

**Технически университет – Варна**

**Факултет по**

**изчислителна техника и автоматизация**

# Курсов проект по

# „Софтуерни технологии“

Изработил:

Стефан Генчев Генев

Фак. № 19623318

Специалност: КСТ-задочно, 4-ти курс, 1а група

# Съдържание

* Задание на проекта
* Операции с обяви за книги
* Операции с потребители
* Функционални изисквания
* Проектиране на системата
  + База данни
  + GUI
  + Бизнес логика
  + Диаграми
* Структура на проекта
* Реализация на системата
* Ръководство за потребителя
  + Автентикация
  + Начална страница
  + Детайли за книга
  + Footer
  + Регистрация
  + Вход
  + Logout
  + Преглед на издателства
  + Добавяне / Редакция на издателство
  + Преглед на автори
  + Добавяне / Редакция на автор
  + Добавяне на книга
  + Профилна страница
  + Редактиране на книга
  + Читатели
  + Заети книги

# Задание на проекта

Проектът представлява система за управление на библиотека. Системата предоставя начини за съхранение и обработка на данни за читатели и за заетите от тях книги.

**Операции с обяви за книги:**

* Създаване, редакция, преглед и изтриване на записи за Автори - тези действия се извършват единствено от потребител с роля администратор;
* Създаване, редакция, преглед и изтриване на записи за Издателства - тези действия се извършват единствено от потребител с роля администратор;
* Създаване, преглед, редакция и изтриване на обяви за книги, като тези действия се извършват единствено от потребител с роля администратор;
* Заемане на книга – записване на заявка за вземане на книга и намаляне на наличния брой копия от дадената книга;
* Връщане на книга – записване в записа за заявката, че е върната книгата и увеличаване на броя налични копия на дадената книга.

**Операции с потребители:**

* Регистрация на нов потребител с роля администратор – това се извършва автоматично, като при първоначална инициализация на системата се извършва проверка дали съществува такъв потребител. Ако не съществува такъв се създава нов с входни данни предварително дефинирани в така нареченият “User Secrets” файл дефиниран извън проекта в JSON формат
* Регистрация на нов потребител с роля читател – всеки потребител може да си направи профил, за да преглежда, заема и връща книги

# Функционални изисквания

Потребители с роля администратор извършват управлението по системата, като на тях е позволено да добавят, редактират и изтриват записи за книги и потребители, както и да преглеждат отчет със заети книги от читатели.

Добавянето на книги се осъществява чрез попълване на заглавие, описание, автор, издателство, цена, година на издаване, снимка на корица и налични копия във формуляр с осигурена валидация против невалидни данни. Авторът и издателството се избират от наличните в базата записи за тях, като и те могат да се редактират от администраторската роля чрез потребителският интерфейс.

Всички потребители, независимо от ролята им, могат да преглеждат книги, които имат налични копия, както и да ги заемат и връщат. В потребителската страница на всеки потребител са показани неговите данни, профилна снимка и заявки за книги и дали са върнати.

# Проектиране на приложението

## **База данни**

* + За база данни се използва **Microsoft SQL Server**
  + Данновият слой и комуникация с базата данни са реализирани с **Entity Framework Core**

## **Graphical User Interface**

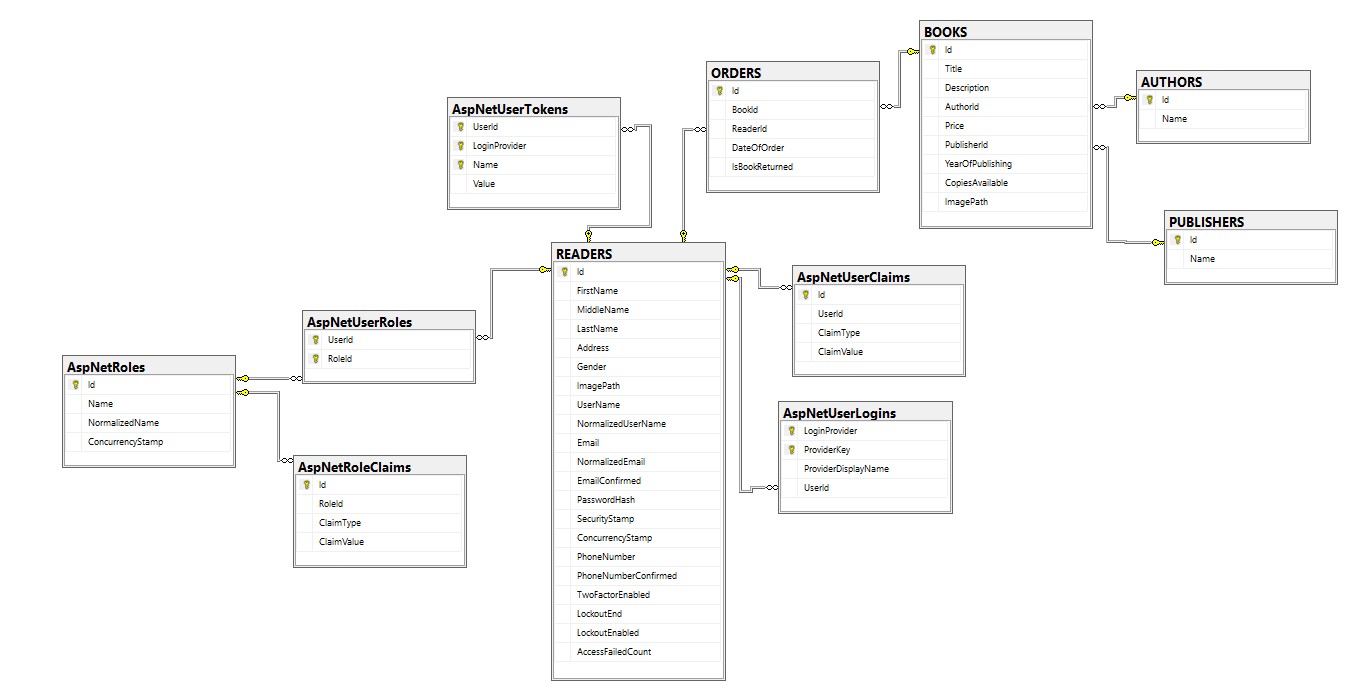
* Bootstrap библиотека с предоставени HTML и CSS стилове и шаблони за дизайн на страниците

