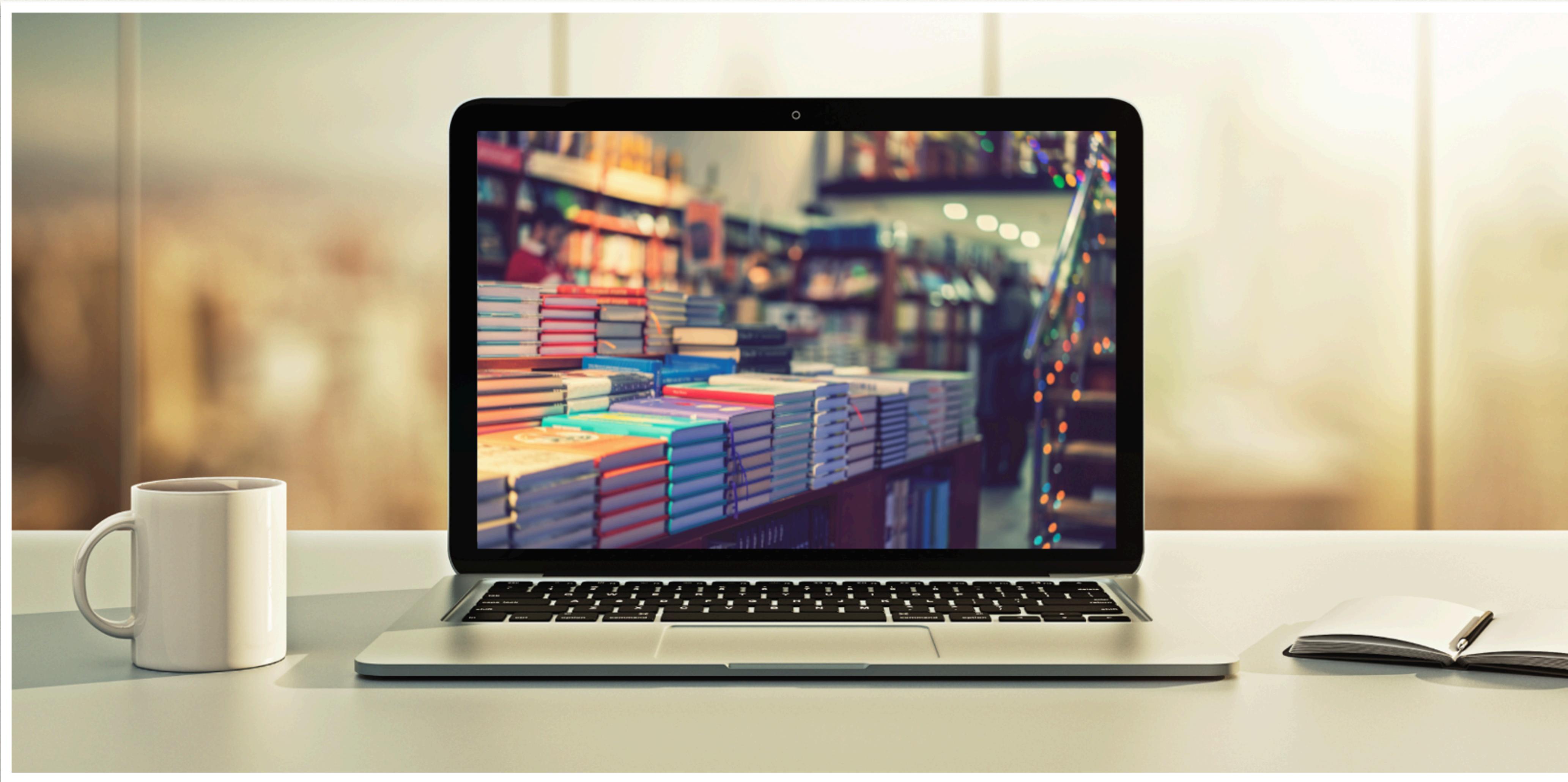
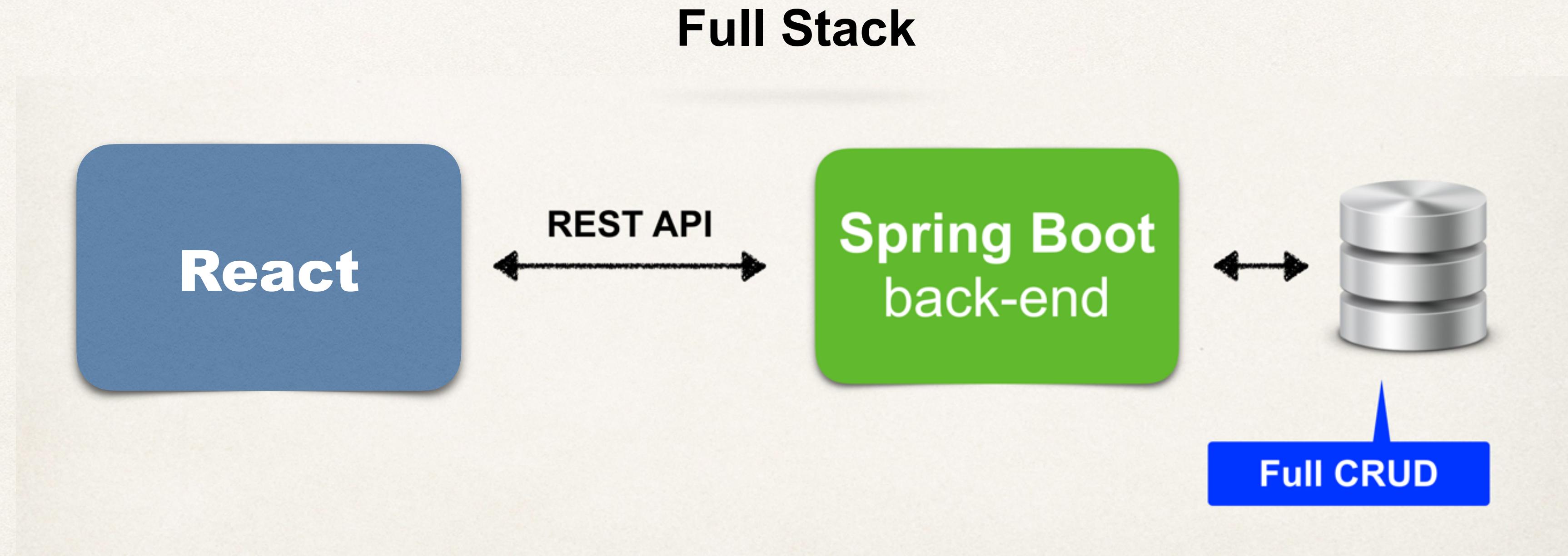


