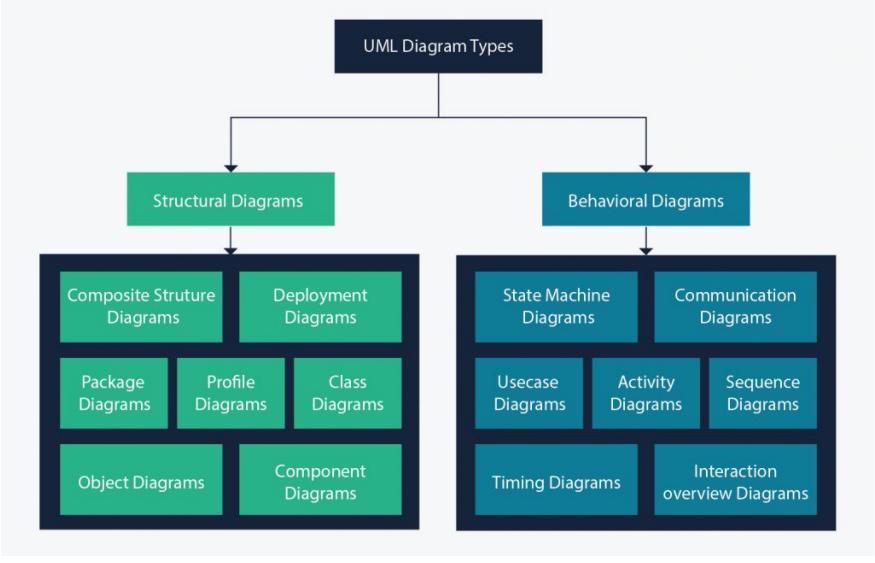
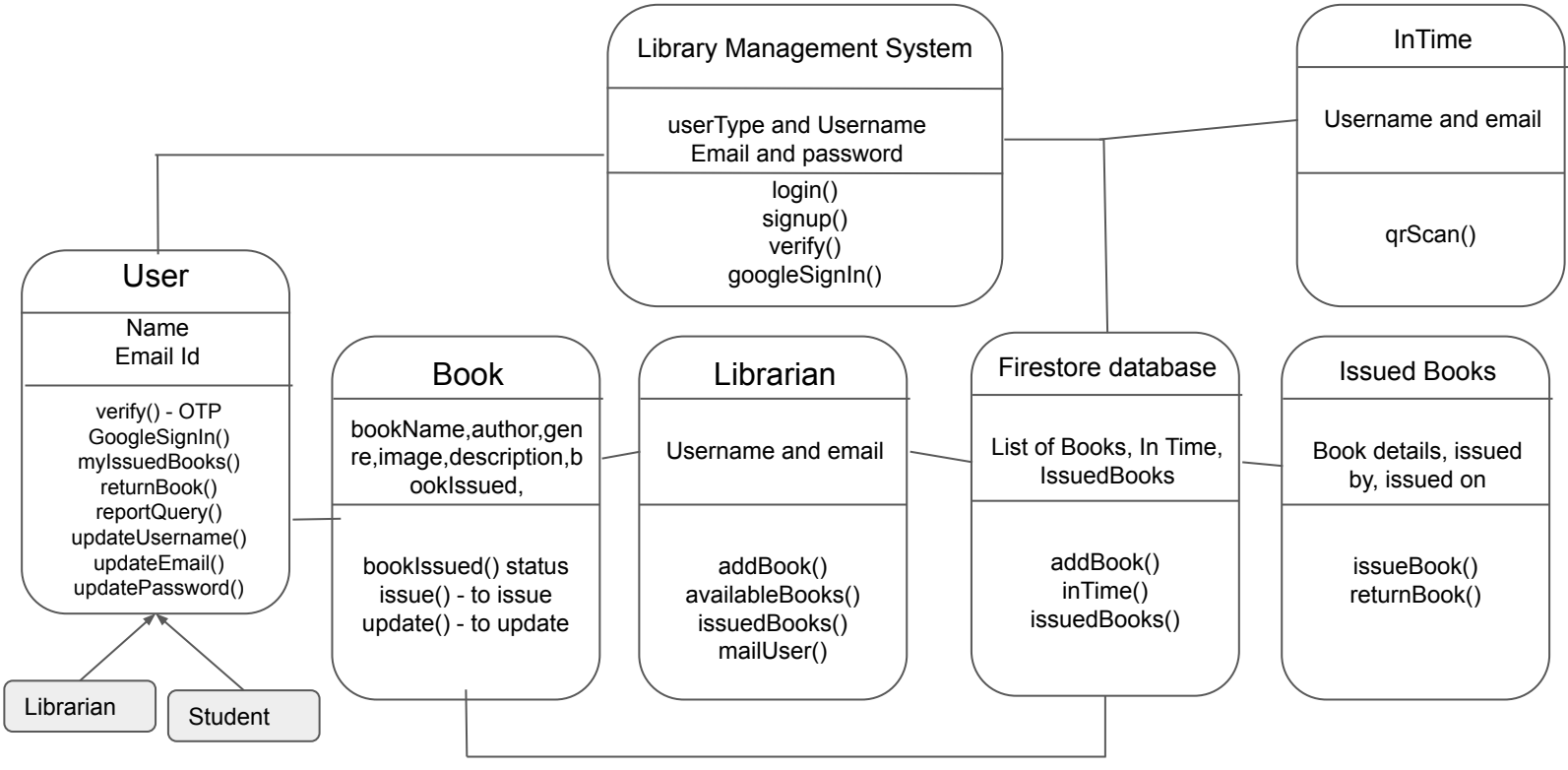
UML Diagrams