**Бизнес логика**

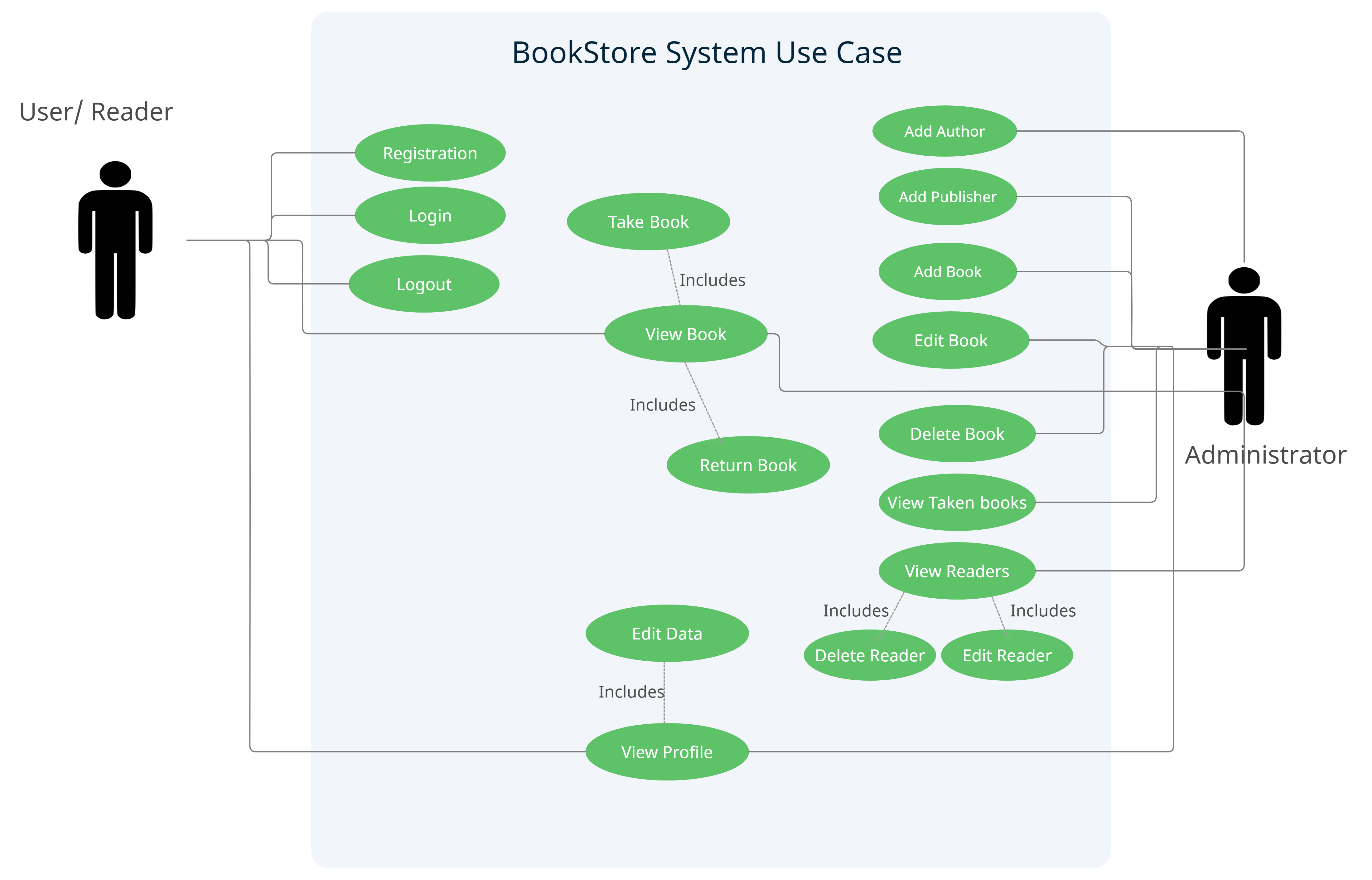
* Бизнес логиката на системата е пакетирана в различните контролери

## **Диаграми**

Релационна диаграма



Управление на системата



# Структура на приложението

# C:\Users\MSI\Downloads\Untitled Workspace (1).png

**Моделът (Model)** - данните върху които се работи и с които се извършват операции като промяна, добавяне и изтриване. Този код може да съдържа необработени данни или да идентифицира основните компоненти на програмата. В базата данни моделите биват представени като таблици

**Изгледът (View)** - Служи за определяне как потребителят вижда и взаимодейства с информацията.

**Контролерът (Controller)** – Съдържа бизнес логиката на приложението. Определя от къде ще се взимат входните данни, и осъществява комуникация с модела и изгледа

# Реализация на системата

Използва се **NuGet** за добавяне и използване на пакети. Съдържа списък от всички инсталирани пакети, чрез който има възможност за възстановяване и актуализиране на тези пакети.

