



**Faculty of Engineering, Architecture and Science  
Department of Electrical and Computer Engineering  
Laboratory Report Cover Page**

Course Number	COE692
Course Title	Software Design Architecture
Semester/Year	Winter 2022
Instructor	Faezeh Ensan
TA Name	Parastoo Jafarzadeh

Lab/Tutorial Report No.	3
-------------------------	---

Section No.	8
Group No.	N/A
Submission Date	Tuesday, March 7th, 2022
Due Date	Tuesday, March 7th, 2022

Student Name	Student ID	Signature
Hamza Iqbal	500973673	H.I
Abdulrehman Khan	500968727	A.K

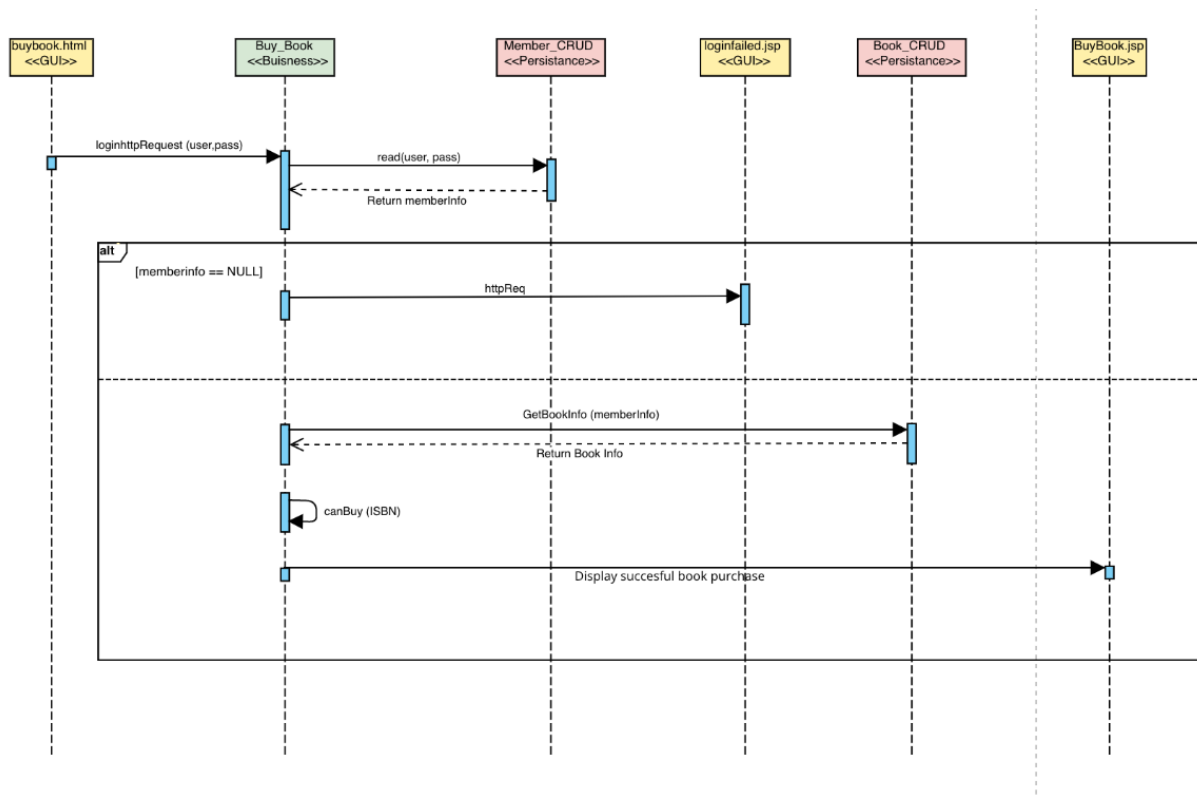


Figure 1: Sequence diagram for a member buying a book

### **Description for Sequence Diagram 1:**

1. The member logs into the system by entering their username and password (loginHttpRequest).
2. A Buy\_Book object will capture the request. Then it will call the “read” method from the MEMBER\_CRUD to retrieve the member username and password. MEMBER\_CRUD bundles information in a memberInfo object (memberInfo is a class within the helper component)
3. If no member with the provided info exists (memberInfo == NULL):  
The Buy\_Book object dispatch the httpRequest to an object of type LoginFailed.jsp
4. Otherwise:  
The Buy\_Book object gets a list of the book(s) requested by the member by calling a method (getBookInfo(memberInfo)) from an object of type Book\_CRUD in the persistence layer. BOOK\_CRUD will return a list of the books which is a class in the Helper component for bundling information related to books.

After which a method called canBuy will be called upon to decide if the book can be bought or not based on the information collected.

Finally if the canBuy returns true the Buy\_Book object will send a successful book purchase message to the BuyBook.jsp.