Review Component Overview



React Front End

- Create React Front End components
- Retrieve data from Spring Boot REST APIs



Spring Boot REST Endpoint

- Create new Review Repository

```
public class Review {  
    private Long id;  
    private String userEmail;  
    private Date date;  
    private double rating;  
    private Long bookId;  
    private String reviewDescription;  
}
```

Add Hibernate/Lombok Annotations
@Entity
@Table
@Data
...

Review DAO

```
public interface ReviewRepository extends JpaRepository<Review, Long> {  
    page<Review> findByBookId(@RequestParam("book_id") Long bookId, Pageable pageable);  
}
```

Spring Boot REST Endpoint

- Add Review Entity to our MyDataRestConfig

Expose the PK for each review

```
config.exposeIdsFor(Review.class);
```

Disable POST, PATCH, PUT, DELETE

```
disableHttpMethods(Review.class, config, theUnsupportedActions);
```

React TypeScript Model

- Create new Review Model that we can use within our application

```
class ReviewModel {  
  
    id: number;  
  
    userEmail: string;  
  
    date: string;  
  
    rating: number;  
  
    book_id: number;  
  
    reviewDescription: string;  
  
    constructor(...) {  
        this.id = id;  
        etc ...  
    }  
}
```

TypeScript Model

Calling Spring Boot API from React Application

- JavaScript **fetch** functionality

```
useEffect (() => {  
  
    const fetchBookReviews = async () => {  
  
        const reviewUrl: string = `http://localhost:8080/api/reviews`;  
  
        const responseReviews = await fetch(reviewUrl);  
  
        const responseJsonReviews = await responseReviews.json();  
  
        const responseData = responseJsonReviews._embedded.reviews;  
  
        ..... (more code) .....  
    };  
  
    fetchBookReviews().catch((error: any) => {  
  
        setIsLoadingReview(false);  
  
        setHttpError(error.message);  
    })  
}, []);
```

Stop loading the app once call is complete

useEffect: React Hook that tells our React app that it needs to do something after render. In this case, fetch information for our reviews.

fetch(): JavaScript method that is an easy way to fetch resources (data) asynchronously across a network. In our case calling the Spring Boot API endpoint.