Класовете в проекта са разделени в директории, които дефинират различните слоеве:

* wwwroot - съдържа директории със статични файлове - css, снимки, библиотечни файлове;
* Controllers - съдържа всички контролер класове;
* Migrations - съдържа всички миграции, които поддържат схемата на базата данни в синхрон с EF Core модела;
* Models - съдържа всички модели;
  + ViewModels - съдържа класове, които съхраняват данни от два или повече модела в един общ клас за изобразяване;
* Repository - съдържа всички класове и методи, които комуникират с базата данни;
* SQL Scripts – SQL скриптове за първоначално добавяне на данни
* Utils - съдържа помощни класове;
* Views - съдържа всички изгледи

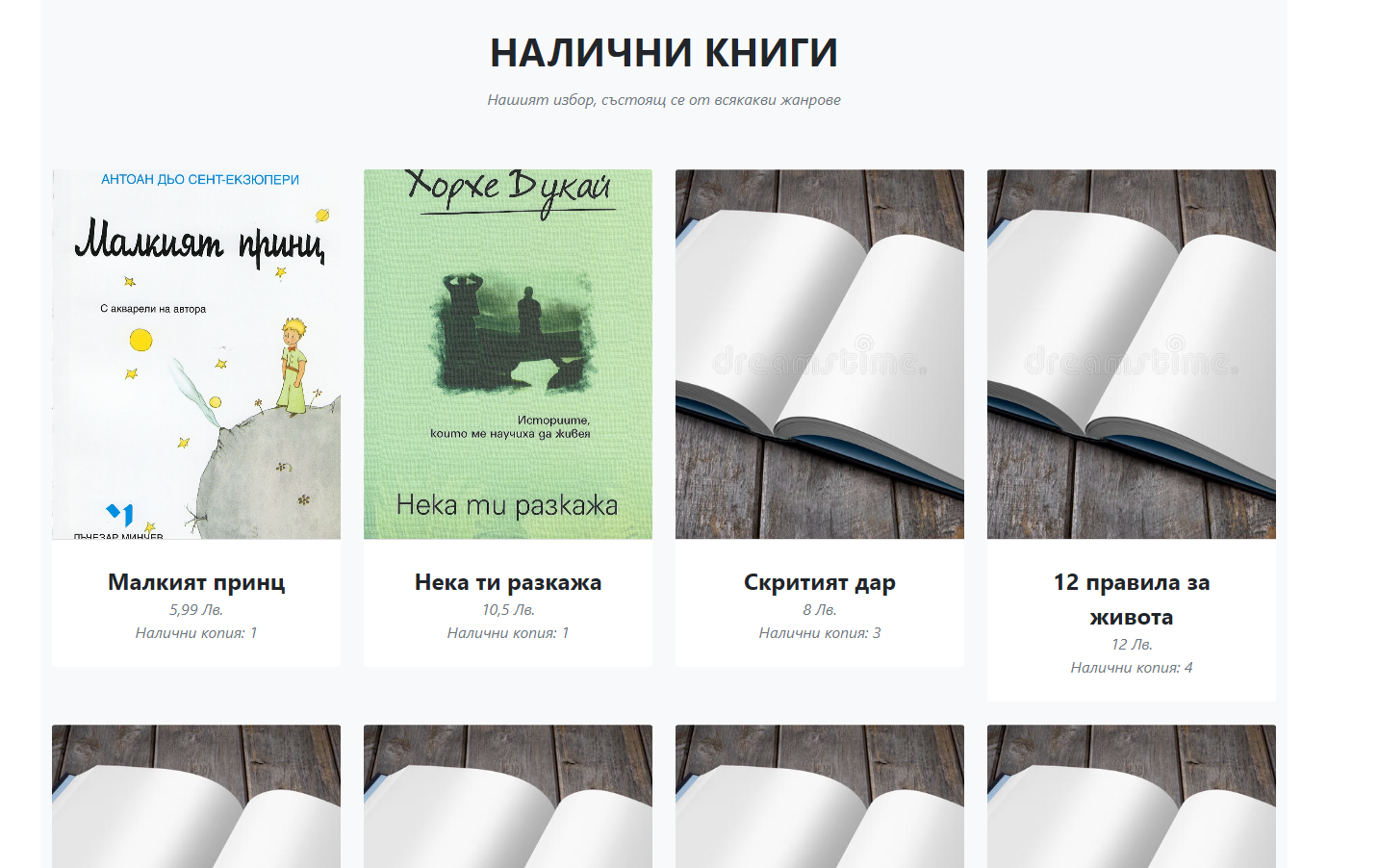
# Ръководство за потребителя

## **Автентикация**

Ако потребителят опита да достигне ресурс, за който му е нужна автентикация, ще бъде прехвърлен към логин секцията.