- A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of **visually representing a system** along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.
- It is based on **diagrammatic representations** of software components.
- It is used as a general purpose modeling language in field of software engineering.
- For example, activity diagrams, a type of UML diagram, can be used as a replacement for flowcharts. They provide both a more standardized way of modeling workflows as well as a wider range of features to improve readability and efficacy.
- There are in total 14 types of UML diagrams.
- Majorly Classified in two types:
  - **Structural Diagram**
  - **Behavioral Diagram**
- This are further classified into different sub diagrams:



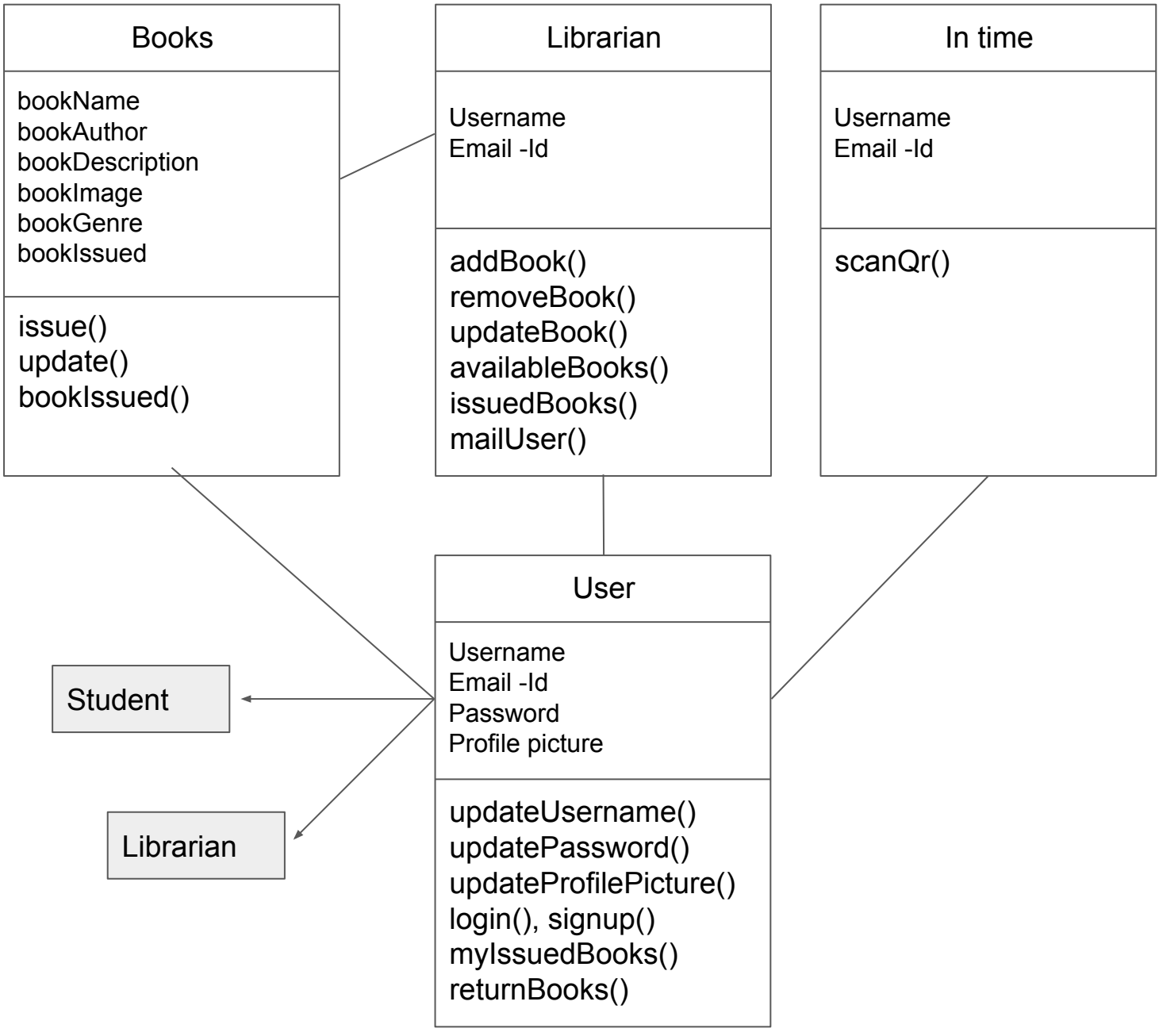
- **Structural Diagram:**
  - A structure diagram is a conceptual modeling tool used to document the different structures that make up a system such as a database or an application.
  - It shows the hierarchy or structure of the different components or modules of the system and shows how they connect and interact with each other.
  - It is a tool used to guide developers to ensure that all parts of the system work as intended in relation to all the other parts.
  - It is further classified into various other sub diagrams which are shown in the table present above.

