**LIBRARY MANAGEMENT SYSTEM**

1. **Problem Statement**

The Library Management System is a software solution that automates the processes involved in managing a library. The current manual system used by the library is time-consuming and prone to errors, which affects the efficiency of the library operations. The library staff spends a significant amount of time on tasks such as book issuing, record-keeping, and generating reports. The system also lacks real-time access to information, making it difficult for users to check the availability of books or for staff to keep track of the status of books. To address these challenges, a computerized Library Management System is needed that can automate the library processes, reduce errors, and provide real-time access to information. The system should allow users to search for books, place requests, check book availability, and borrow books. It should also provide staff with the ability to manage books, members. Overall, the system should be user-friendly, secure, and scalable to accommodate future growth. The Library Management System project aims to develop and implement a comprehensive software solution that will enhance the efficiency of the library operations and improve the user experience for both staff and users.

1. **Modules of Project**

The modules for the Library Management System project could be:

Book class: This module defines the Book class and its properties such as title, author, publisher, year, and stock. It also provides methods to create a Book object and access its properties.

Library class: This module defines the Library class and its properties such as a vector of Book objects. It provides methods to add a book to the library, search for a book by title, borrow a book, and return a book.

User interface: This module provides the user interface for the library management system. It interacts with the user and calls the appropriate methods of the Library class based on the user's input.

Database: This module could be used to store and manage the data of the Library system, such as book information and user information. It could use a database management system or file-based storage to manage the data.

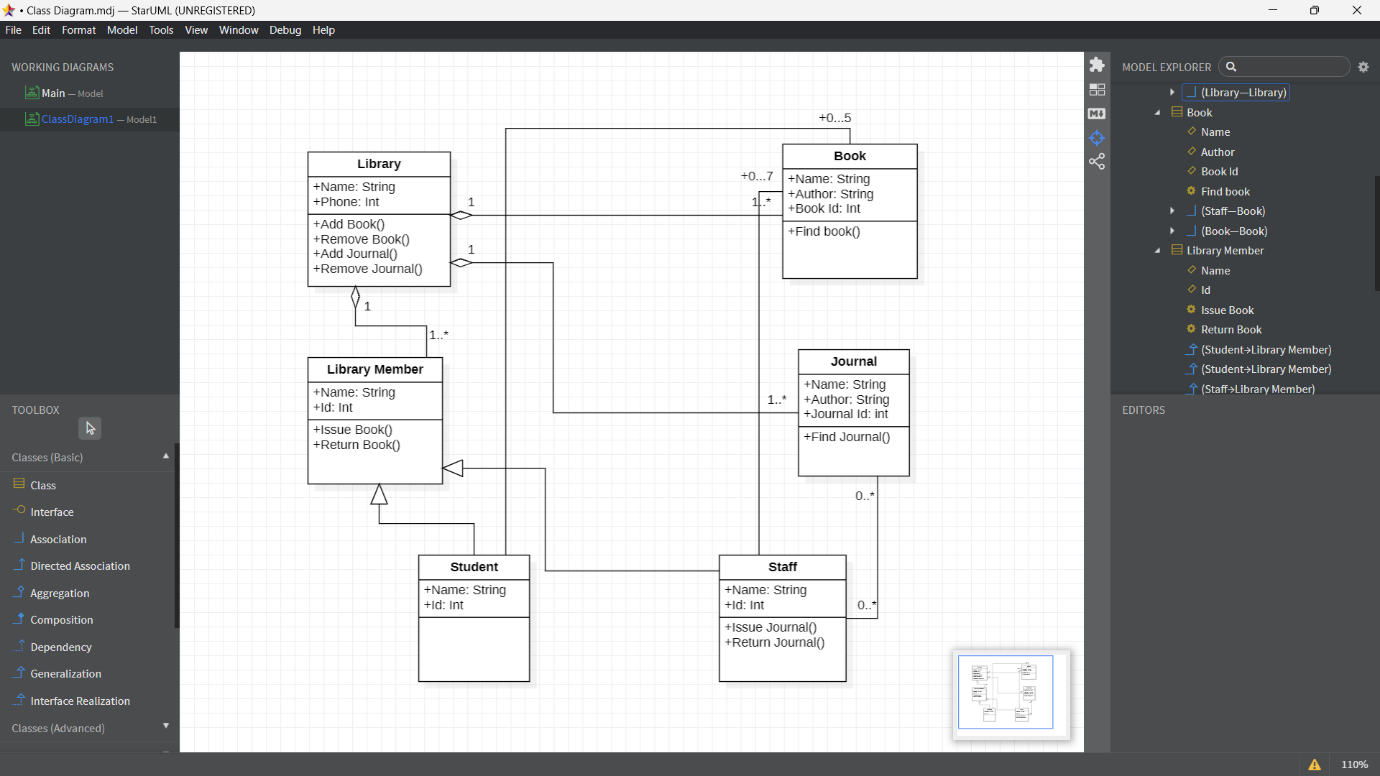
Reporting: This module could generate various reports based on the data stored in the system, such as books borrowed by a user, books available in the library, etc. It could provide these reports in a printable format or export them to a file.

1. **Importance of UML Diagrams for Library Management System**

The **UML Diagrams for Library Management System** are based in Unified Modelling Language which is standard language for describing, visualizing, building, and documenting software system artifacts. These are also used in business modelling and non-software systems. It has been discovered that all UML diagrams serve an important role in Library Management System development.

**a. Class Diagram for Library Management System**

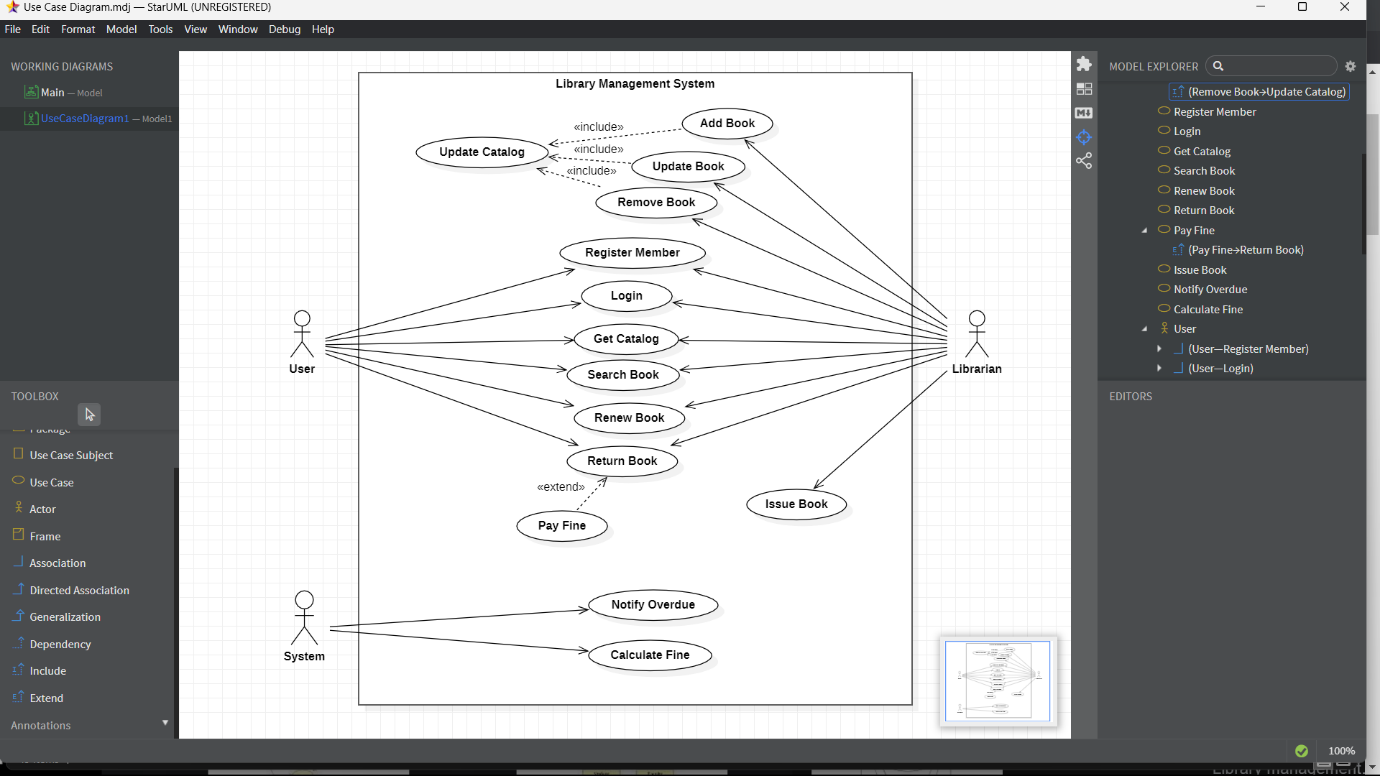
The [Class diagram for Library Management System](https://itsourcecode.com/uml/student-management-system-class-diagram-uml/) shows the structures of information or data that will be handled in the system. These data or information will be represented by classes. Each of the classes will have their attributes in accord to the methods they will use.