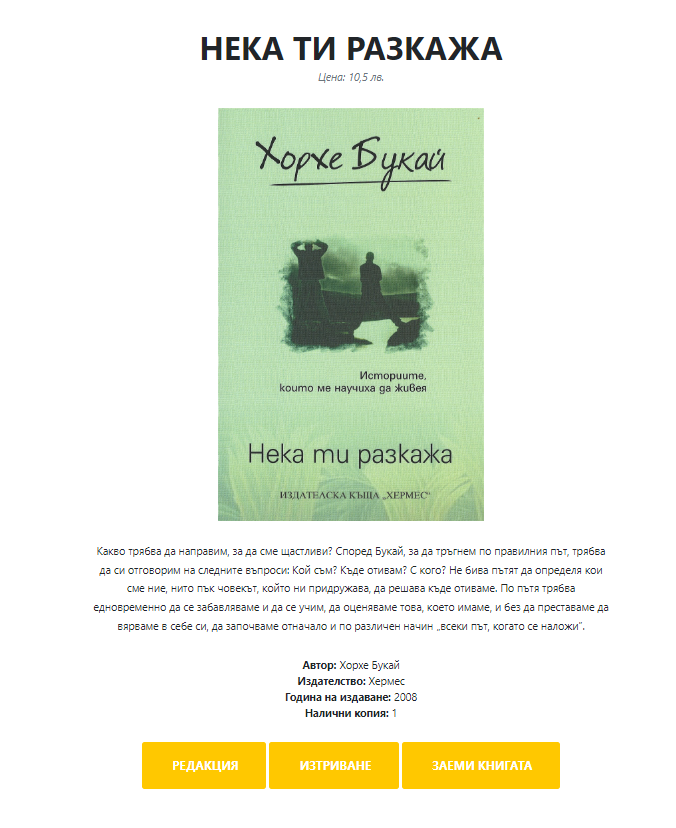
## **Начална страница**

На началната страница са наличните книги, които имат поне 1 наличен брой копия

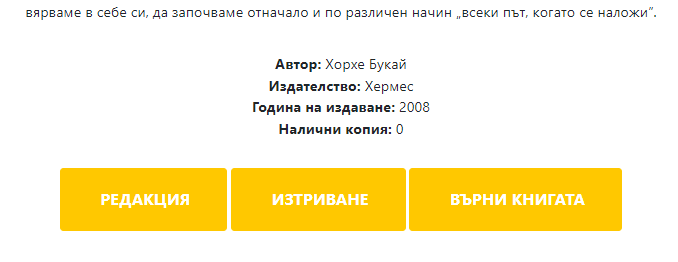
**** ****

## **Детайли на книга**

При клик върху книга се отваря страница с детайлна информация за книгата. В детайлната страница има бутони за заемане и връщане на книга ако е вече заета. Ако потребителят е с роля администратор се извеждат и бутони за редакция и изтриване. Ако потребителят не е логнат се извършва пренасочване към страница за вход на потребител.



След заемане на книга се намалят наличните копия и се извежда бутон за връщане:



**Header**

Съдържа всички налични функционалности спрямо ролята на потребителя

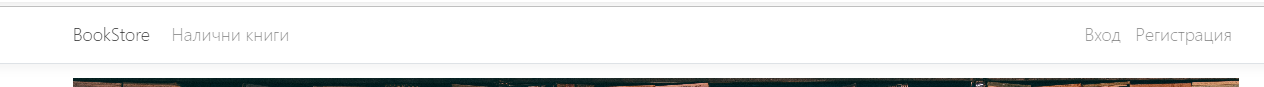
Роля “Читател”:



Роля “Администратор”:



Преди вход/регистрация:



**Footer**

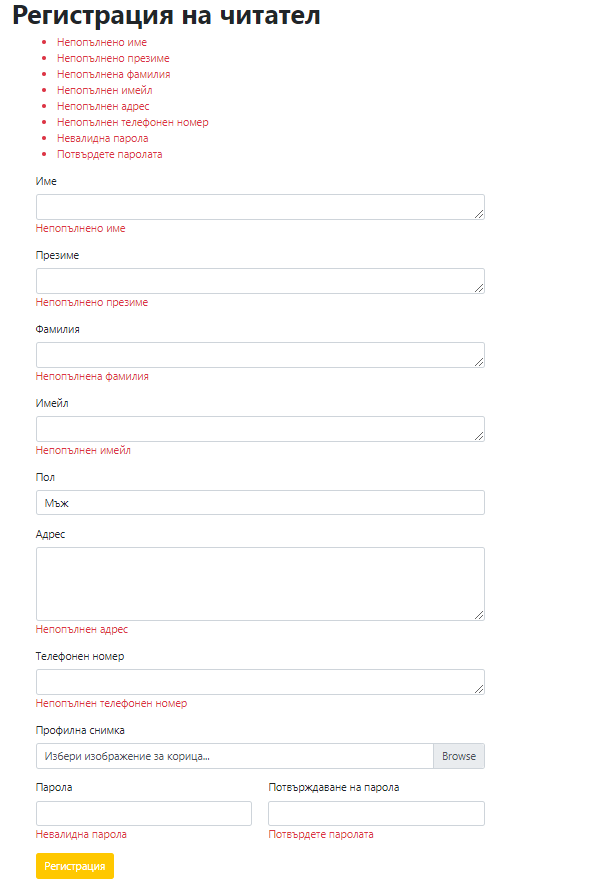
Съдържа името на приложението и линк къв Github Repository на проекта

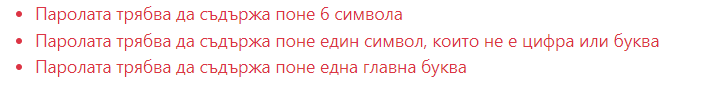


**Регистрация**

Формуляр за попълване на данни на читател с осигурена валидация включваща и проверка за съществуващ вече имейл. Данните се попълват в текстови полета, с изключение на пол, който се избира от dropdown поле и профилна снимка, която се избира чрез файл от системата, който да бъде качен.