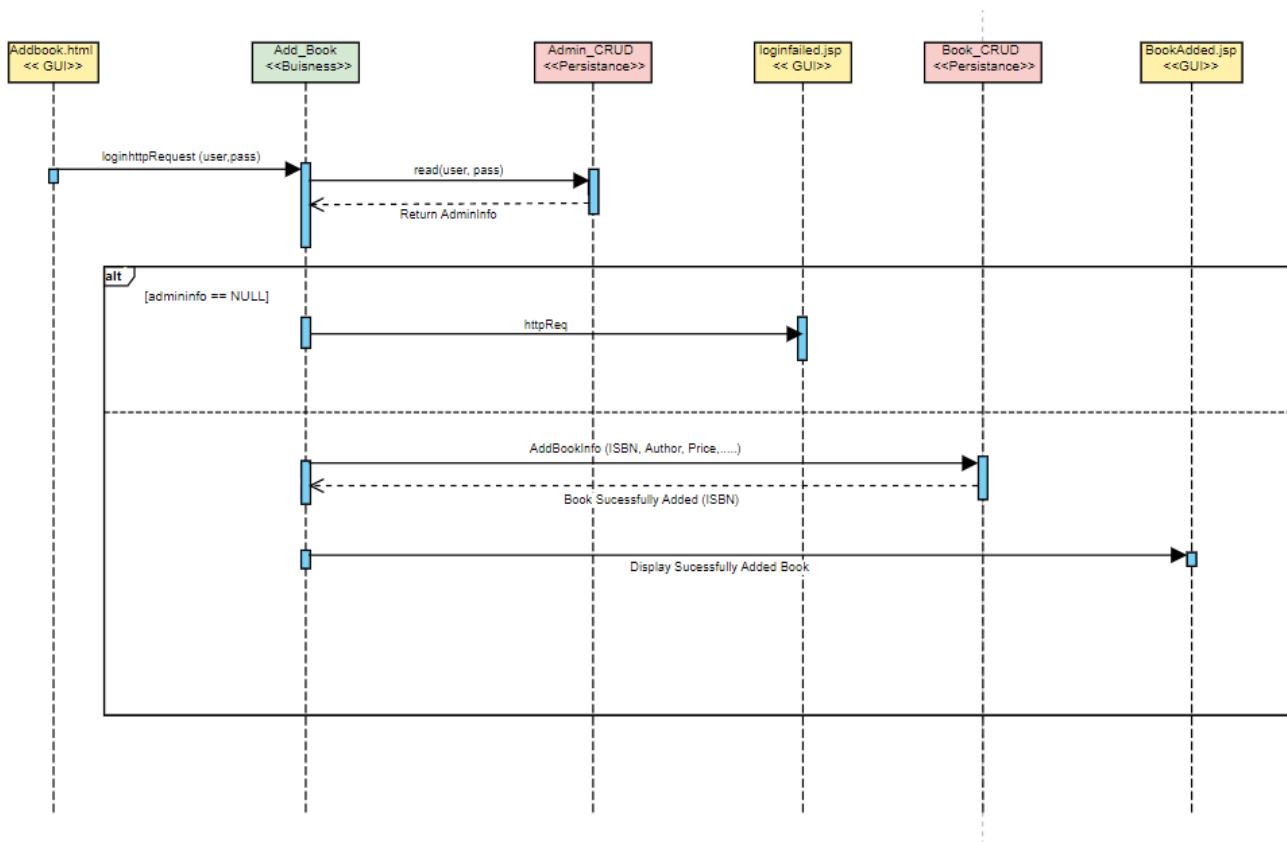


Figure 2: Sequence diagram for admin adding a book

### **Description for Sequence Diagram 2:**

1. The Admin logs into the system by entering their username and password (loginHttpRequest).
2. An Add\_Book object will capture the request. Then it will call the “read” method from the ADMIN\_CRUD to retrieve the member username and password. ADMIN\_CRUD bundles information in a adminInfo object (adminInfo is a class within the helper component)

3. If no member with the provided info exists (admininfo == NULL):

The AddBookInfo object dispatch the httpRequest to an object of type LoginFailed.jsp