So the classes that must be made in a Library Management System are the **Library**, **Library Member**, **Books**, **Journal**, **Student** and **Staff**. The mentioned classes were just general. If you want more complex or wider scope of your Library management system, then you can add your desired classes.

**b. Use Case Diagram for Library Management System**

The [use case diagram](https://itsourcecode.com/uml/student-information-system-use-case-diagram/) represents the main processes in Library management system. Then they will be broken down into more specific use cases depending on the included processes of the main use case. Each of these use cases explains how the system handles the actions or scenarios requested by the user.



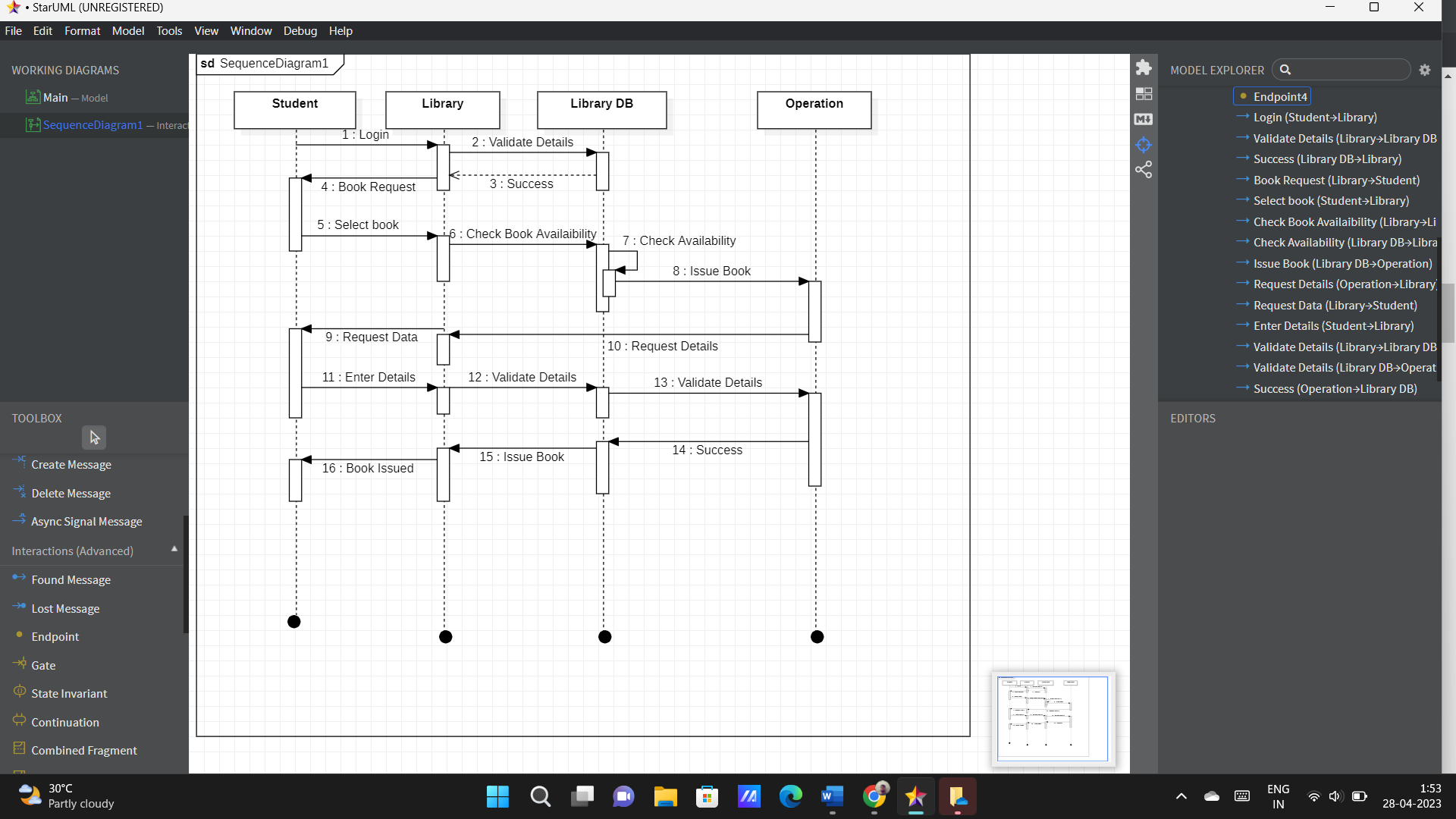
The UML Use Case Diagram is a design used as one of the Methodology on Library Management System development. It represents the main functions or processes of the system as well as the specific processes included. They were also labelled properly to guide programmers and users about the structure of Library Management System.

In this UML Use Case Diagram we have taken three actors a user, librarian, and system.

**c.** **Sequence Diagram for Library Management**

The designed sequence diagram illustrates the series of events that occurs in Library Management System. In this illustration, the actors are represented by a stick man and the transactions or classes are represented by objects. It will give you clear explanation about the behaviour of a Library Management System in terms of processing the flow of instructions.