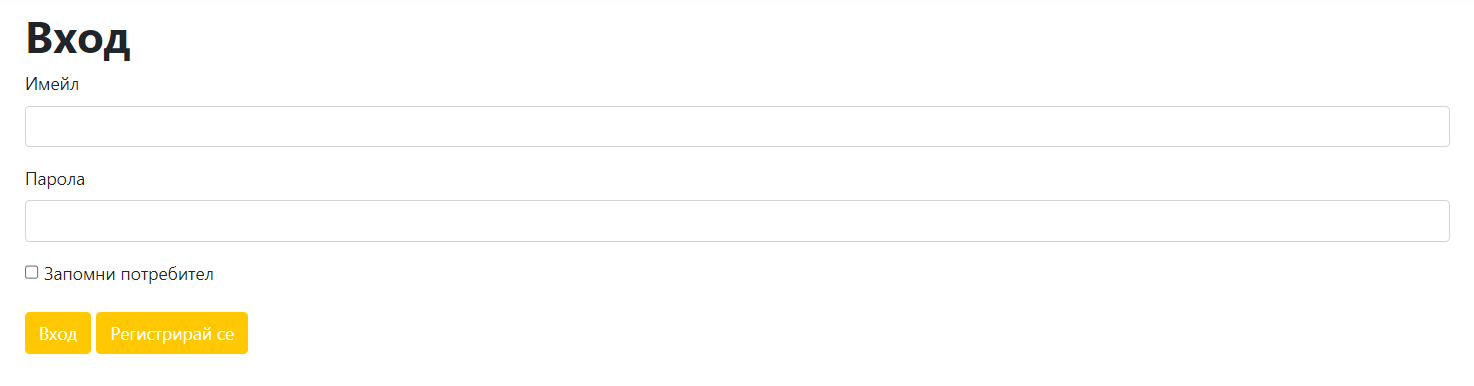
Ако не се подаде профилна снимка се използва предварително дефинирана от системата такава.





**Вход**

Формуляр за вход чрез имейл и парола с опция за запомняне на потребител при следващо отваряне





**Logout**

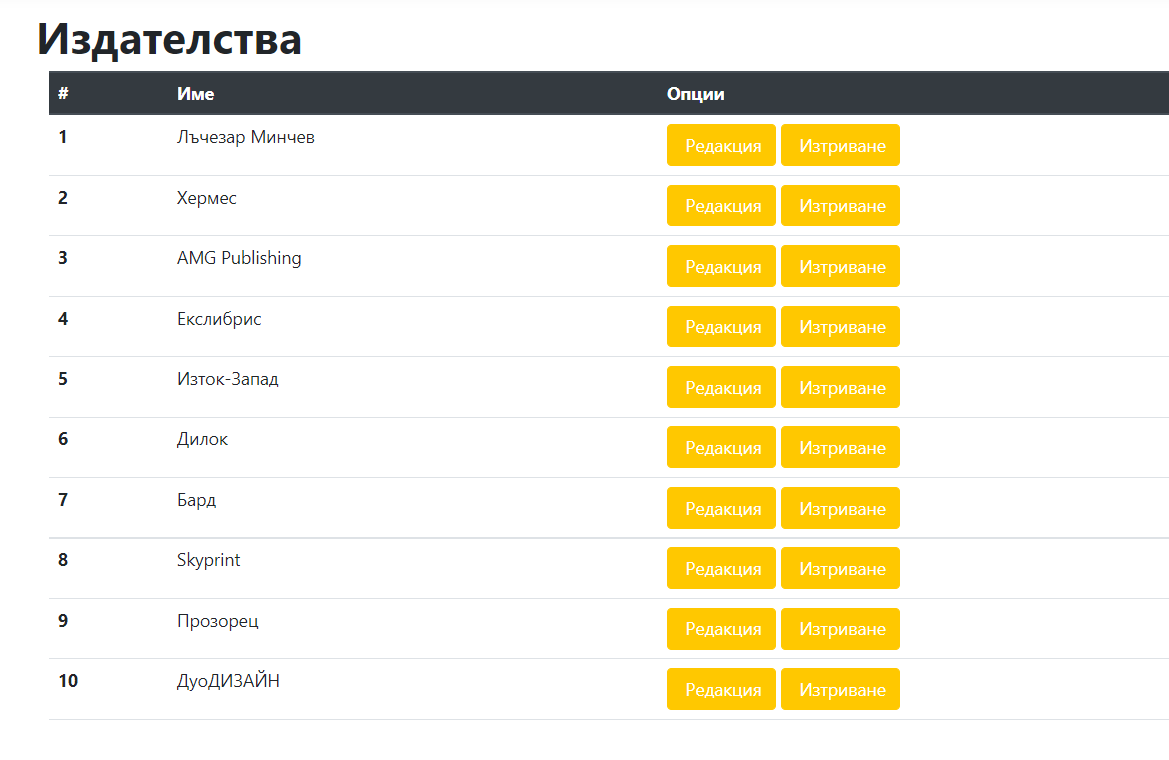
Прекратяване на сесия и отваряне на начална страница.

