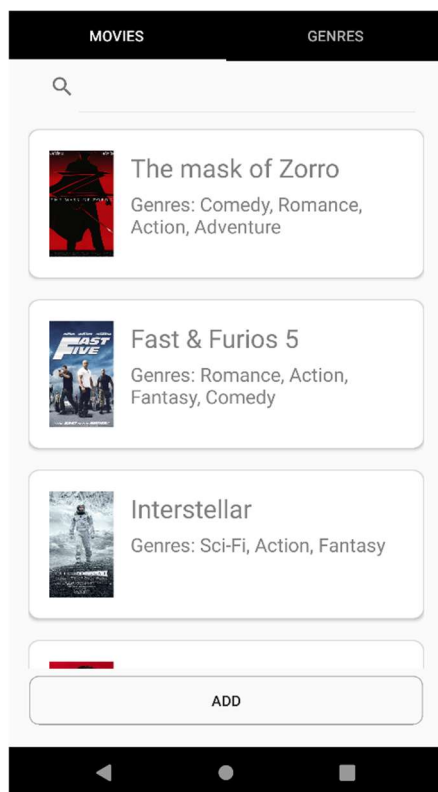
### 1. Използвани технологии и библиотеки

- Visual Studio
- C#
- Xamarin.Forms - Технология позволяваща споделянето на огромно количество код за различните мобилни платформи – Android, iOS, Windows Phone, правеща създаването на мобилни приложения изключително лесно, посредством технологията XAML

### 2. Изисквания към приложението

- Да съдържа поне 4 отделни екрана
- Да са смислово свързани все едно приложението е работещо
- Да са адаптивни и добре изглеждащи

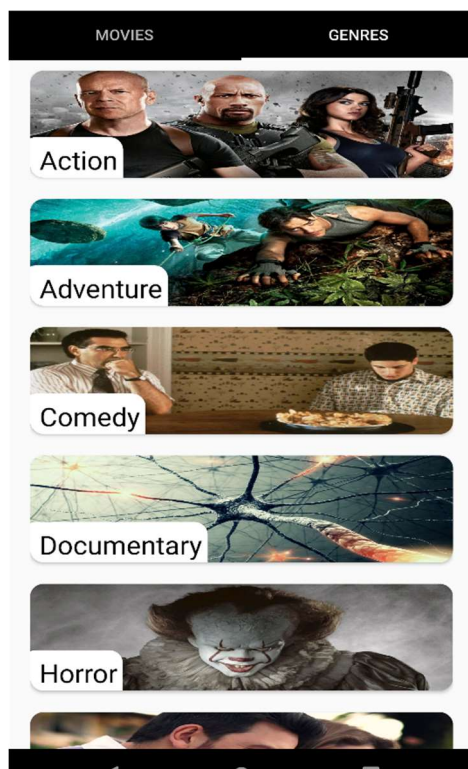
## 2. Потребителски инструктаж



Фигура 1 Основен екран - списък с филми

При зареждане на приложението се отваря основния екран, който е списък с филми, като за всеки филм са визуализирани неговата корица, името и жанровете му. Чрез полето за търсене над списъка, може да се търси по заглавие на филм. Отдолу, под списъка с филми

е поставен бутон, който ни отвежда към страница за добавяне на филм. Отгоре ясно се отличава навигационната лента, чрез която бихме могли да посетим и друга основна страница – страницата с жанровете.