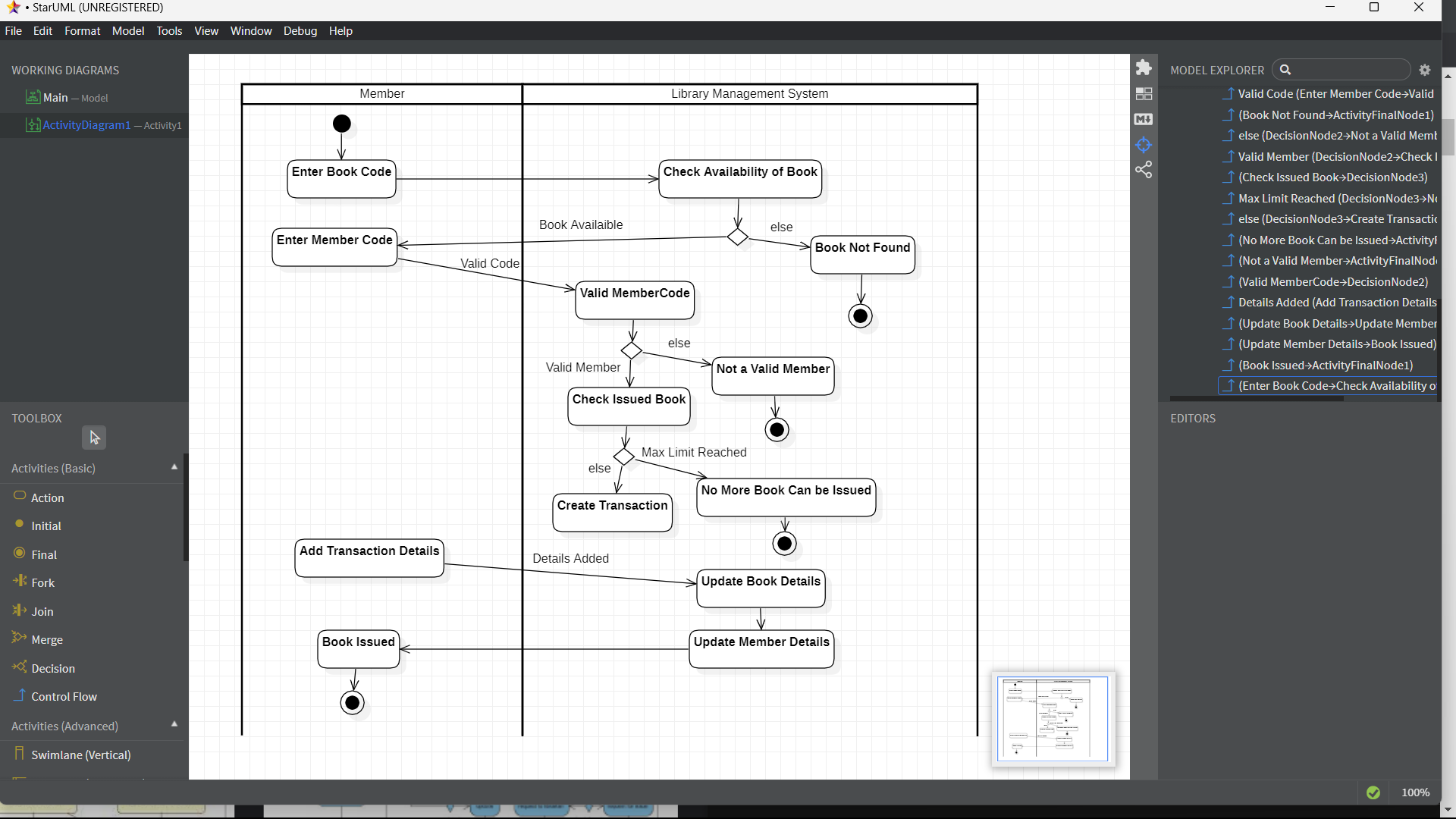
The Sequence Diagram for Library Management System represents the scenario and the messages that must be passed between objects. This is done for the scenario’s functionality to be realized. It’s an interaction diagram that shows how activities are carried out, including when and how messages are sent.



**d. Activity Diagram for** **Library Management System**

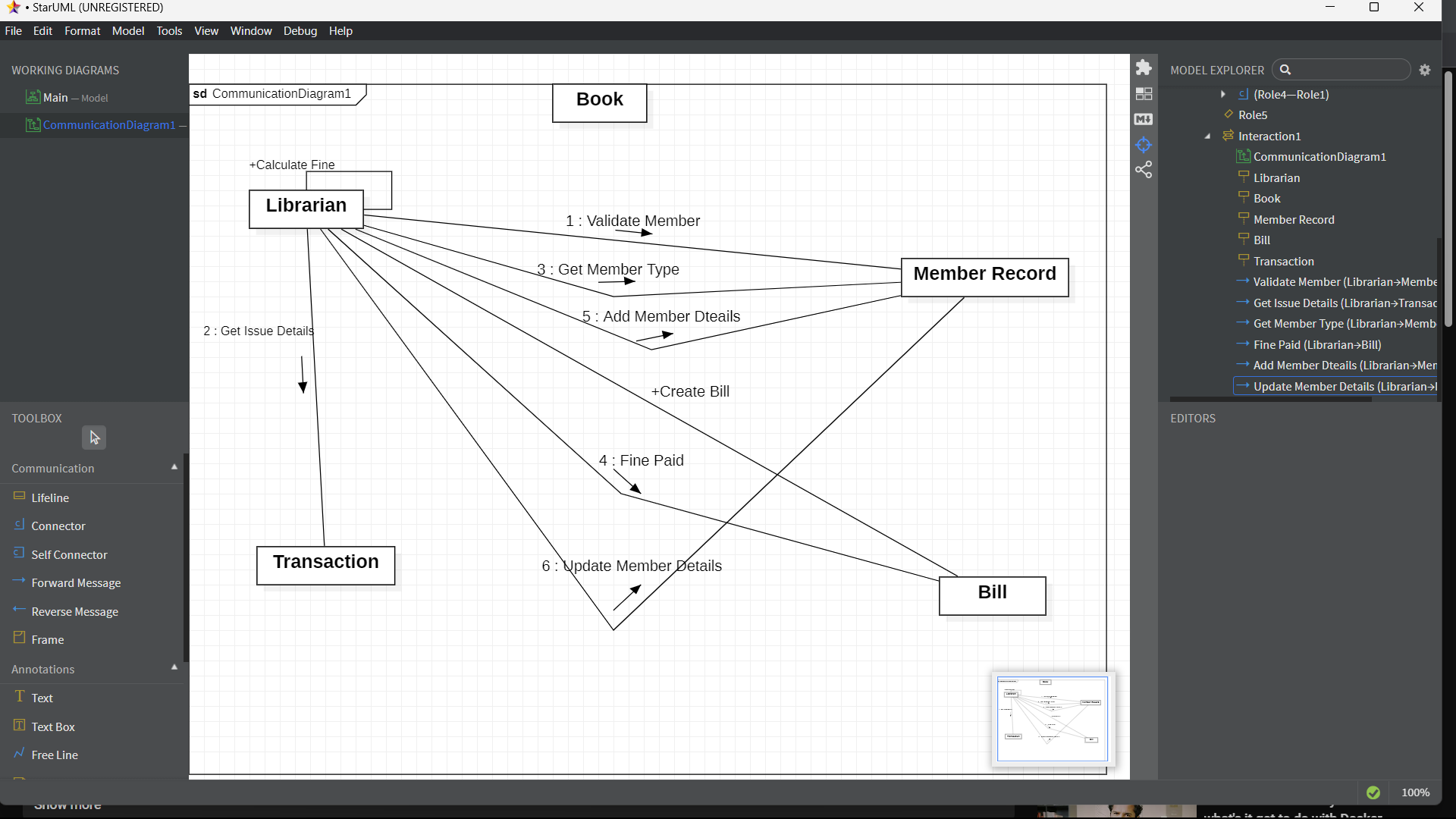
In UML, an activity diagram provides a view of the behaviour of a system by describing the sequence of actions in a process. Activity diagrams are similar to flowcharts because they show the flow between the actions in an activity; however, activity diagrams can also show parallel or concurrent flows and alternate flows.

Activity diagram for library management system which shows the flows between the activity of members, librarian, library system and books.