4. Otherwise:

The Add\_Book object adds the book(s) by calling a method (AddBookInfo(ISBN, Author, Price....)) from an object of type Book\_CRUD in the persistence layer. BOOK\_CRUD will create and return a successful message of the books which is a class in the Helper component for bundling information related to books.

Finally the Add\_Book object will send a successful book addition message to the BookAdded.jsp.

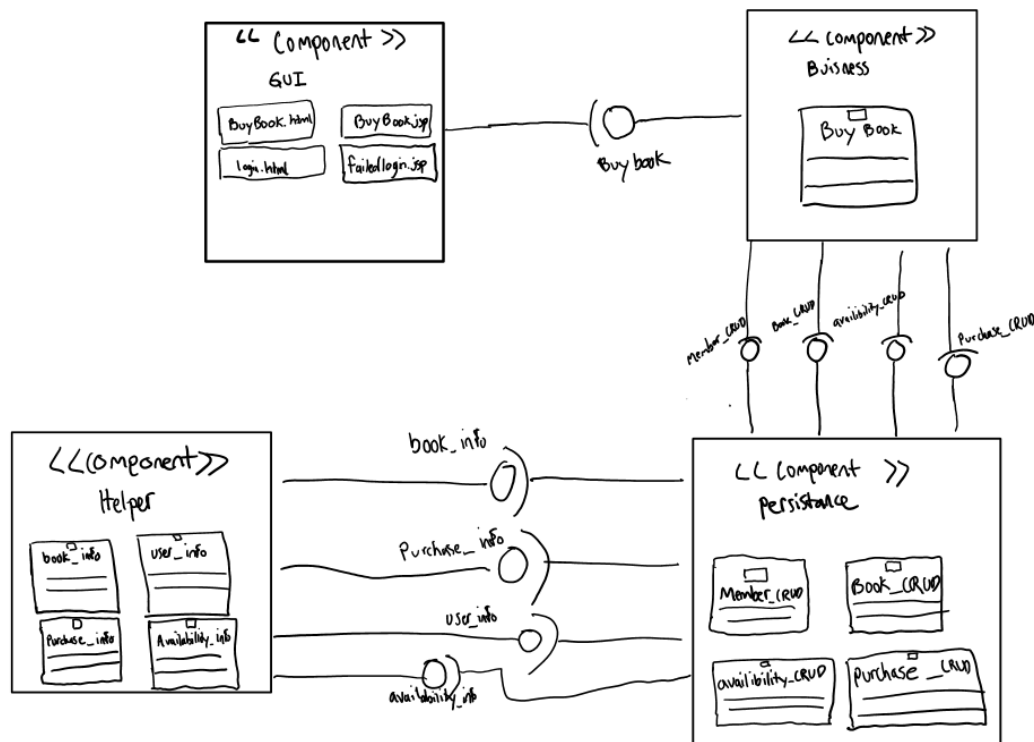


Figure 3: Component diagram for member purchasing a book

### **Description of Component diagram 1:**

The component diagram has four main components:

1. Persistence that includes all classes that query the database and create, read, update, and delete (CRUD) table rows related to the Member, Book, Purchase, and availability.
2. Business includes the class for buying a book as that is the only class that relates to the specific use case we chose.
3. GUI includes a login.html and a BuyBook.html and two jsps (BuyBook) and (loginFailed) that represent data to the user.
4. Helper includes all classes used for passing data between the components. For example the Member\_CRUD class (in the Persistence component) bundles member in an object of type MemberInfo (in Helper component) and passes it to the Buy\_Book class in the business component.

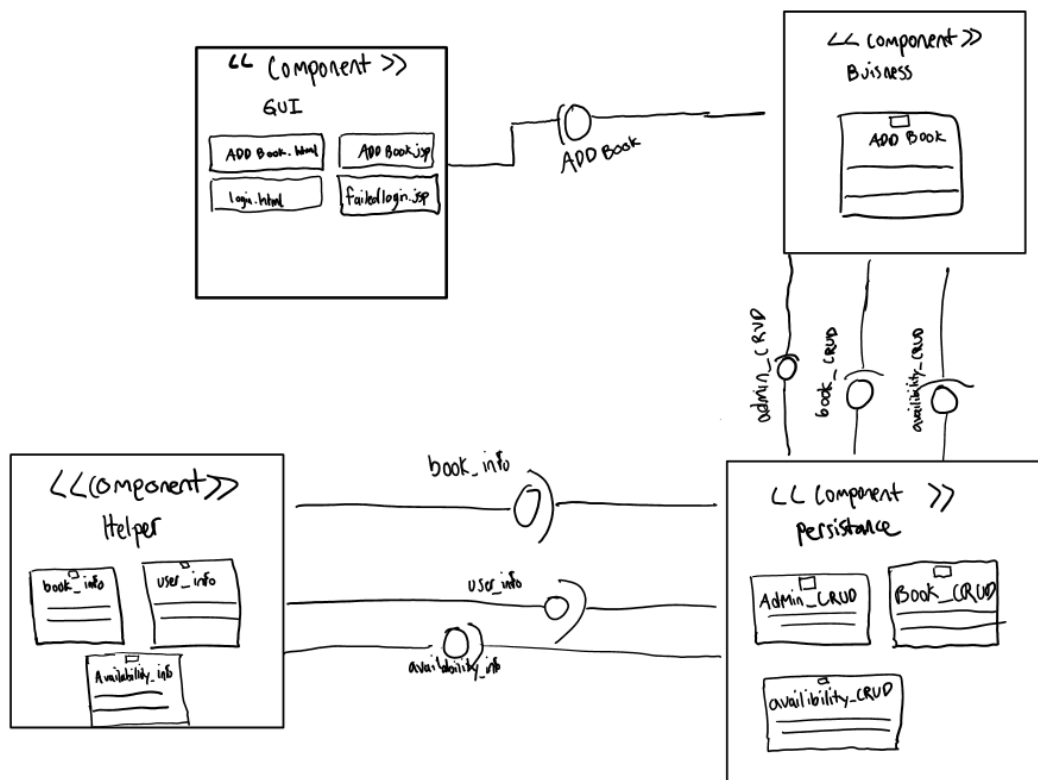


Figure 4: Component diagram for Admin adding a book

## **Description of Component diagram 2:**

This component diagram contains a GUI component which contains the addBook.html and AddBook.jsp, which will showcase the new page upon successful addition of a new book. The login and failedlogin.jsp showcase the

login page details of whether a login is successful, or not. The GUI component asks the business layer for the add book logic functionality, to which the business layer asks the persistence layer for various data inquiries. The Persistence layer which contains all classes that query the database and create, read, update, and delete (CRUD) table rows, will retrieve data from the database along with help from the helper component, and will send the necessary data back to the business layer, which will then appropriately inform the GUI.