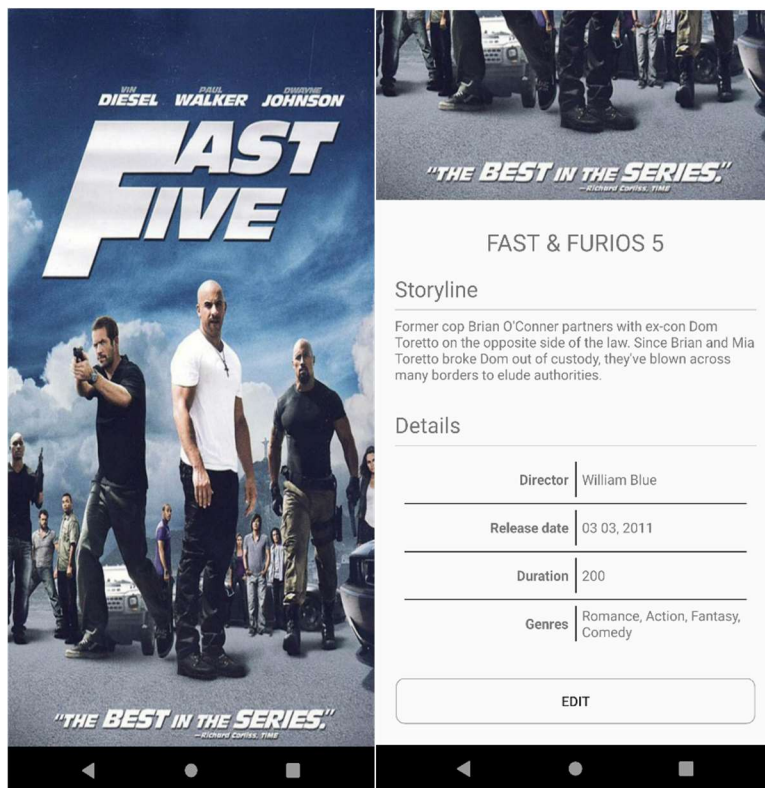


Фигура 2 Екран със списък с жанрове

На Фигура 2 Екран със списък с жанрове Фигура 2 е показан екрана със списъка с жанрове, където при натискането на някой от жанровете, се отваря основната страница и са изведени всички филми, които са от този жанр.

Връщайки се обратно на основния екран, при натискане на кой да е от филмите, се отваря екран с детайли за филма, който е изобразен на Фигура 3. На тази страница, в горната страница е поставена корицата на филма, а отдолу е поместена информацията.

Информацията включва заглавието, сюжета както и таблица с детайли за филма като име на директор, продължителност и т.н. Най-отдолу е поместен бутон, който ни отвежда към страница за промяна на информацията за филма. **Важно е да се спомене, че понеже бутонът за връщане назад е махнат е добавена функционалност, която улавя ‘swipe’ надясно и ни връща към предишната страница.**



Фигура 3 Екран с детайли за филм

SELECT A PICTURE

Name  
FAST & FURIOS 5

Description  
Former cop Brian O'Conner partners with ex-con Dom Toretto on the opposite side of the law. Since Brian and Mia Toretto broke Dom out of custody, they've blown across many borders to elude authorities.

Director Name  
William Blue

Duration  
200

Release Date  
3/3/2011

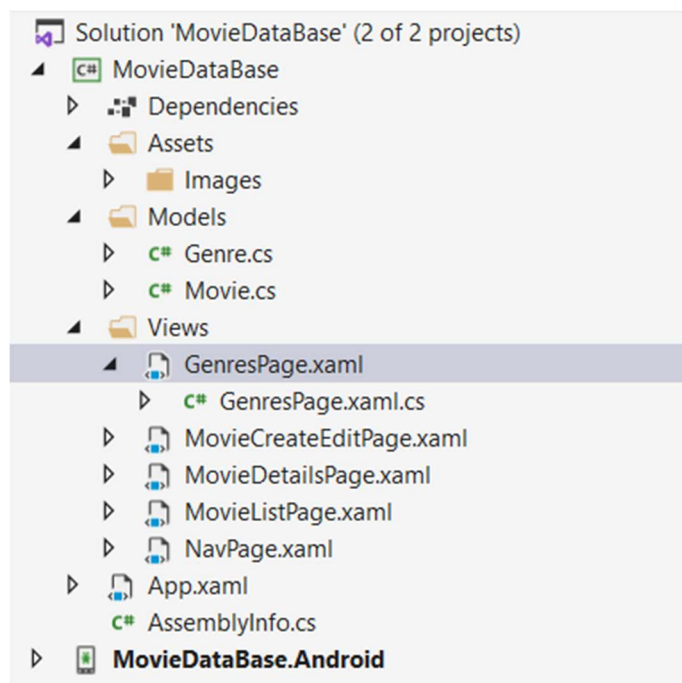
Genres  
☒ Romance  
☒ Action  
☒ Adventure  
☐ Horror  
☐ Mystery  
☐ Documentary  
☒ Comedy  
☐ Fantasy  
☐ Sci-fi  
☐ Horror

SAVE

Фигура 4 Екран за промяна информацията на филм

На Фигура 4 е показан екранът за промяна на информацията за даден филм. Този екран съдържа един бутон за избиране на снимка, множество текстови полета, бутони тип 'checkbox', с които да се изберат жанровете на филма, както и бутон за запазване на промените. Екранът за добавяне на филм е изцяло идентичен, единствената разлика е, че не съдържа никакви попълнени полета.

### 3. Имплементация



Фигура 5 Структура на проекта

На Фигура 5 е представена структурата на проекта. Както споменах по-горе, проекта е правен с технологията разработена от **Майкрософт**, *Xamarin.Forms*. В solution файла, се съдържат два проекта – *MovieDataBase*, *MovieDataBase.Android*. Вторият проект съдържа едно *MainActivity* от което се зарежда *App.xaml.cs* класа и се стартира приложението. Идеята на втория проект е да съдържа специфичен код за платформата *Android*, като по този начин основната част от кода стои в първия проект и би могъл да се сподели с различните платформи. Оттук насетне фокусирам вниманието към втория проект, понеже той е основен.