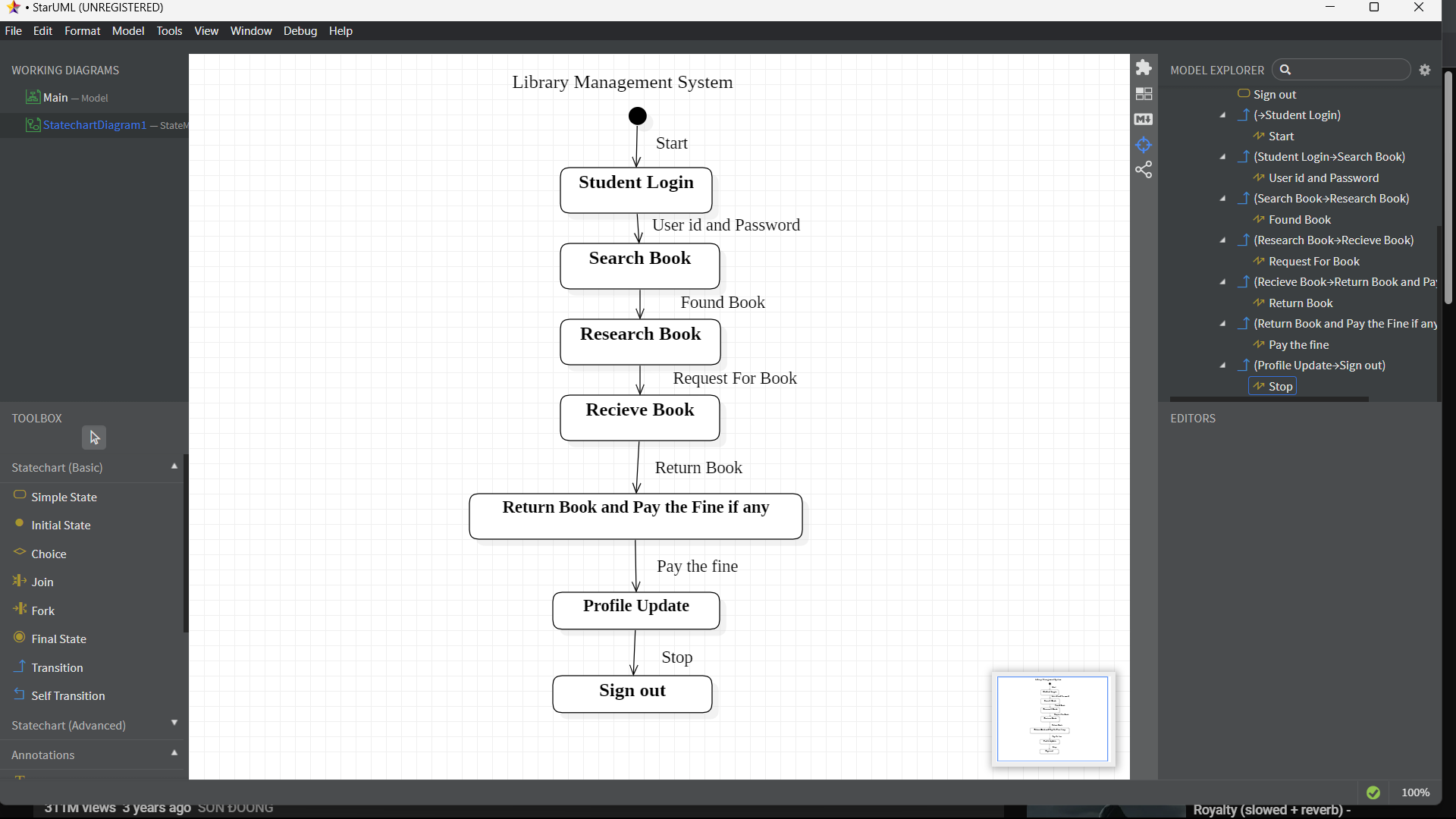
**e. Collaboration Diagram for Library Management System**

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modelling Language (UML). Developers can use these diagrams to portray the dynamic behaviour of a particular use case and define the role of each object.



**f. State Chart Diagram for Library Management System**

A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It’s a behavioural diagram and it represents the behaviour using finite state transitions. State diagrams are also referred to as State machines and State-chart Diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behaviour of a class in response to time and changing external stimuli. We can say that each and every class has a state but we don’t model every class using State diagrams. We prefer to model the states with three or more states.

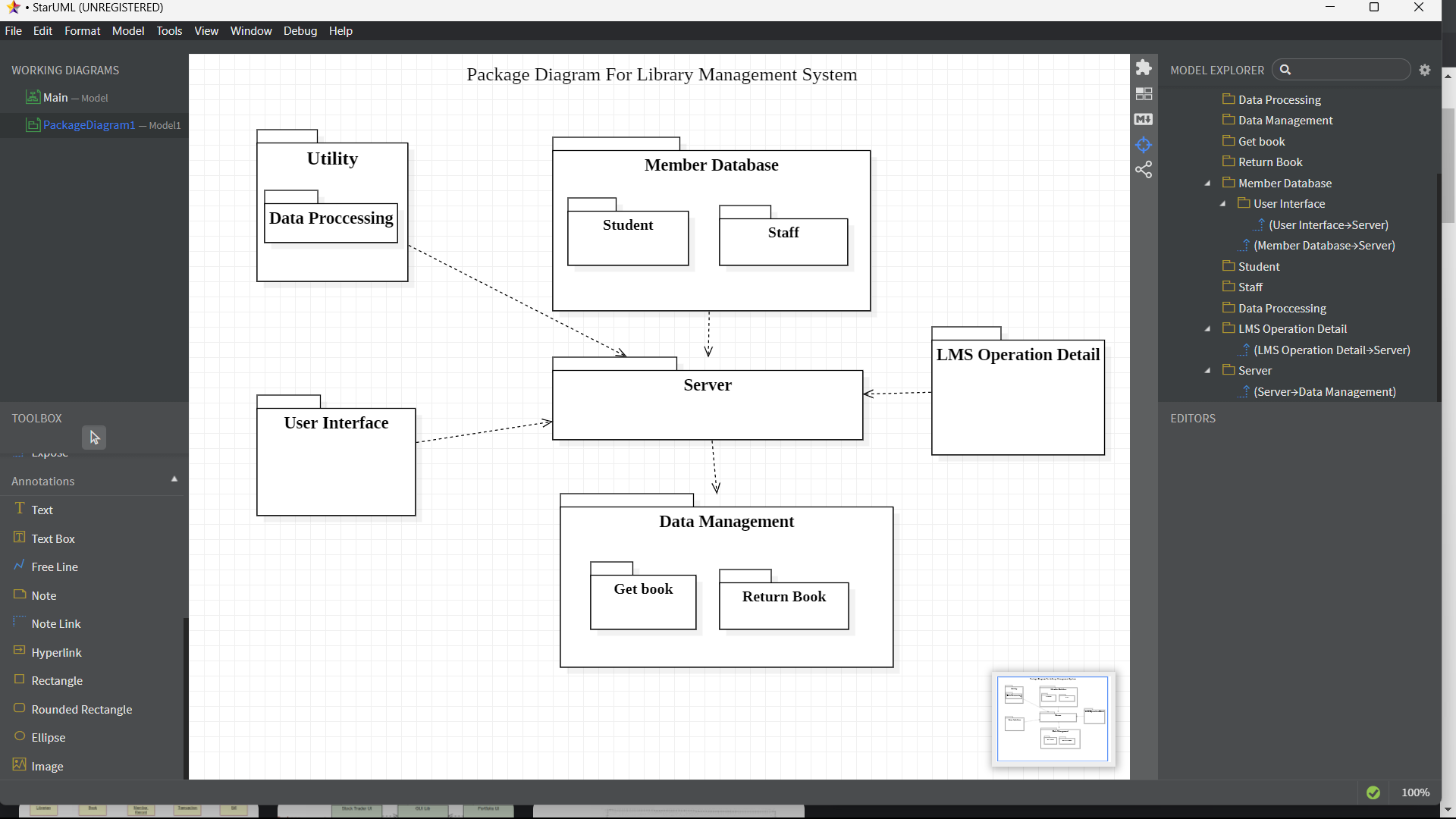


**g. Package Diagram for Library Management System**

A package is a group of related classes, interfaces, and other elements that are used together to provide a specific functionality or service. A package can also contain other sub-packages.

A package diagram for a library management system might include packages for the user interface, the database access layer, the library services, reporting, security, and configuration. Each of these packages would contain classes and other elements that are related to their specific area of functionality.