Извършва се чрез бутон “Изход” от Header секцията



**Преглед на издателства**

Страница за преглед на добавени издателства

****

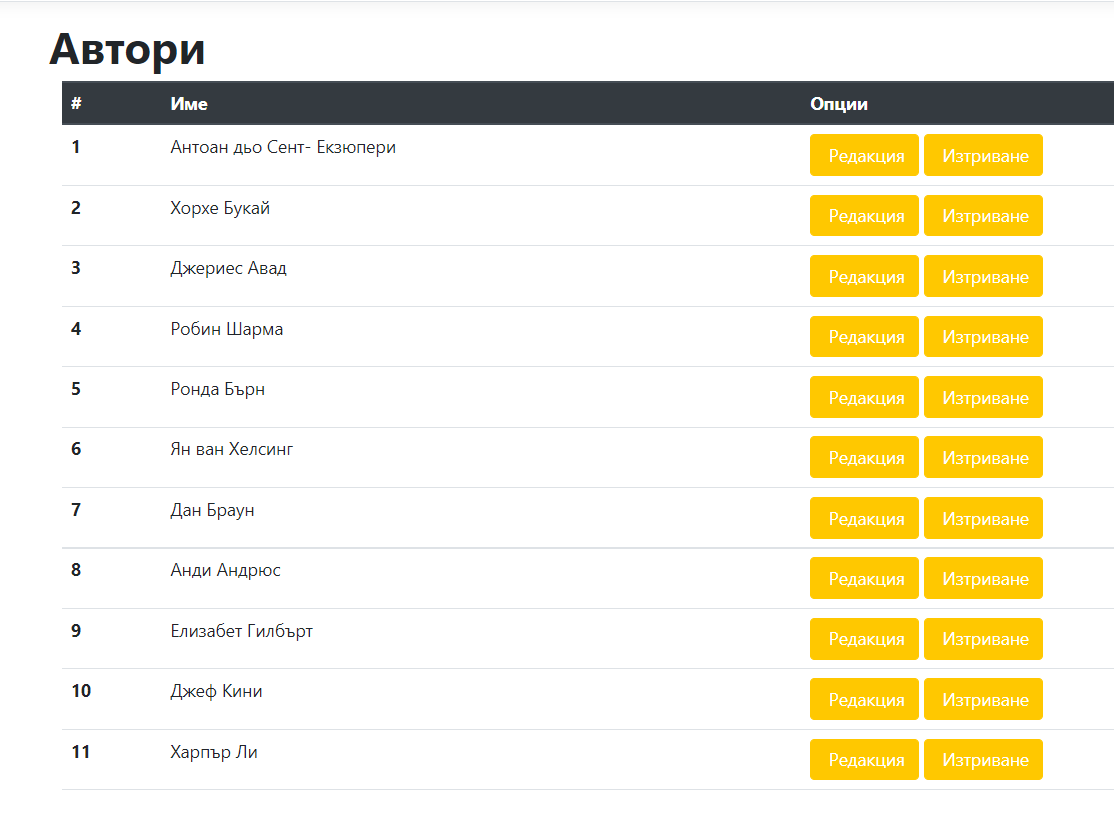
**Добавяне / Редакция на издателство**

Формуляр за Добавяне / Редакция на издателство, сменящ заглавието и името на бутоните си спрямо това дали сме от контекст на добавяне или редакция



