**Национальный исследовательский университет**

**«Высшая школа экономики»**

Факультет Бизнеса и менеджмента

***Пояснительная записка***

***к итоговому проекту по дисциплине***

***«Программирование»***

**Выполнили:**

студенты группы ББИ142

Шахов Михаил,

Байрамова Камила,

Ошанина Наталья

**Руководитель:**

Ефремов Сергей Геннадьевич

Москва, 2016 год

**1. Описание проекта**

Разработанное приложение Book Organizer помогает пользователю в создании читательского дневника. Основные возможности программы:

* Хранение списка прочитанных книг;
* Создание списков начатых и запланированных книг;
* Загрузка тематических списков книг из веб-сервиса и добавление их в свою библиотеку.

Все книги хранятся в общей базе данных. При работе со списками доступно добавление комментариев и аннотаций, дат начала и окончания чтения, выставление оценок книгам. Программа также позволяет редактировать и удалять записи.

**2. Адрес репозитория:** <https://github.com/Nattican/BookOrganizer>

**3. Участники команды:**

Шахов Михаил: разработка базы данных и функциональной логики приложения.

Байрамова Камила: создание пользовательского интерфейса.

Ошанина Наталья: загрузка веб-данных, написание тестов.

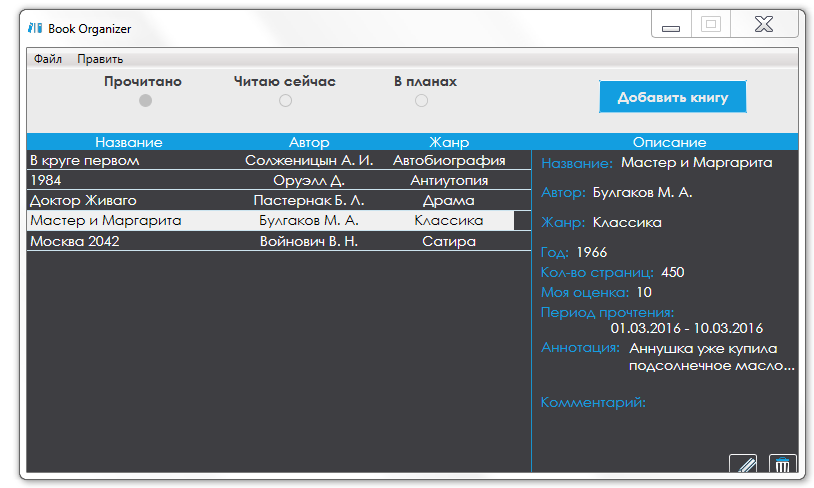
**4. Описание классов приложения**

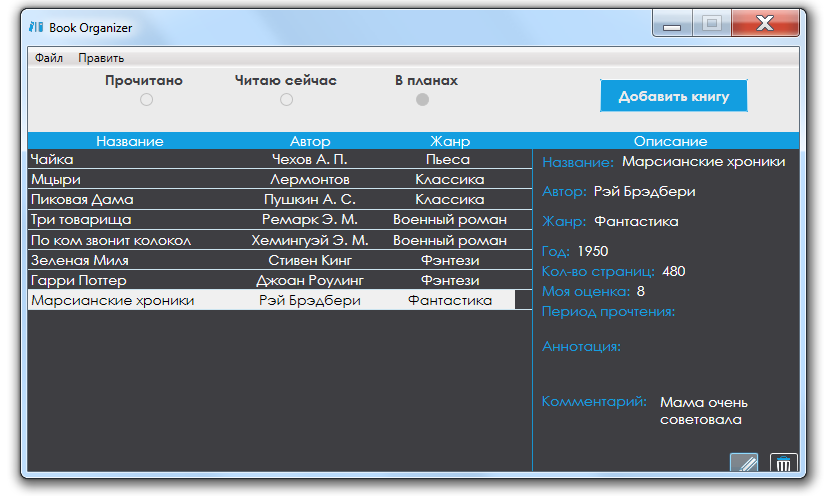
* Классы MainView, AddBookView – представляют графический интерфейс соответствующих окон.
* Класс MainViewModel – унаследован от абстрактного класса ViewModelBase, реализует логику MVVM шаблона. Содержит команды для добавления/удаления книг из базы данных, открытия различных списков, отмены/возврата последних действий.
* Класс AddBookViewModel – унаследован от абстрактного класса ViewModelBase, реализует логику MVVM шаблона. Содержит объект Book с полями информации о книге, логику создания новой книги и добавления её автора и жанра в базу данных (в случае, если они упоминаются впервые).
* Класс Context – унаследован от DbContext, хранит сущности базы данных и устанавливает соединение.
* Классы Book, Genre, Author – соответствуют таблицам базы данных.
* Класс ApiWindow – содержит логику загрузки данных из веб-сервиса.
* Классы API\_Data, Books и Book – предназначены для десериализации JSON-данных, загруженных с помощью API.
* Класс Tests – содержит тесты к приложению.