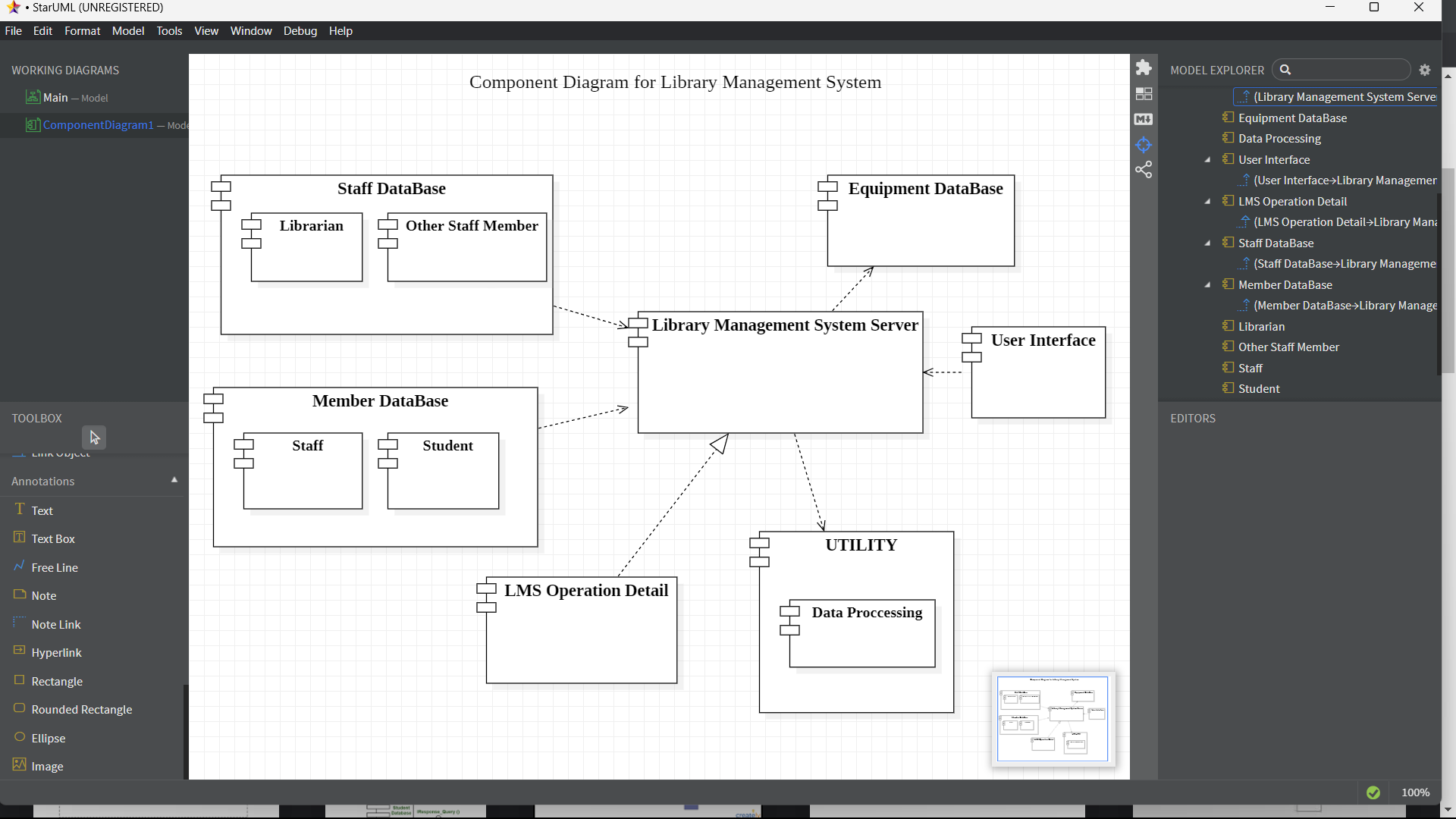
By using a package diagram, the developers of the library management system can see the relationships and dependencies between the different packages and can more easily manage and maintain the system. It can also help in identifying potential issues and facilitate communication among team members during the development process.



**h. Component Diagram for Library Management System**

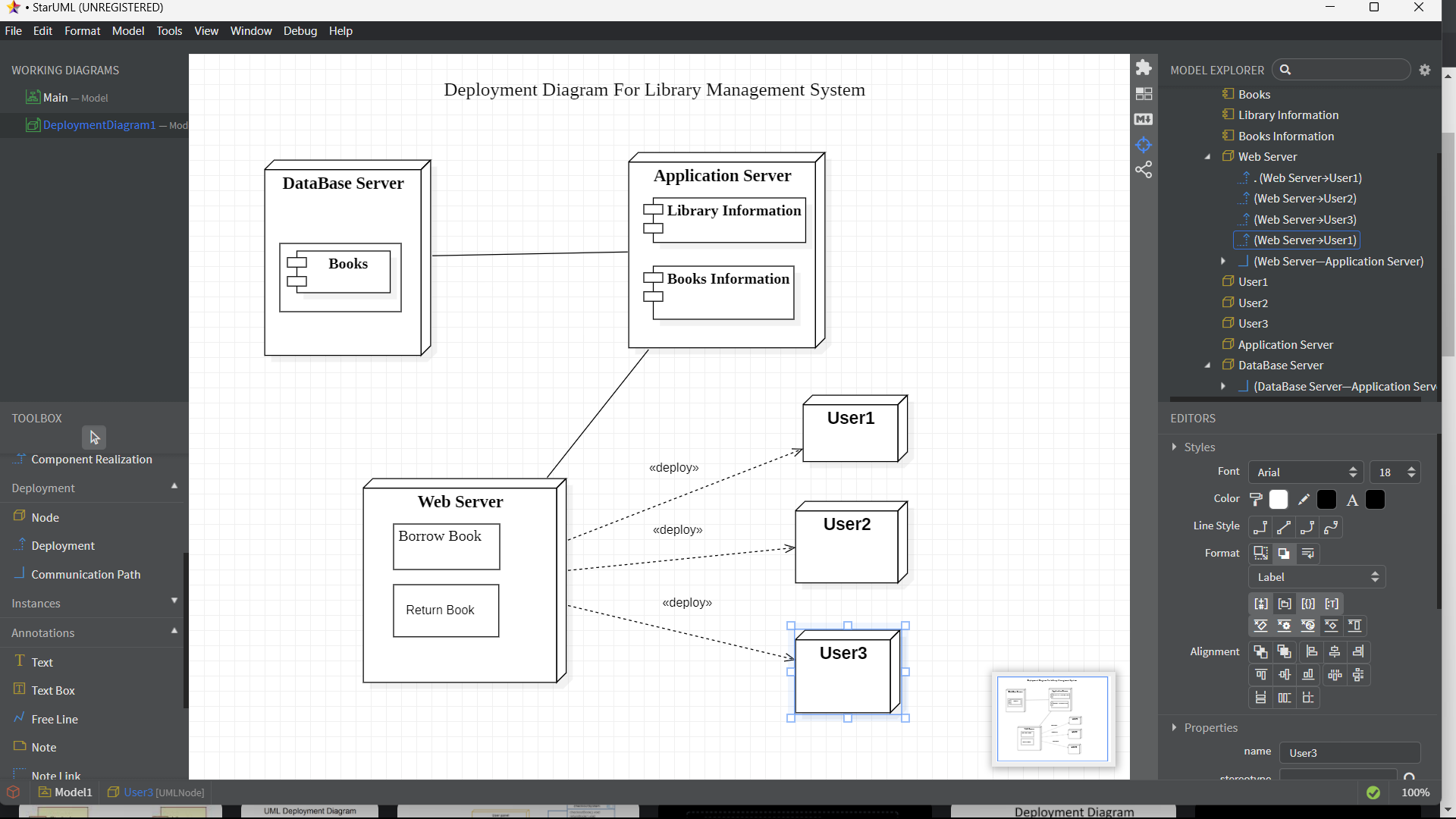
A component diagram is a type of UML diagram that shows the organization of components and their relationships in a software system. A component is a modular, self-contained unit of software that performs a specific function or service. Components can be implemented as classes, libraries, executables, or other software artifacts.

A component diagram for a library management system might include components for the user interface, the database access layer, the library services, reporting, security, and configuration. Each of these components would have well-defined interfaces and would interact with other components through these interfaces.



