|  |
| --- |
| Stručni kurs Razvoj bezbednog softvera |
| Izveštaj |
| Pronađene ranjivosti u projektu “RealBookStore” |

|  |
| --- |
| Julijana Jevtić |

# Istorija izmena

|  |  |  |  |
| --- | --- | --- | --- |
| Verzija | Datum | Izmenio/la | Komentar |
| 1.0 | 28.2.2024. | Uroš Dragojević | Kreiran izveštaj |
| 1.1 | 13.3.2024. | Julijana Jevtić | SQL injection i Cross-site scripting |
| 1.2 | 30.4.2024. | Julijana Jevtić | Cross-site request forgery |
| 1.3 | 2.5.2024. | Julijana Jevtić | Implementacija autorizacije |
| 1.4 | 3.5.2024. | Julijana Jevtić | DevOps |
| 1.5 | 3.5.2024. | Julijana Jevtić | Primena alata za statičku analizu |
|  |  |  |  |

Sadržaj

[Istorija izmena 1](#_Toc165586895)

[Uvod 3](#_Toc165586896)

[O veb aplikaciji 3](#_Toc165586897)

[Kratak pregled rezultata testiranja 3](#_Toc165586898)

[SQL injection 4](#_Toc165586899)

[Napad: Ubacivanje novog usera u tabelu “persons” (SQL injection) 4](#_Toc165586900)

[Metod napada: 4](#_Toc165586901)

[Predlog odbrane: 4](#_Toc165586902)

[Cross-site scripting 5](#_Toc165586903)

[Napad: Ubacivanje novog usera u tabelu “persons” i krađa kolačića sesije 5](#_Toc165586904)

[Metod napada: 5](#_Toc165586905)

[Predlog odbrane: 5](#_Toc165586906)

[Cross-site request forgery 6](#_Toc165586907)

[Napad: Menjanje ličnih podataka. 6](#_Toc165586908)

[Metod napada: 6](#_Toc165586909)

[Predlog odbrane: 7](#_Toc165586910)

[Implementacija autorizacije 8](#_Toc165586911)

[DevOps 9](#_Toc165586912)

[Zaključak 10](#_Toc165586913)

# Uvod

Ovaj izveštaj se bavi ranjivostima pronađenim u dole opisanoj veb aplikaciji.

## O veb aplikaciji

RealBookStore je veb aplikacija koja pruža mogućnosti pretrage, ocenjivanja i komentarisanja knjiga.

Aplikacija RealBookStore omogućava sledeće:

* Pregled i pretragu knjiga.
* Dodavanje nove knjige.
* Detaljan pregleda knjige kao i komentarisanje i ocenjivanje knjige.
* Pregled korisnika aplikacije.
* Detaljan pregled podataka korisnika.

## Kratak pregled rezultata testiranja

*Ovde idu kratko opisani rezultati testiranja: pronađene ranjivosti i nivo opasnosti.*

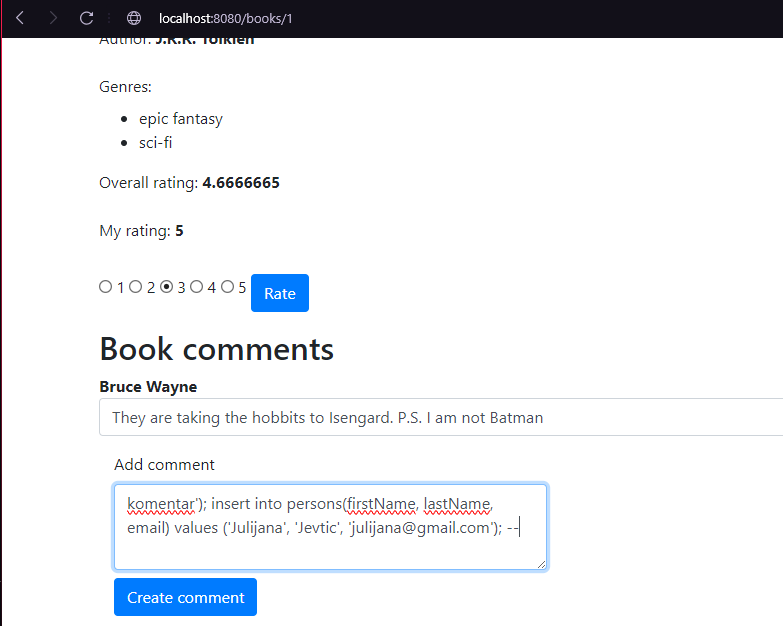
|  |  |
| --- | --- |
| *Nivo opasnosti* | *Broj ranjivosti* |
| *Low* | *3* |
| *Medium* | *2* |
| *High* | *1* |