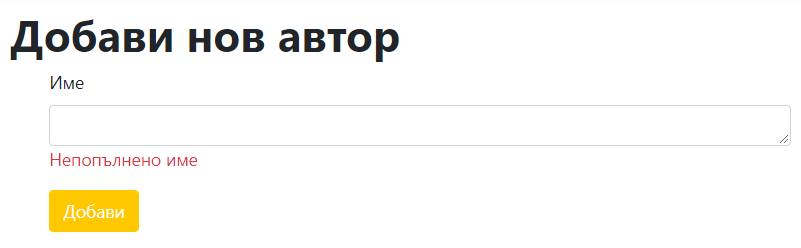
**Преглед на автор**

Страница за преглед на добавени автори



**Добавяне / Редакция на автор**

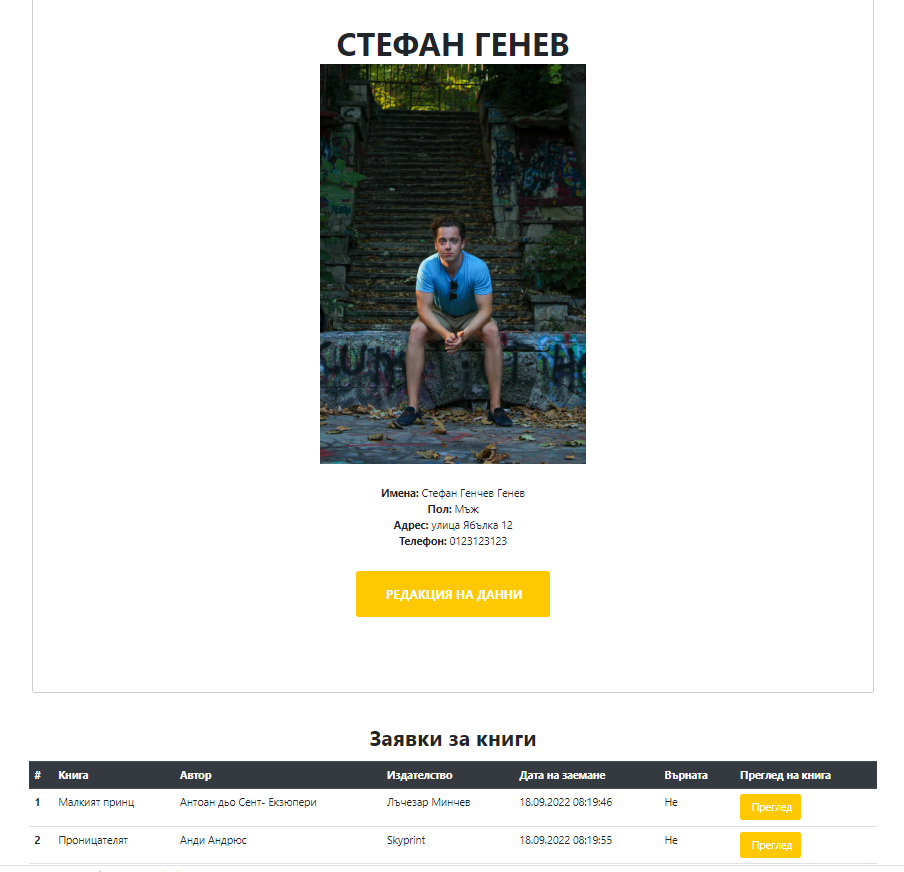
Формуляр за Добавяне / Редакция на автор, сменящ заглавието и името на бутоните си спрямо това дали сме от контекст на добавяне или редакция



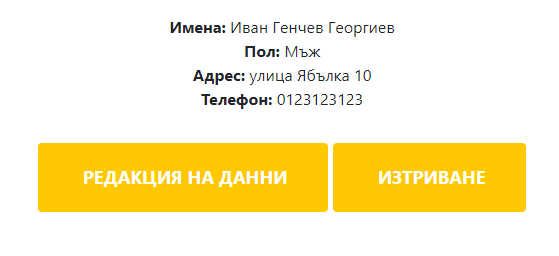
**Профилна страница**

Дава възможност за преглед и редакция на данни на читател, както и преглед на заявки за книги на потребителя

На потребител с роля “Администратор” се изобразява и опция за изтриване на профил, стига да не е неговият

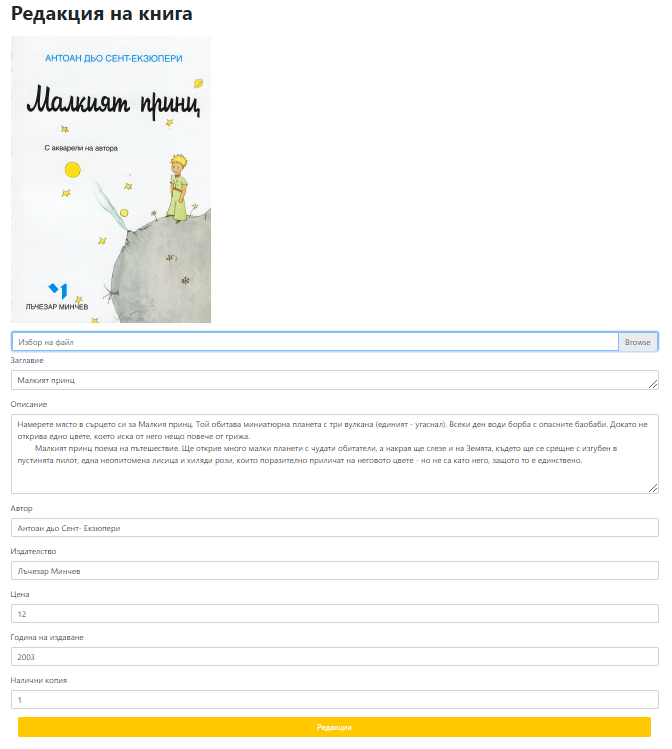
****

Изглед при роля “Администратор”:

****

**Редактиране на книга**

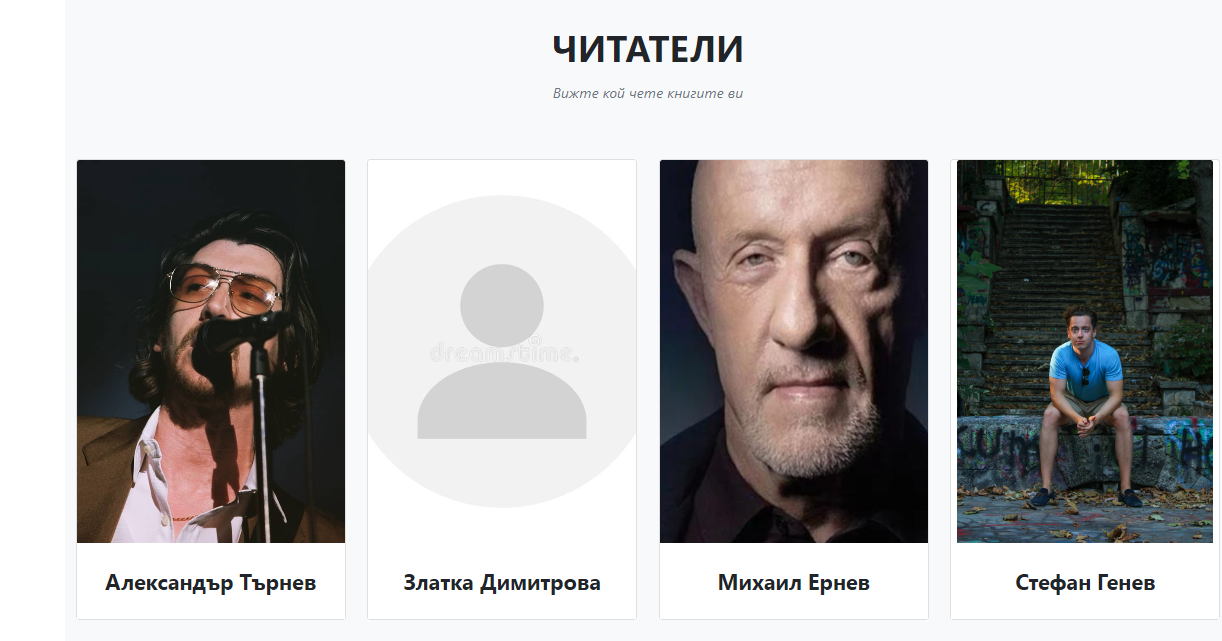
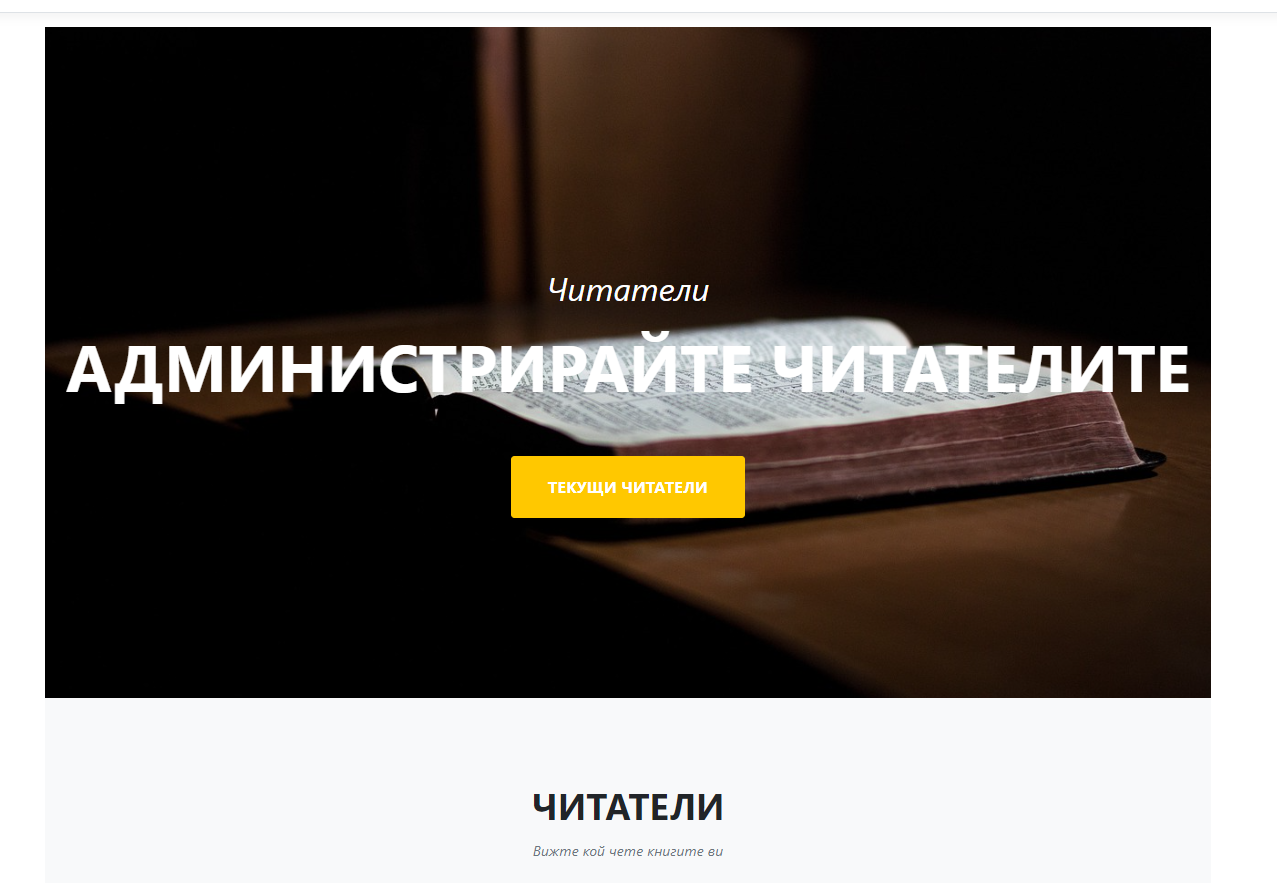
Страница за смяна на детайли на книга предоставена на потребители с роля “Администратор”.



**Читатели**

Страница за преглед на читатели от потребители с роля “Администратор”

При клик на бутона “Текущи читатели” страницата се скролва надолу и показва читателите



**Заети книги**

Страница за заети книги от потребители, достъпна единствено от потребители с роля “Администратор”