Приложение разработано с помощью MVVM шаблона.

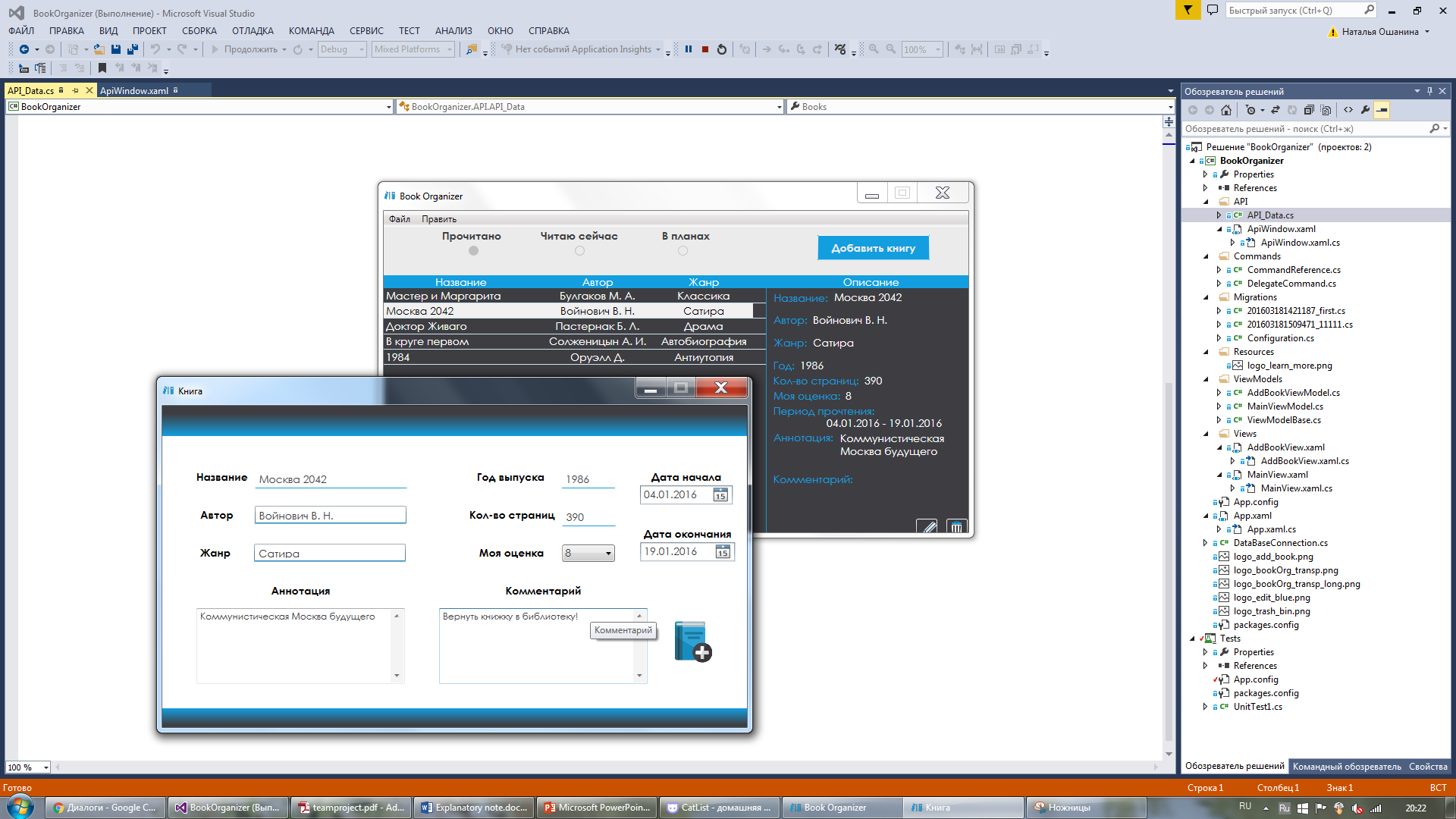