# SQL injection

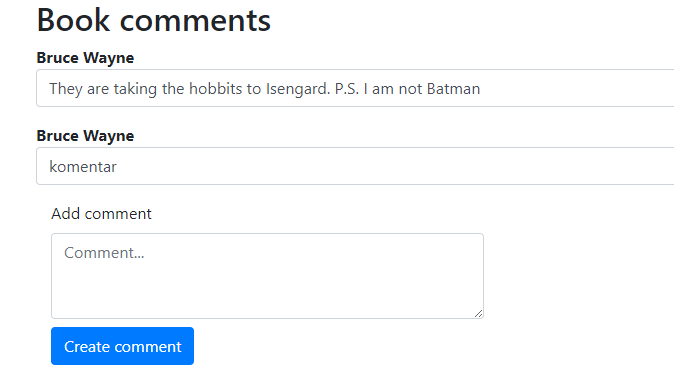
## Napad: Ubacivanje novog usera u tabelu “persons” (SQL injection)

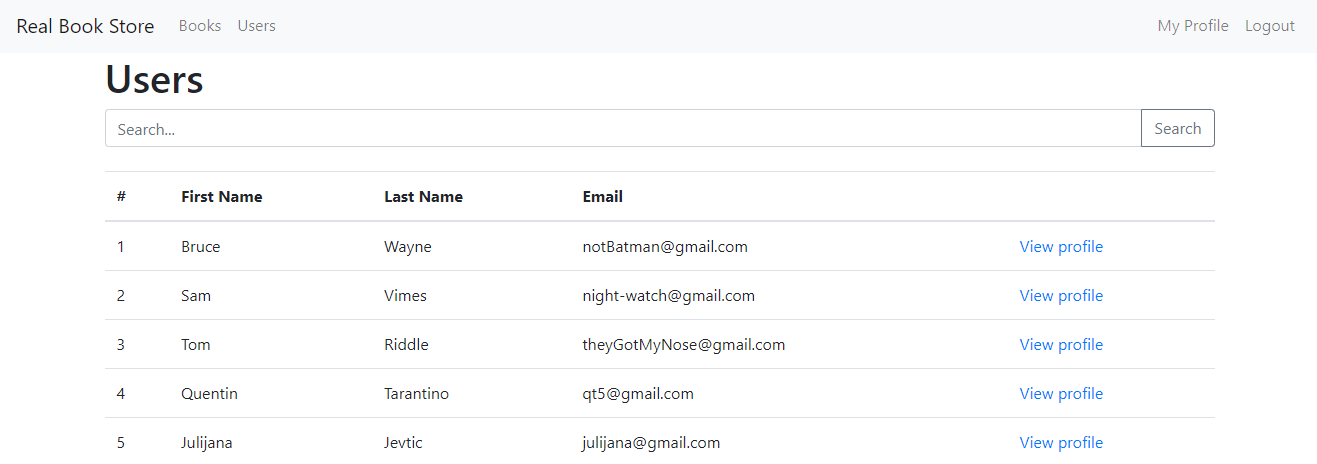
### Metod napada:

Na stranici za pregledanje pojedinačne knjige, uneti sledeći kod u input polje “Comment”:



Prikaz:





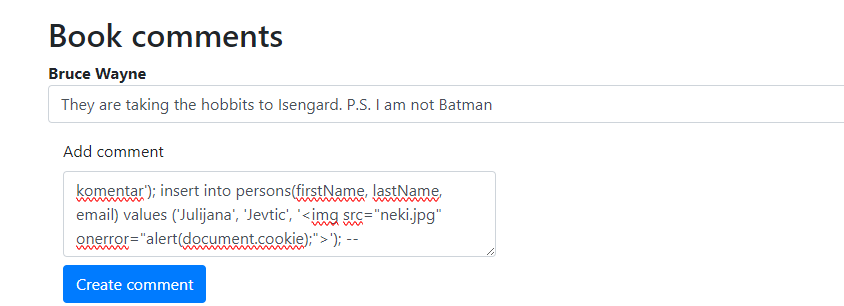
## Predlog odbrane:

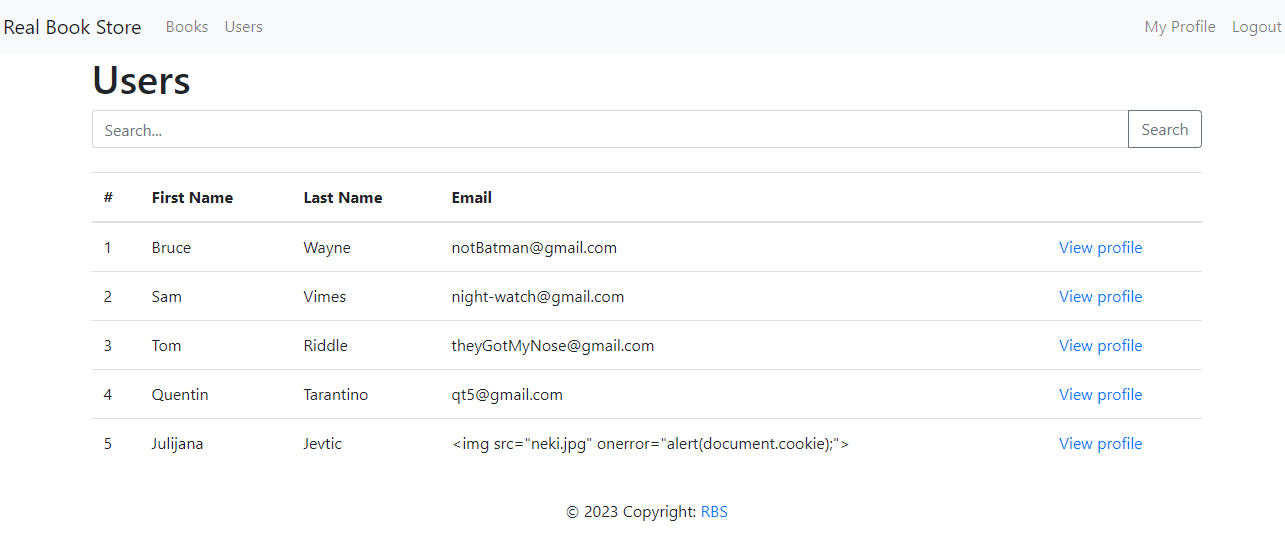
Koristiti PreparedStatement umesto Statement, tj koristimo parametrizovane upite.