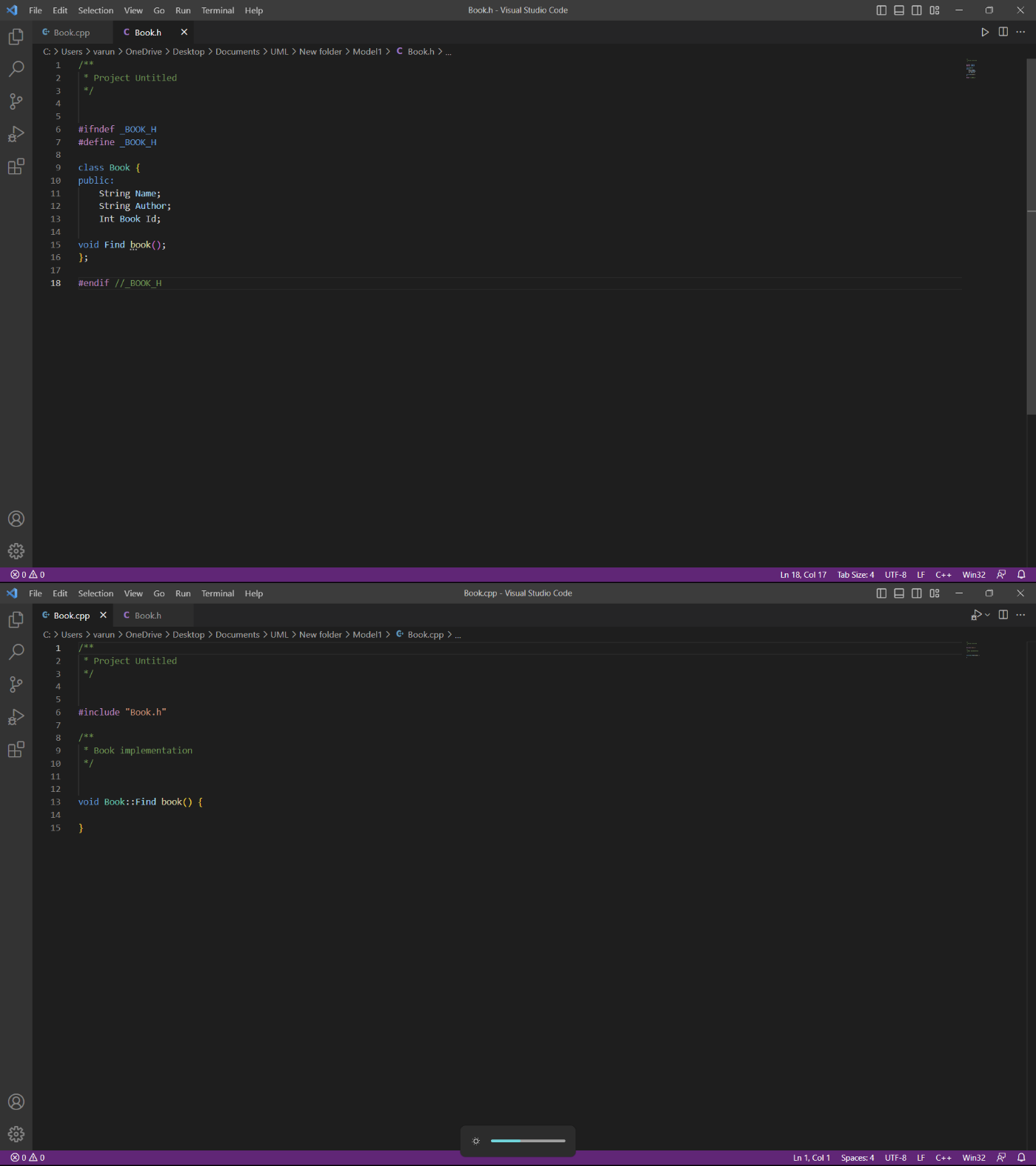
**i. Deployment Diagram for Library Management System**

A deployment diagram is a UML diagram type of the system that represents the execution architecture of the components of a system of the objects, including nodes or modes such as hardware or software execution environments or worlds, and the middleware connecting them. Diagram types mostly outline the logical components of a system. Deployment diagrams are typically or difficultly used to visualize or imagine the physical hardware and software of a system of the component. Using it you can understand how the system of the diagram will be physically deployed on the hardware.