**5. Интерфейс программы**

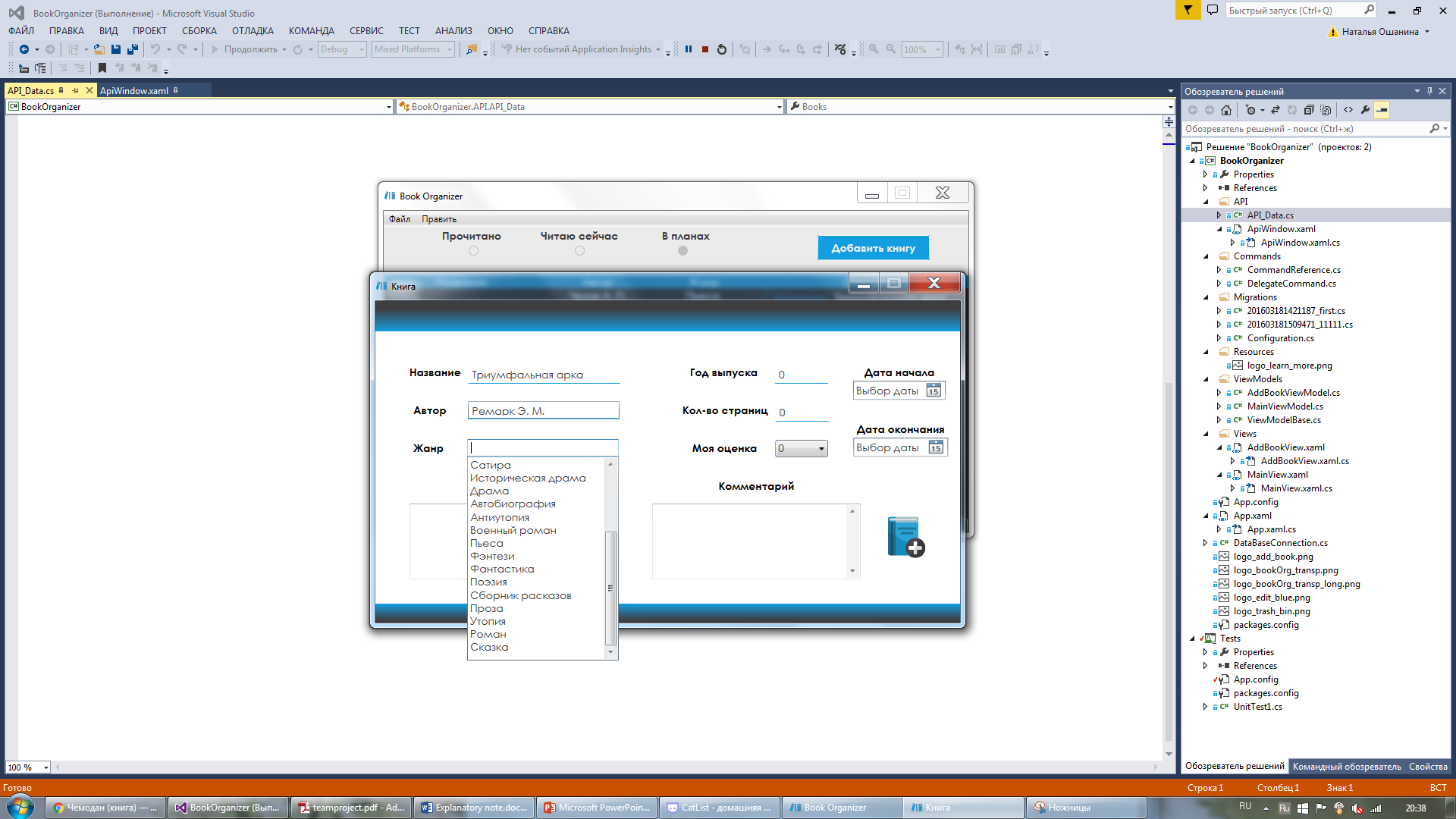