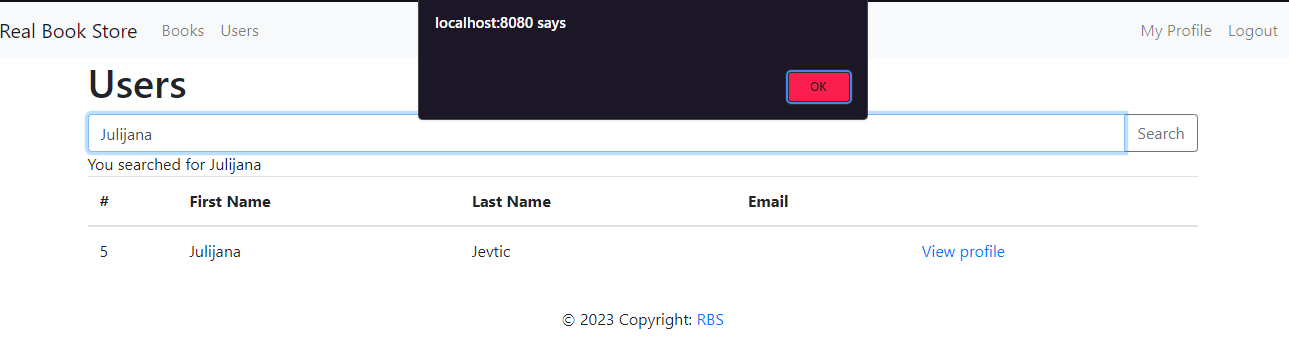
# Cross-site scripting

## Napad: Ubacivanje novog usera u tabelu “persons” i krađa kolačića sesije

### Metod napada:

Na stranici za pregledanje pojedinačne knjige, uneti sledeći kod u input polje “Comment” (kombinujemo sa SQL Injection-om): 

Prikaz:

Ponašanje pri pretrazi ubačenig korisnika: 

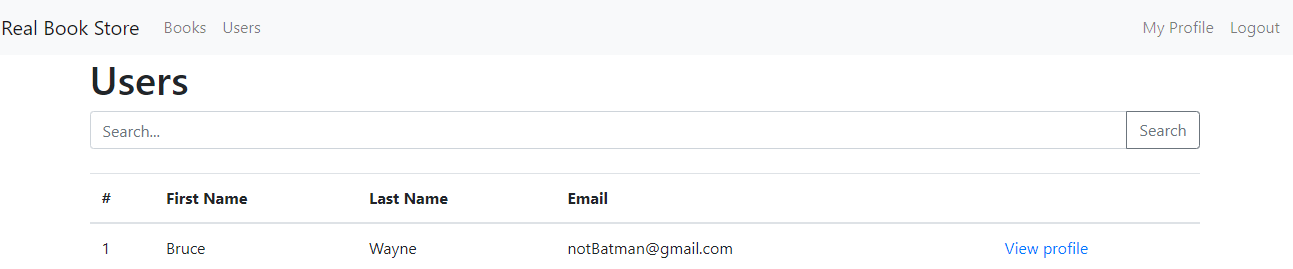
## Predlog odbrane:

Treba koristiti textContent umesto innerHTML i umesto th:text na html tagu koristimo th:utext(unescaped text).

