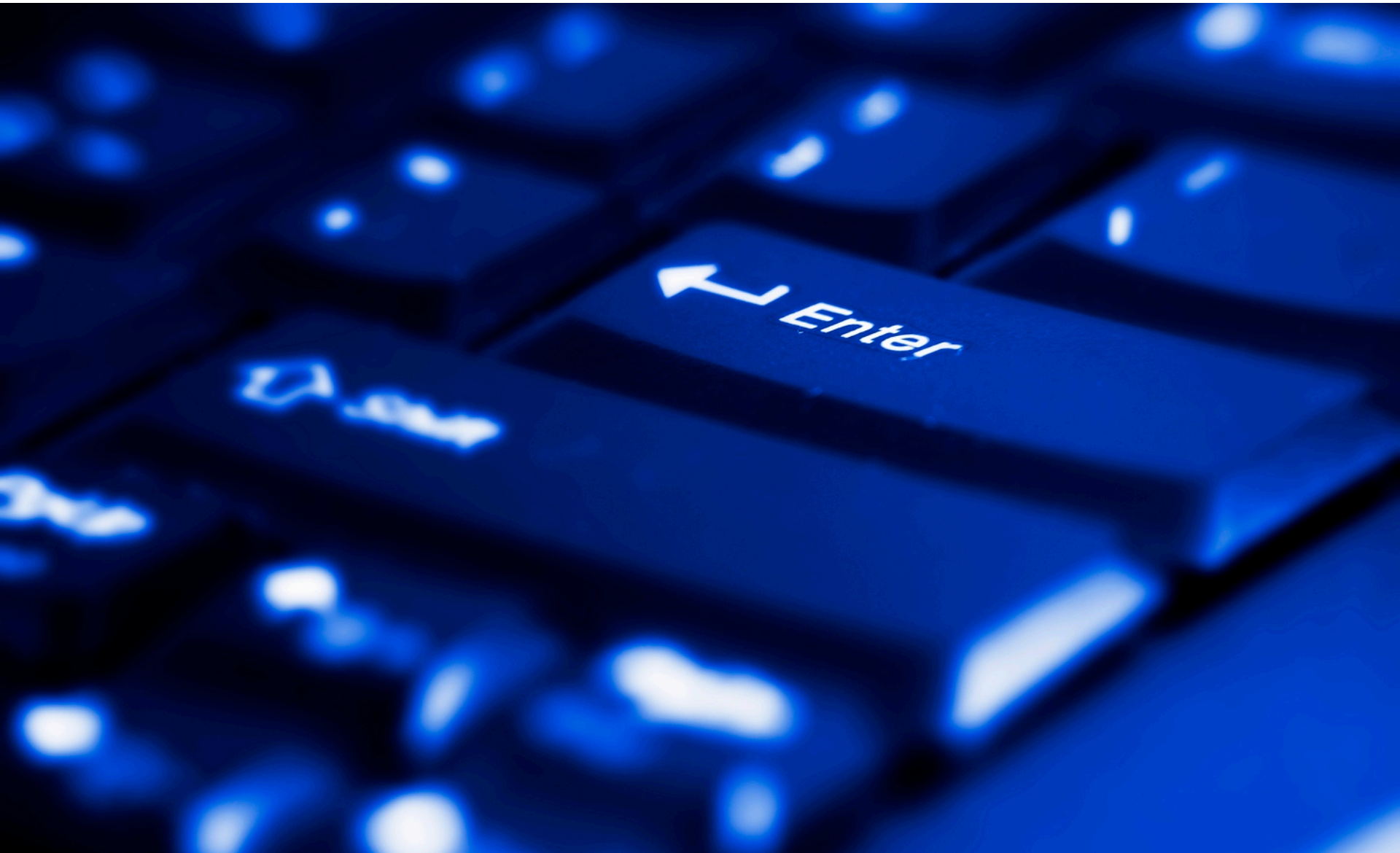




Implementing Microservices

Modern Application Architectures



The Bokker

- # I can see a list of books (with ratings)
- # I can add/modify the book
- # I can see a list sorted by ratings, book name
- # I can rate the book
- # I can see details about the book (with detailed information about rating)
- # I can see history of rating changes (of changes)
- # I can borrow the book
- # I can see all borrowed books sorted by due date
- # I can filter the books list: all, borrowed, available, to be returned

- # Model the Borrowing Aggregate/Entity
- # Introduce appropriate Building Blocks
- # Move all classes to separate package

1. Model the Rating Aggregate/Entity
2. Introduce appropriate Building Blocks
3. Move all classes to separate package
4. Create REST endpoint for:
 - rate (/rating/{id} POST)
 - currentRating (/rating/{id} GET)
5. Move Rating BC classes to separate module (REST module)

1. Install RabbitMQ
2. Write Hello World publisher/subscriber example
3. RatingChanged event
 - Modify Rating MS to send event about rating change (Publisher)
 - Modify Main Application to listen to the rating change and update model accordingly (Subscriber)

(for ambitious)

1. Move Borrowing to separate service (module) and create REST endpoints
2. Publish and handle Borrowing BookBorrowed/BookReturned event