#### Основни директории и тяхното предназначение

- **Assets** – Директория, в която се съхраняват статични файлове, в случая единствено снимки, които могат да бъдат използвани от различните платформи.
- **Models** – Директория съдържаща домейн моделите на приложението.
- **Views** – Тук се съдържат екраните/страниците на приложението структурата и стиловете, на която са описани с технологията създадена от Microsoft – **XAML**. Заедно с тях, тук е и съпътстващия ги “code-behind”, където се съдържа кода, който обработва различни събития и свързва UI с данните.

Имената на всеки от екраните е достатъчно описателен и се вижда за какво служи. Сега ще обясня за всеки екран как реализира изобразените по-горе потребителски интерфейси и където е необходимо, ще поставям код.

### **1. NavPage.xaml**

Занимава се със създаването на навигацията, от която могат да се изберат страницата за филми и страницата за жанрове.

### **2. MovieListPage.xaml**

За създаването на този екран са използвани един основен StackLayout елемент, който е аналог на Linear Layout, който е с вертикална ориентация и в него са поместени SearchBar елемент, ListView, в който се съдържа списъка с филми, а най-отдолу стои бутон за добавяне на нов филм. Също така тук е добавен и един елемент за разпознаването на жестове. На Фигура 6 е изобразен кода, за създаването на дизайна на всеки елемент от списъка – използван е Frame, чрез който са зададени облите ръбове на елемента, сянката и др. В него има един StackLayout, който е с хоризонтална ориентация и съдържа една снимка – корицата на филма и още един StackLayout, в който са поместени два Label елемента за изобразяване на заглавието

```
<Frame BackgroundColor="White" CornerRadius="10" HasShadow="True" BorderColor="LightGray"
Margin="0, 10, 0, 10" IsClippedToBounds="True">
  <StackLayout Orientation="Horizontal" VerticalOptions="Center">
    <Image Source="{Binding Picture}" WidthRequest="60" HeightRequest="100" Aspect="AspectFill" />
    <StackLayout>
      <Label Margin="10, 0, 0, 0" Text="{Binding Name}" FontSize="Title" VerticalTextAlignment="Center"
TextColor="Gray" />
      <Label Margin="10, 0, 0, 0" Text="{Binding Genres, StringFormat='Genres: {0}'}" FontSize="Medium"
VerticalTextAlignment="Center" TextColor="Gray" />
    </StackLayout>
  </StackLayout>
</Frame>
```

Фигура 6 Код за дизайн на елемент от списъка с филми

и жанровете.

### **3. MovieDetailsPage.xaml**

Тук основният елемент е StackLayout, който е сложен в ScrollView елемент, който прави съдържанието скролируемо. Тук съдържанието отново е поместено предимно в StackLayout елементи, като за създаването на таблицата в долната част е използван GridLayout. BoxView елементът е използван за създаването на вертикална или хоризонтална черта. Чрез него се получава усещането, че на страницата има таблица, а реално това са редове и колони, разделени с хоризонтални и вертикални черти.

#### 4. *MovieCreateEditPage.xaml*

И тук отново е използван предимно StackLayout елемент, в който са поместени множество Label и Entry елементи. Entry елементът се използва за въвеждане на текст, подобно на TextBox в другите технологии. За разполагането на жанровете в две колони отново е използван GridLayout, в който за всеки отделен е използван StackLayout съдържащ в себе си Label и самия CheckBox елемент, както е показано на Фигура 7.

```
<StackLayout Grid.Column="1" Orientation="Horizontal" VerticalOptions="Center">
    <CheckBox IsChecked="False" Color="Black" />
    <Label Text="Documentary" VerticalOptions="Center" />
</StackLayout>
```

Фигура 7 Изобразяване на жанр

#### 5. *GenresPage.xaml*

Тук по-интересното е, че за всеки елемент от списъка с жанрове е използван RelativeLayout. Снимката е зададена да заема целия RelativeLayout елемент, а Label елемента съдържащ името на жанра, е позициониран в левия долен ъгъл на RelativeLayout елемента.

### 4. Заключение

Приложението съдържа основните за такъв тип приложение екрани, като биха могли да се добавят и множество други, както и множество различни начини за филтриране. В бъдеще биха могли да се добавят и екрани визуализиращи актьори, както и информация за тях. Само малка част от приложението е работеща, свързана с навигацията и визуализираните на статично вкарани филми.

### 5. Библиография

- Microsoft. (н.д.). *Adding a swipe gesture recognizer*. Извлечено от <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/gestures/swipe>
- Microsoft. (н.д.). *Xamarin.Forms Frame*. Извлечено от <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/layouts/frame>
- Microsoft. (н.д.). *Xamarin.Forms Grid*. Извлечено от <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/layouts/grid>
- Microsoft. (н.д.). *Xamarin.Forms Layouts*. Извлечено от <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/controls/layouts>
- Microsoft. (н.д.). *Xamarin.Forms RelativeLayout*. Извлечено от <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/user-interface/layouts/relative-layout>