# Cross-site request forgery

## Napad: Menjanje ličnih podataka.

### Metod napada:

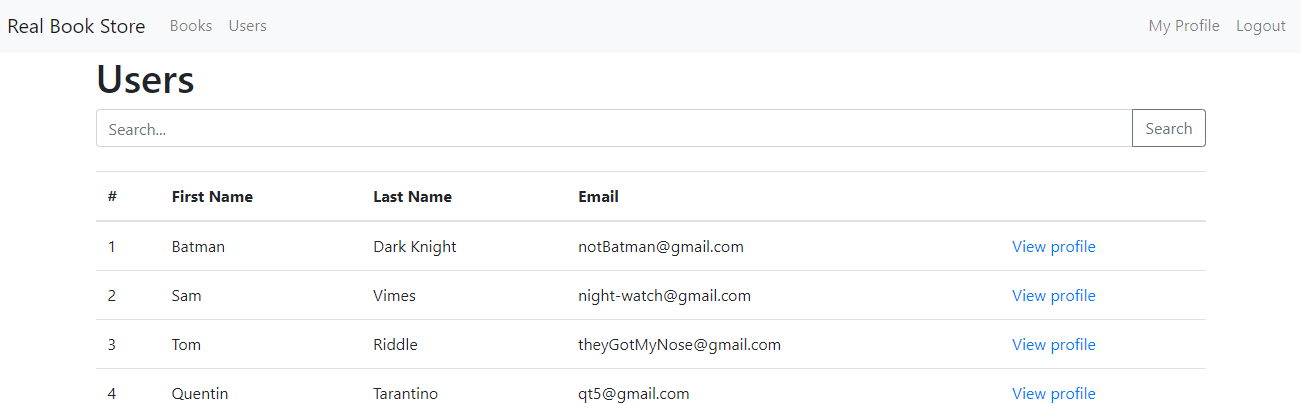


Klikom na maliciozni link, pokreće se skripta koja šalje zahtev serveru i menja podatke korisnika sa id = 1, tako da je firstName = “Batman” i lastName = “Dark Knight”.

Funkcija:

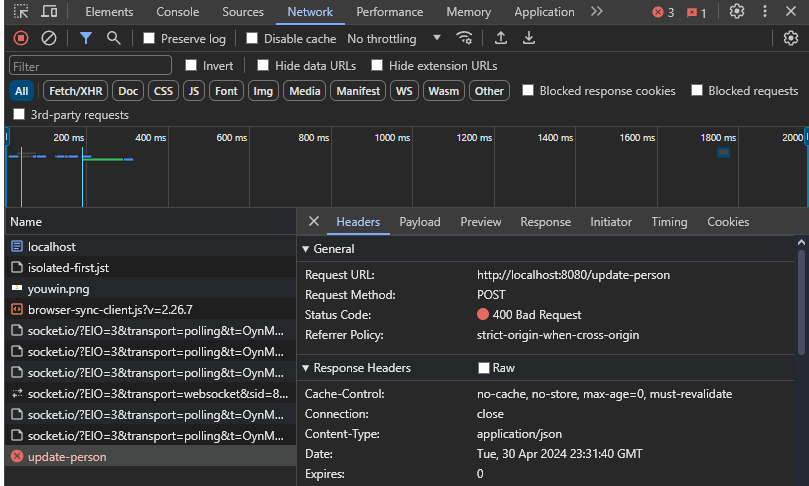


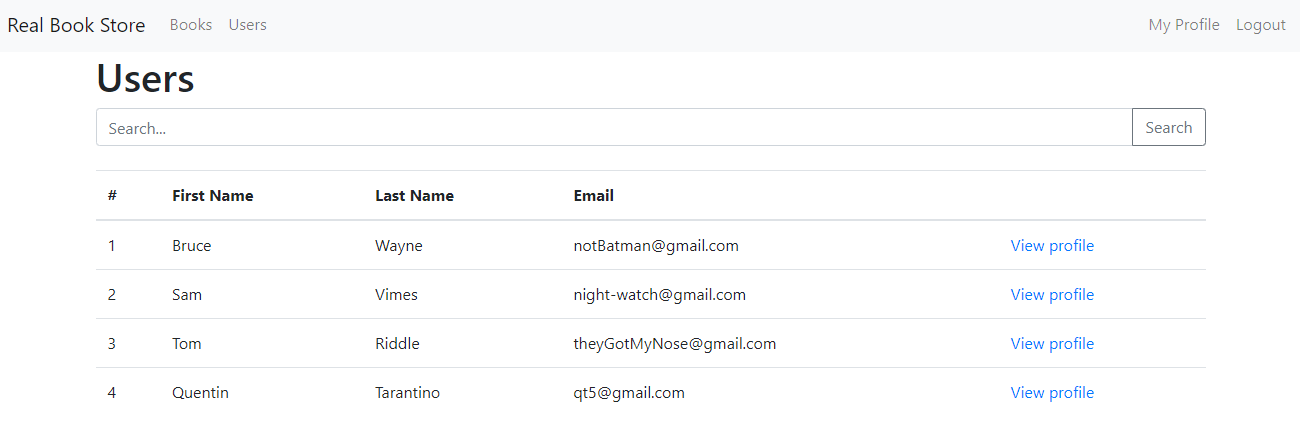
Klikom na maliciozni link promenili smo ime i prezime:



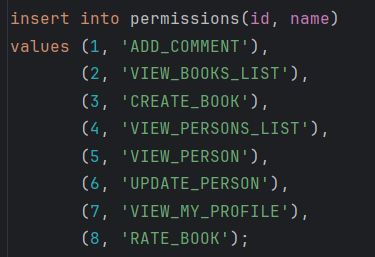
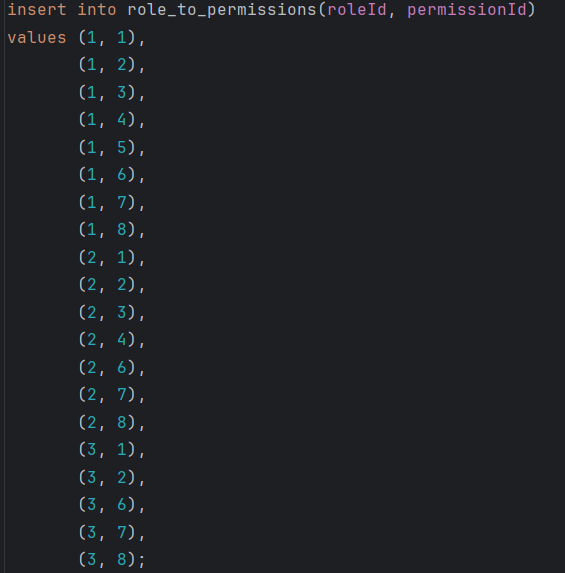
Predlog odbrane:

Uključiti CORS zaštitu. Koristiti token (šablon - Synchronizer Token Pattern.).