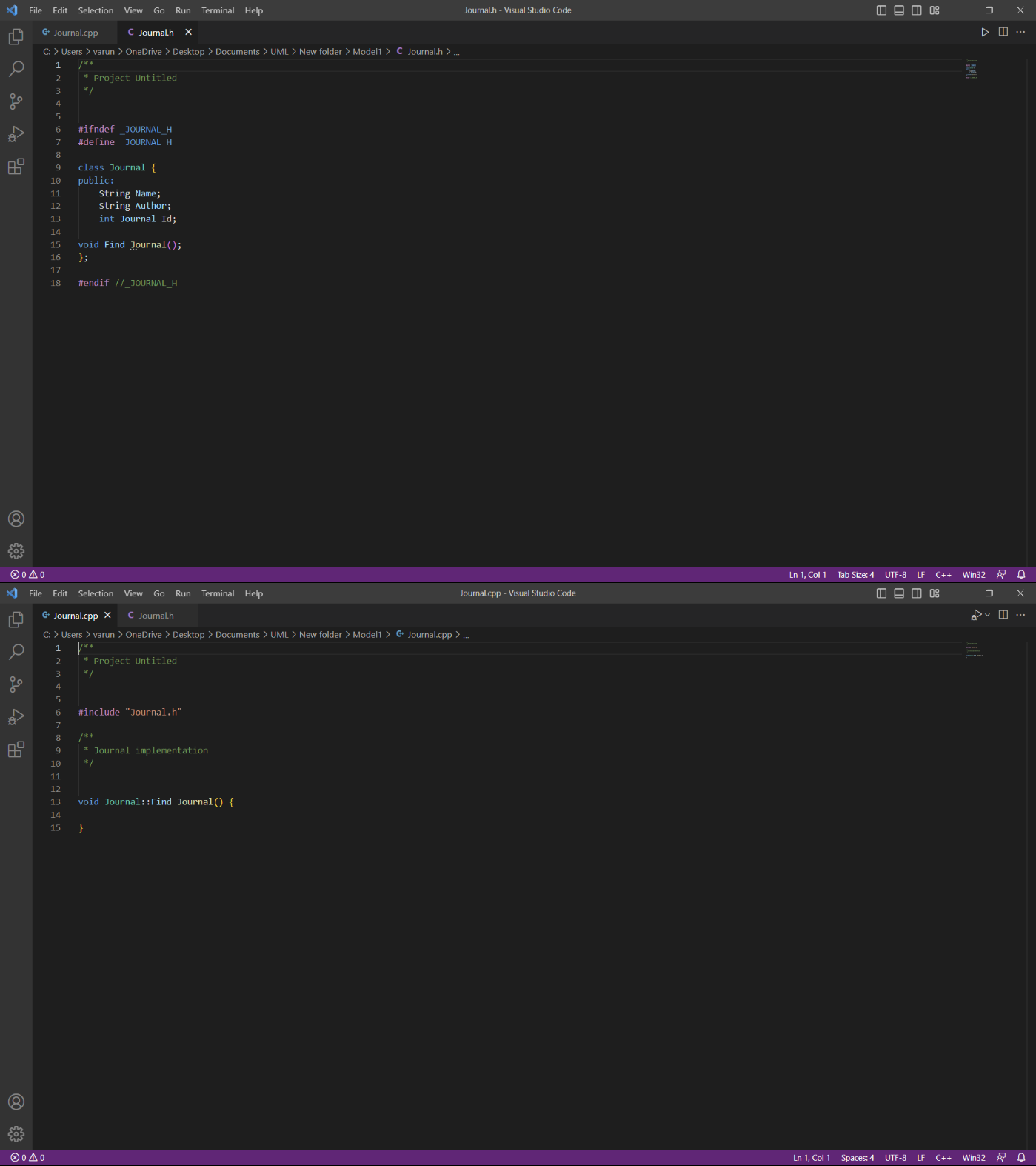


1. **Coding/Output**

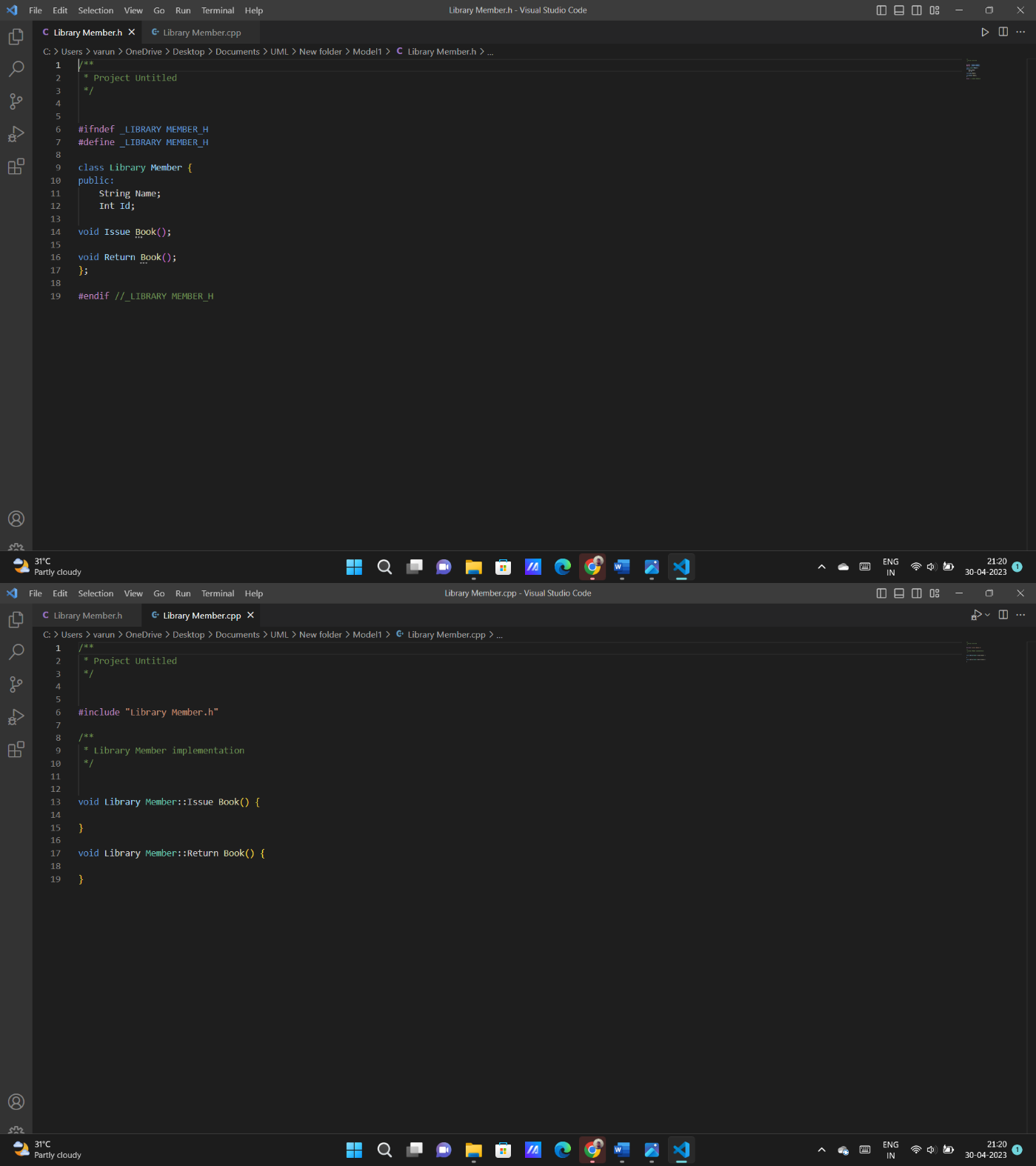
* **for Class “Book”**

****

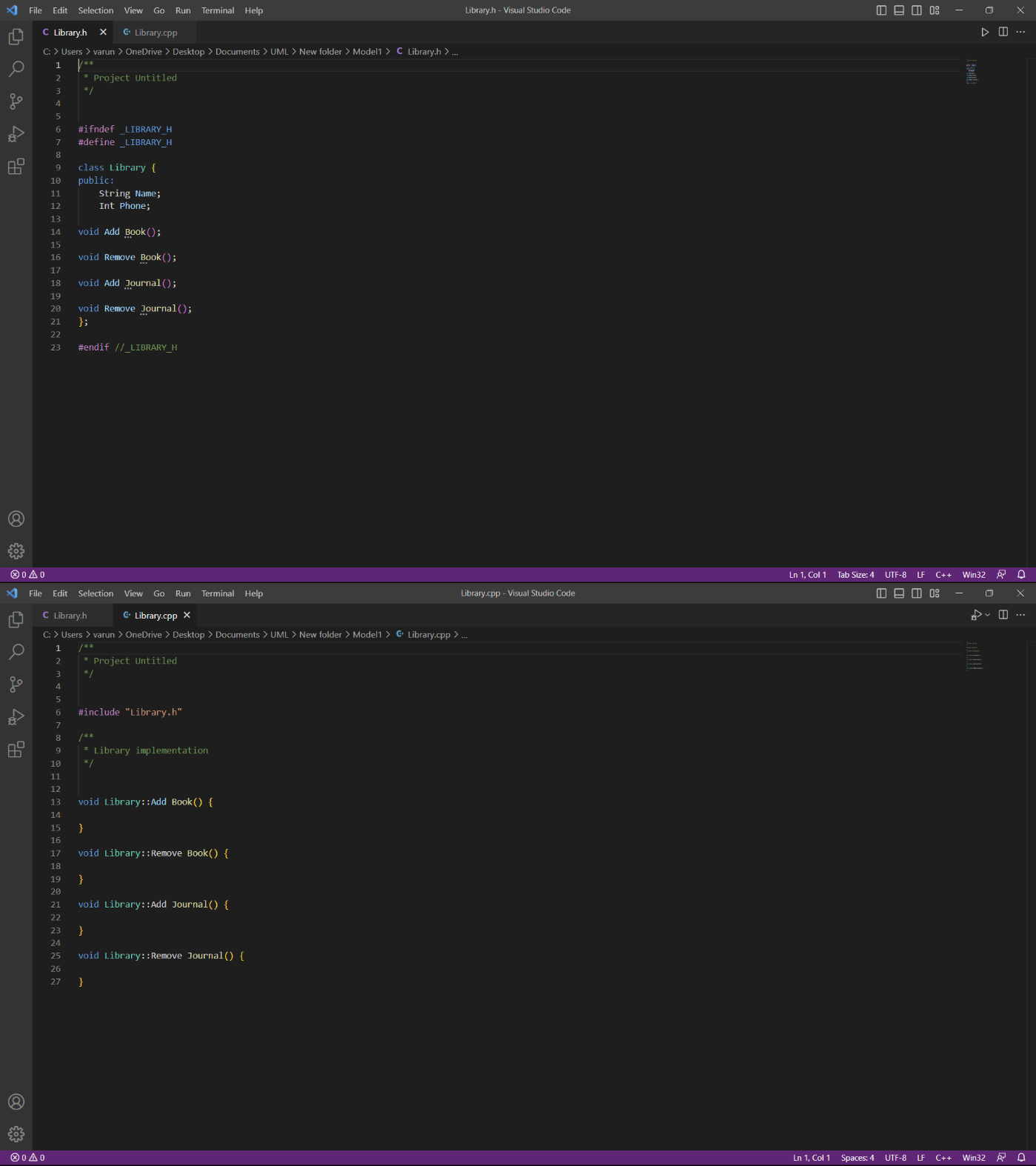
* **for Class “Journal”**

****

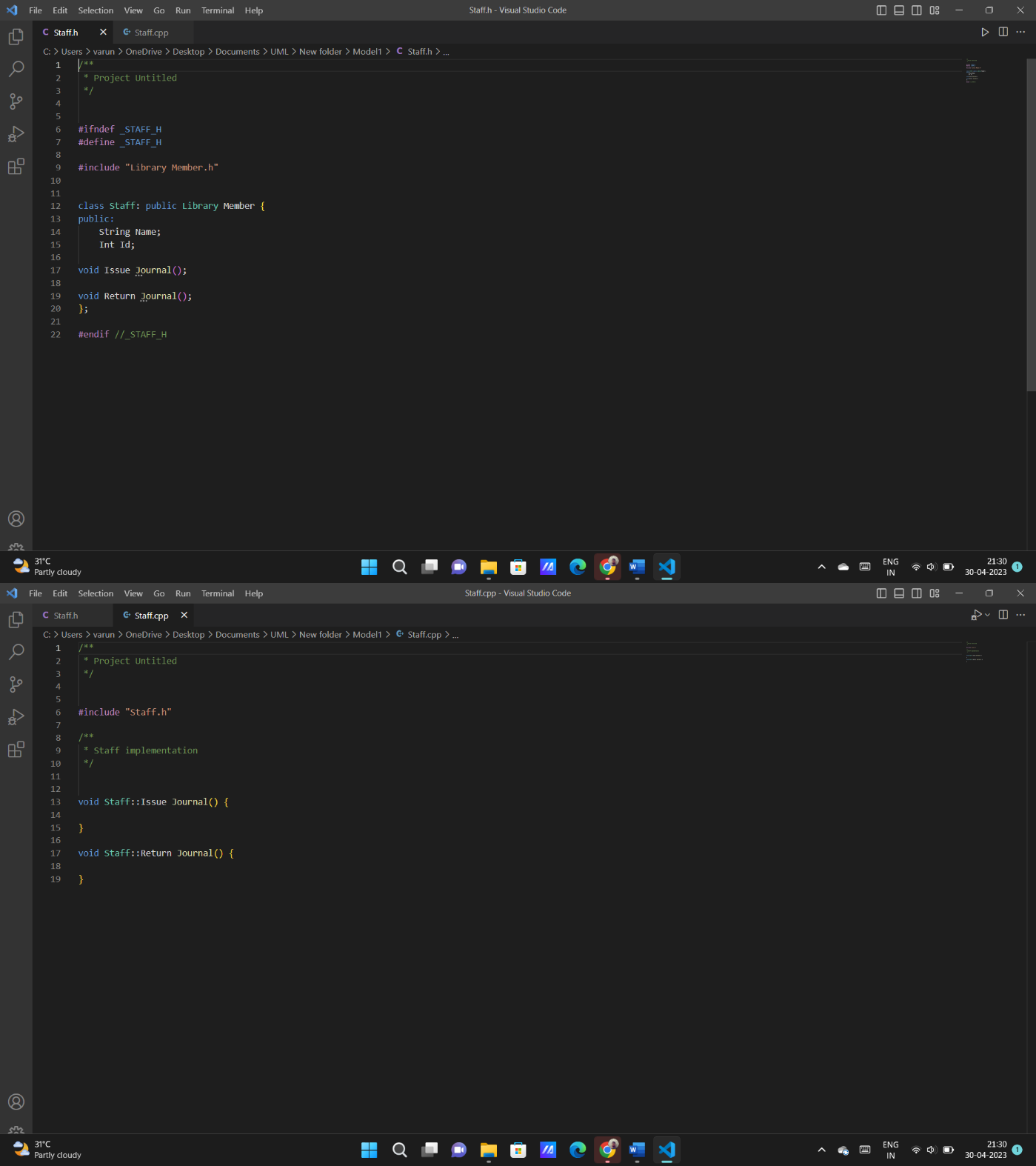
* **for class “Library Member”**

****

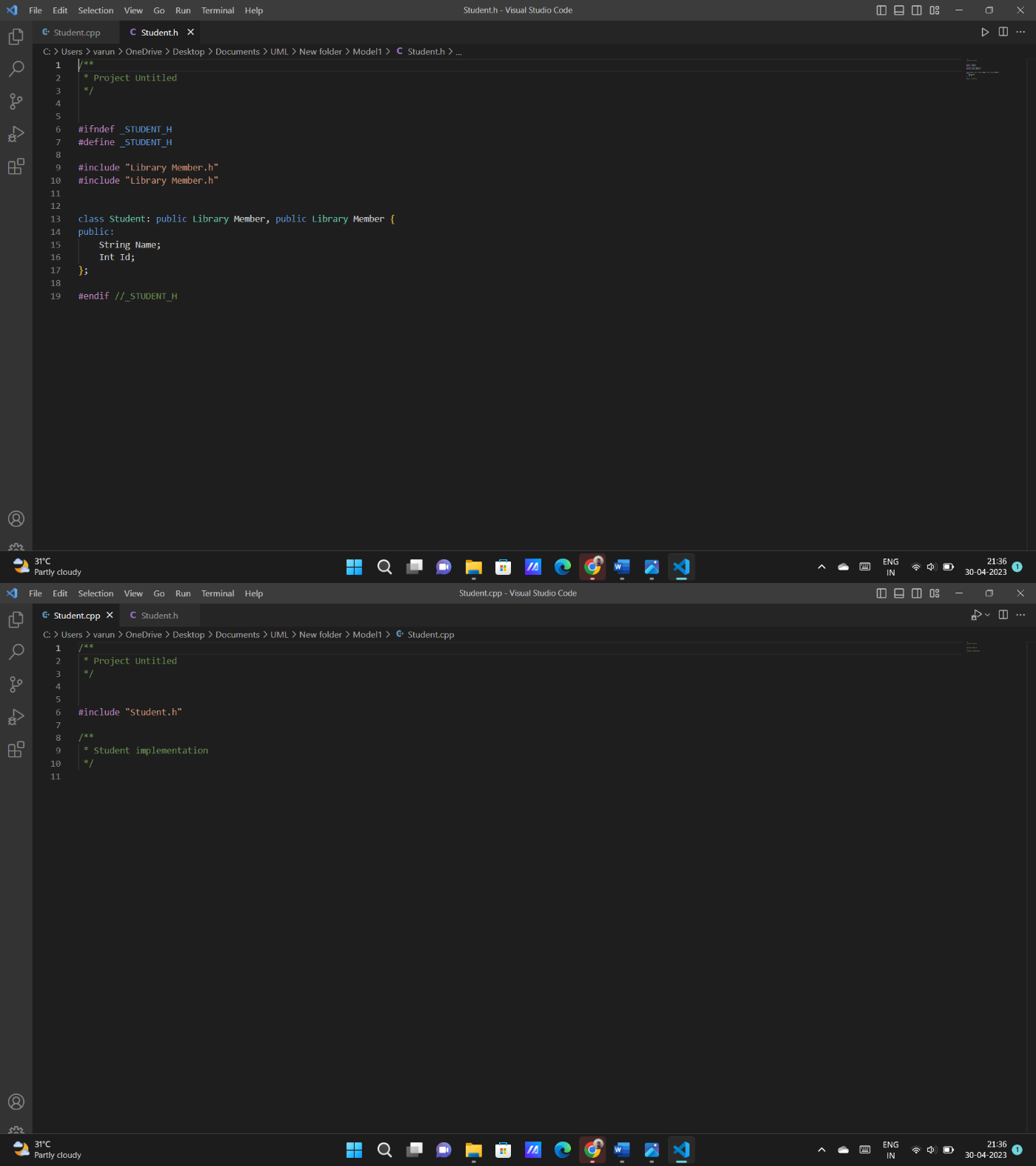
* **for class “Library”**

****

* **for class “Staff”**

****

* **for class “Student”**

****

**Result.**

Thus, the UML modelling for the project “Library Management System” is modelled successfully.

The project code for “Library Management System” has been run, compiled and executed successfully.

1. **References**

Here are some references we have used for our Library Management System project:

* **SRM UNIVERSITY LIBRARY**: We have noticed and reviewed our own library management system by issuing a book and taking collective information from the staff.
* **GOOGLE:** We have also used google for our information searching.
* **STAR UML:** We have used Star UML application for making UML diagram.