Use Cases and Logical Architecture

- XID: X00179223
- Name: Raymond McCarthy
- Project Title: Computing with Software Development

Provide at least 6 Use-cases describing the functionality of the proposed system

Section 1: For Each Use Case:

Title (goal)	Add book to library inventory
Primary Actor	Library administrator
Story	Admin can add a book to the library inventory which is available for loan

Title (goal)	Remove book from library inventory
Primary Actor	Library administrator
Story	Admin can remove a book from library inventory which will no longer be available for loan

Title (goal)	Edit book in library inventory
Primary Actor	Library admin
Story	Admin can update details about books and availability

Title (goal)	Reserve book
Primary Actor	Library member/Library admin
Story	Library member can reserve a book to borrow. Library admin can reserve a book on behalf of a member.

Title (goal)	Cancel reservation
Primary Actor	Library member/Library admin
Story	Library member can cancel a reservation. Library admin can cancel a reservation on behalf of a member.

Title (goal)	Check out book
Primary Actor	Library admin
Story	Library admin can check out a book on behalf of a member.

Title (goal)	Check for late book return
Primary Actor	System
Story	System will run a job frequently to check if books are not returned on time, if a book is late, it will send an automated email to the library member at fault

Title (goal)	Administer fine for overdue book
Primary Actor	System
Story	If a book is overdue greater than a period (e.g., one day), the system will add a fine to the library members account

Title (goal)	Check in books
Primary Actor	Library admin
Story	Library admin can check in books on behalf of a member.

Title (goal)	Create account
Primary Actor	Library member/Library admin
Story	Library members can create an account. Library admin can create an account of behalf of a new member.

Title (goal)	Create admin account
Primary Actor	Library admin
Story	Library admin can create account on behalf of new library admin staff.

Title (goal)	Log in/log out
Primary Actor	Library member/Library admin

Story	All users can log in and log out of their account

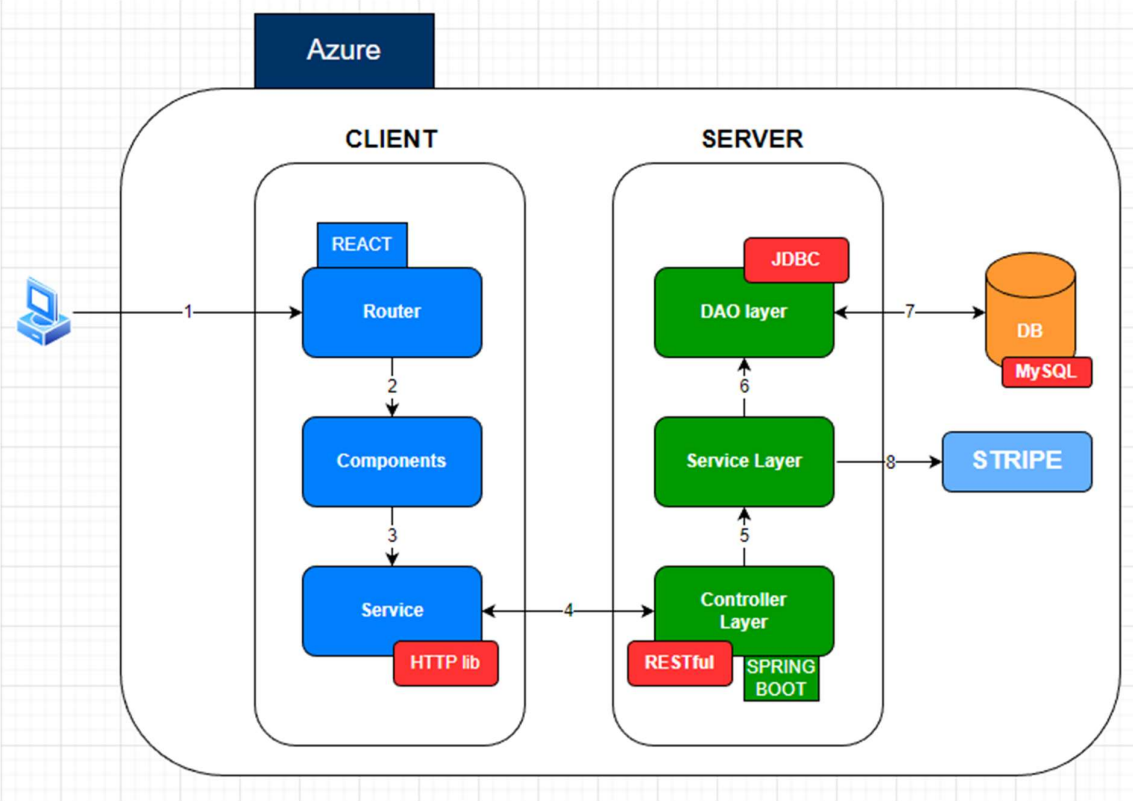
Title (goal)	Edit account details
Primary Actor	Library member/Library admin
Story	Library members can edit their account details. Library admin can edit their account details. Library admin can edit account details on behalf of a member.

Title (goal)	Extend book checkout lease length
Primary Actor	Library admin
Story	Library admin can book checkout lease length on behalf of a member.

Title (goal)	Search through book catalogue
Primary Actor	Any user
Story	Any user can look through the library’s catalogue of books

Title (goal)	Pay fine for late book return
Primary Actor	Library member/Library admin
Story	Library member can pay a fine for late book return.

Section 2: Logical Architecture



- 1
  - This is the access point for the user, they will enter a URL which the ‘React Router’ will use to route requests throughout the program.
- 2
  - Routes will call components and determine what is displayed on the UI.
- 3
  - How the user interacts with a component will determine what requests are made to the service.
- 4
  - Requests will be sent from the client to the server’s ‘Controller Layer’ endpoints using a HTTP client (Axios).
- 5
  - Depending on the content of the request the ‘Controller Layer’ will route it to the correct service based on the URI and HTTP request method.
- 6
  - The ‘Service Layer’ will breakdown any data passed to and perform logic based on the nature of the request.
- 7
  - The ‘DAO layer’ will be used to interact with the database.
- 8
  - Payments for late fees will be performed using Stripe.

Client languages and libraries: JavaScript, JSX, React, Axios, Bootstrap
Server languages, libraries and framework: Java, JDBC, Spring Boot
Database: MySQL
Deployment service: Azure