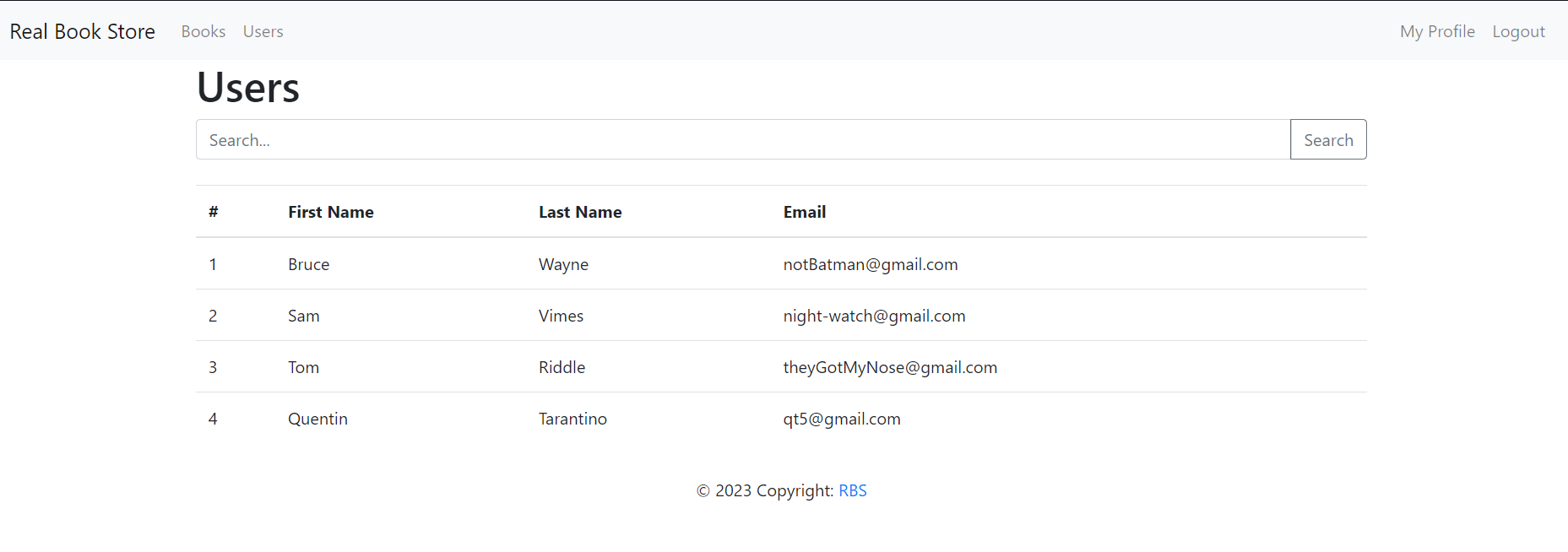


# Implementacija autorizacije

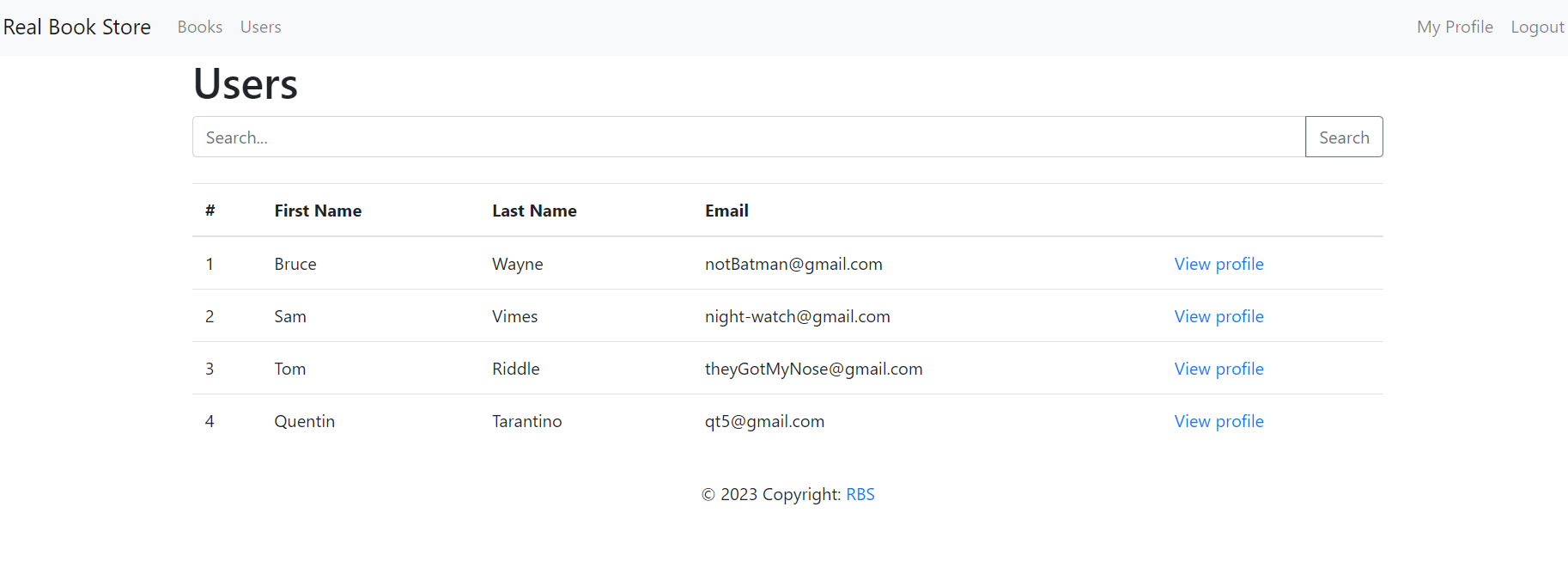
Implementirati autorizacioni model (matricu permisija) u bazi podataka.

Primenili smo Spring Security I Thymeleaf koncepte.