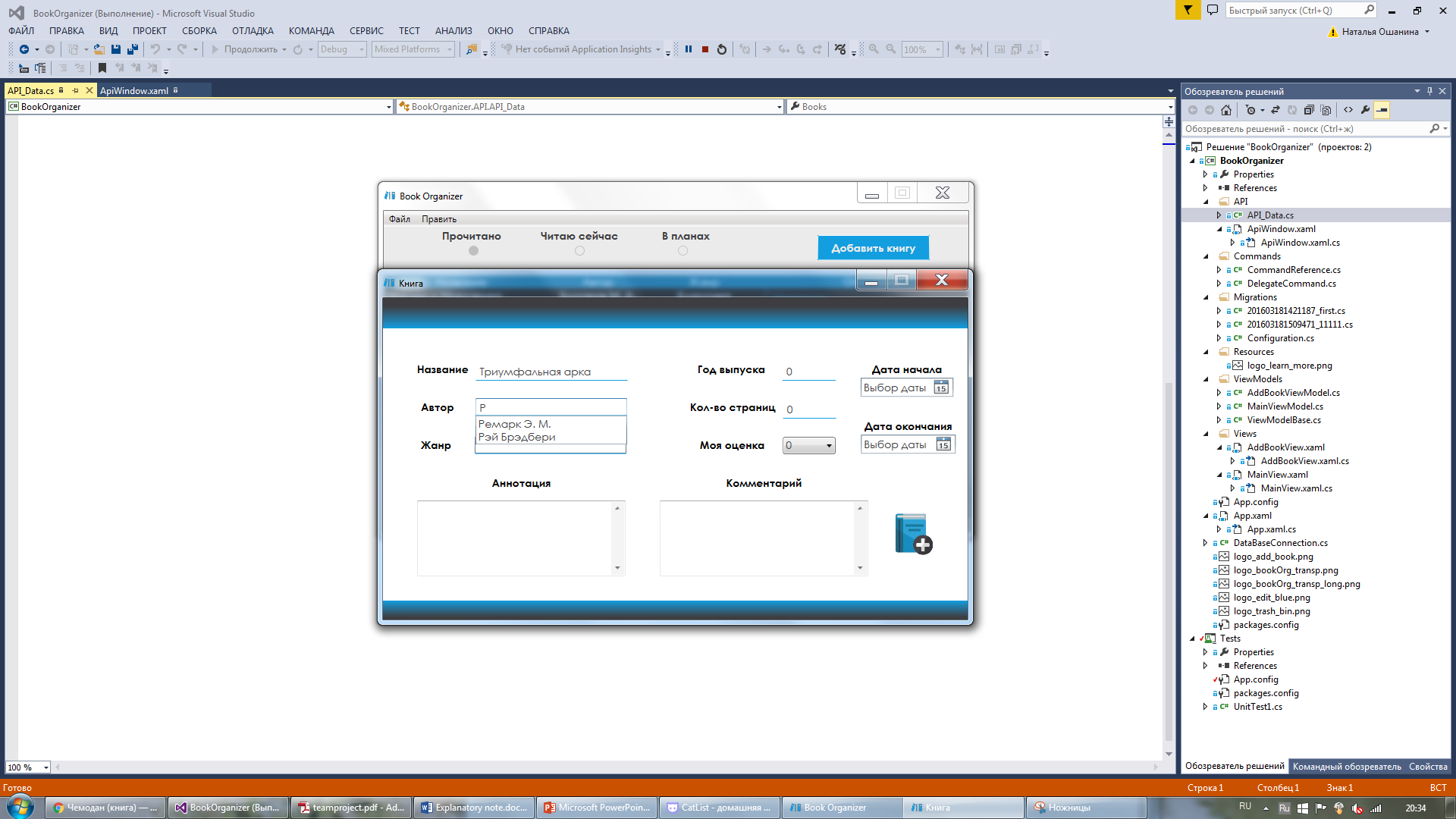
Основное окно, позволяющее переключаться между тремя списками:



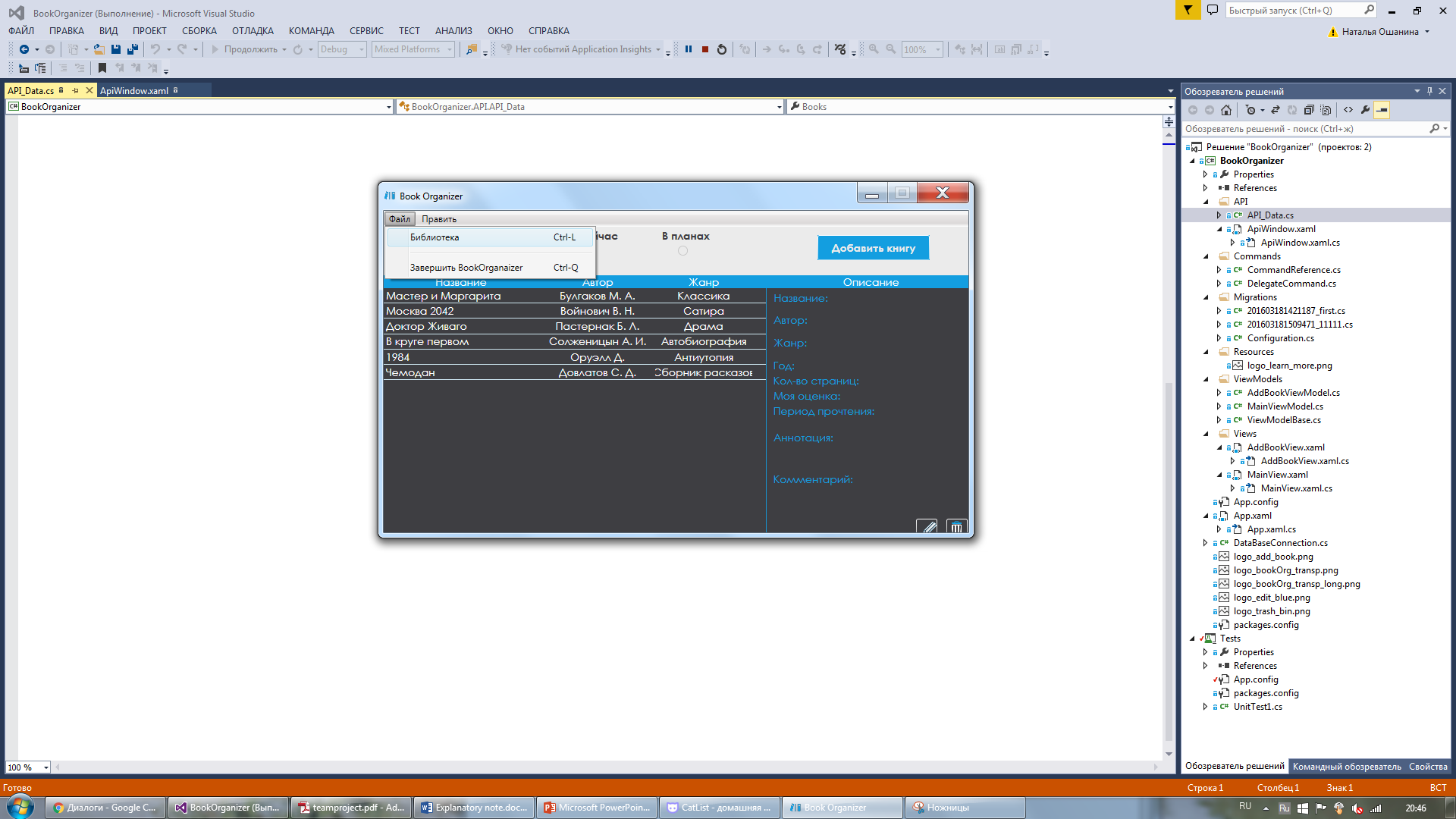


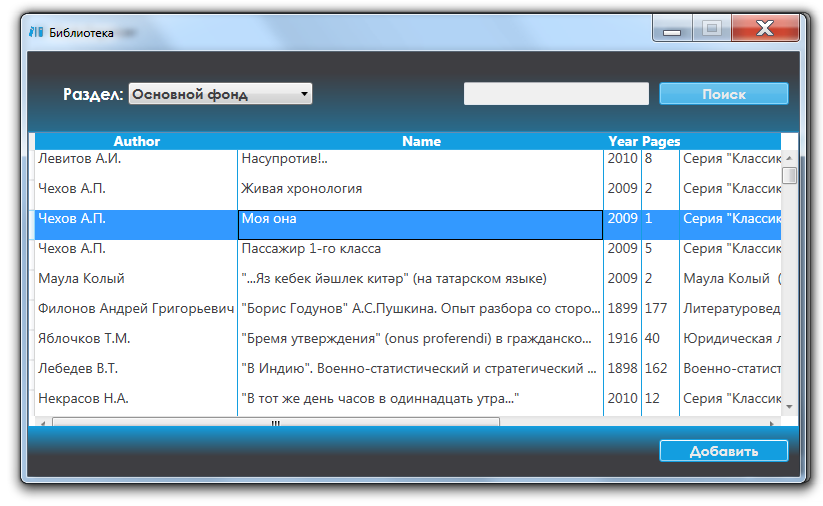
Редактирование существующей книги:



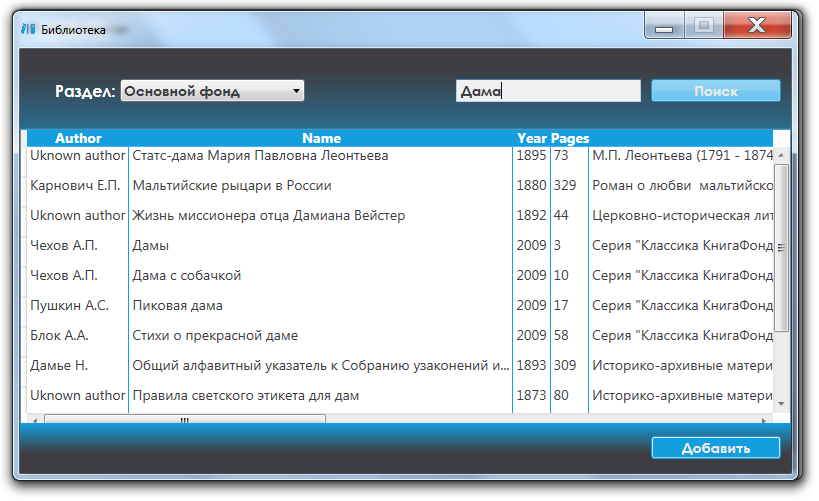
Добавление новой книги с всплывающими подсказками уже известных программе авторов и жанров: 

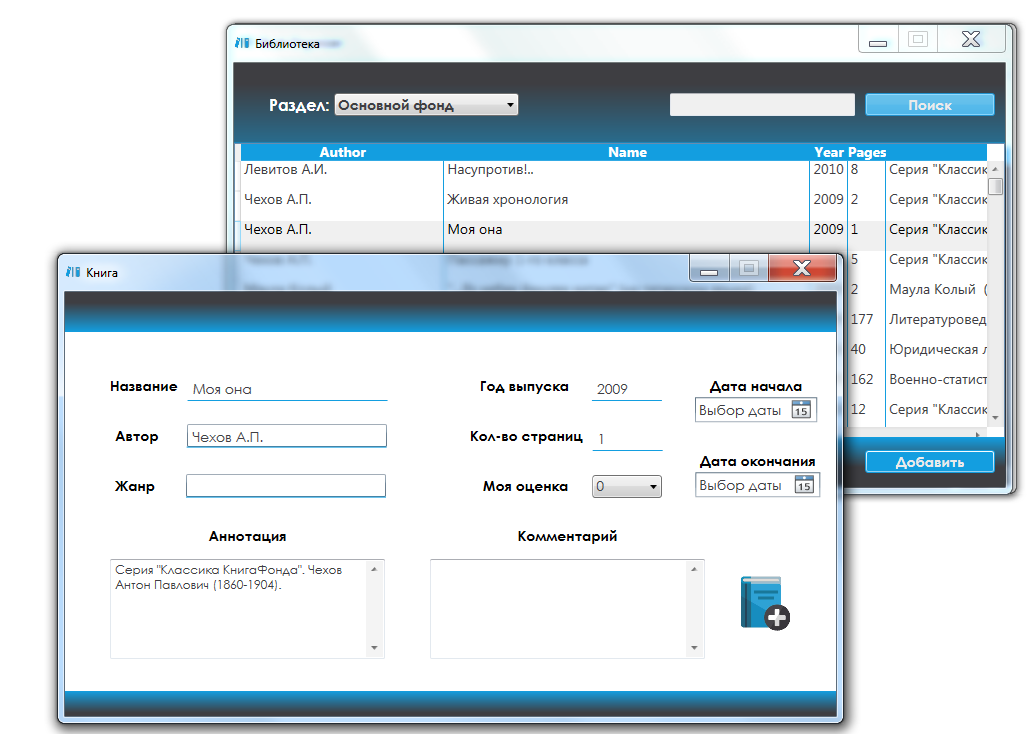
Списки книг, загружаемые из веб-сервиса:





Поиск книги по названию:



Добавление книги в основную базу: 

Приятного использования!