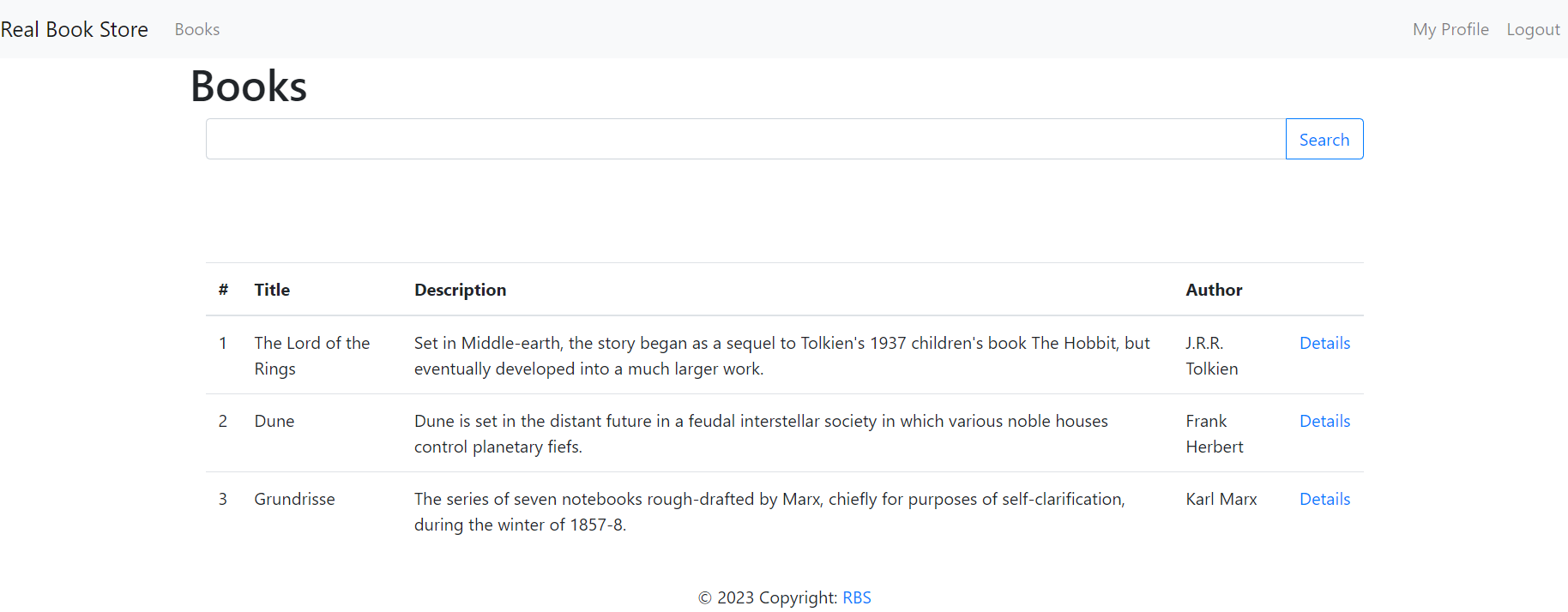
Perspektiva toeLover (MANAGER):



Perspektiva tom (ADMIN):



Perspektiva bruce (REVIEWER):



# DevOps

Radi pratnje promena i grešaka:

* Uvedena obrada i logovanje svih izuzetaka u aplikaciji. Dodati logovi koji bi bili korisni u analizi u slučaju napada.
* Uveden auditing aplikaciji.

# Zaključak

|  |  |
| --- | --- |
| **Tema** | **Opis** |
| Uvod u kurs i alate |  |
| SQL Injection | Kako identifikovati i popraviti ranjivost koristeći ustanovljene pristupe i frameworke |
| Cross Site Scripting (XSS) |
| Cross Site Request Forgery (CSRF) |
| Alati za statičku i dinamičku analizu | SonarQube (statička)  OWASP ZAP (dinamička)  Tumačenje izveštaja |
| Sigurna implementacija autentifikacije | Sigurno skladištenje lozinke  Dvofaktorska autentifikacija |
| Autorizacioni modeli | Analiza zahteva i implementacija RBAC |
| Sigurnost i DevOps | Obrada izuzetaka, logging i auditing |