- **Class Diagram:**
  - Class diagrams are generally used for conceptual modeling of static view of a software application, and for modeling translating models into programming code in a detailed manner.
  - It is widely used diagram during the development of software and also in data modeling.
  - It is used to show classes, relationships among them, interface, association, etc.
  - Aggregation and Multiplicity are two important points that need to take into consideration while designing a Class Diagram.

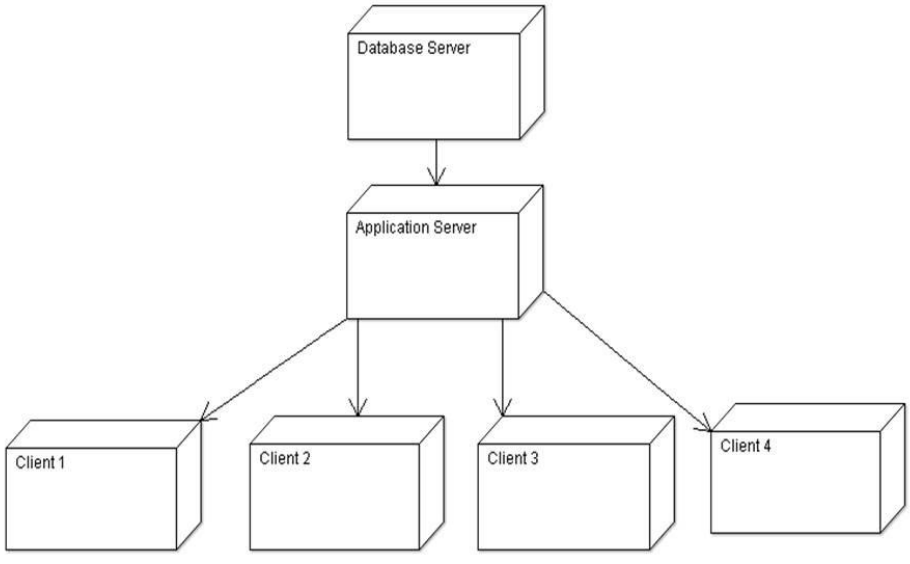


- **Object Diagram:**
  - Object Diagrams specifies the models needed to create for the software.
  - As it is seen that there are separate objects for each model like book, librarian, in time, etc
  - In every object of model there exist several functions which are used to complete the action take by that model

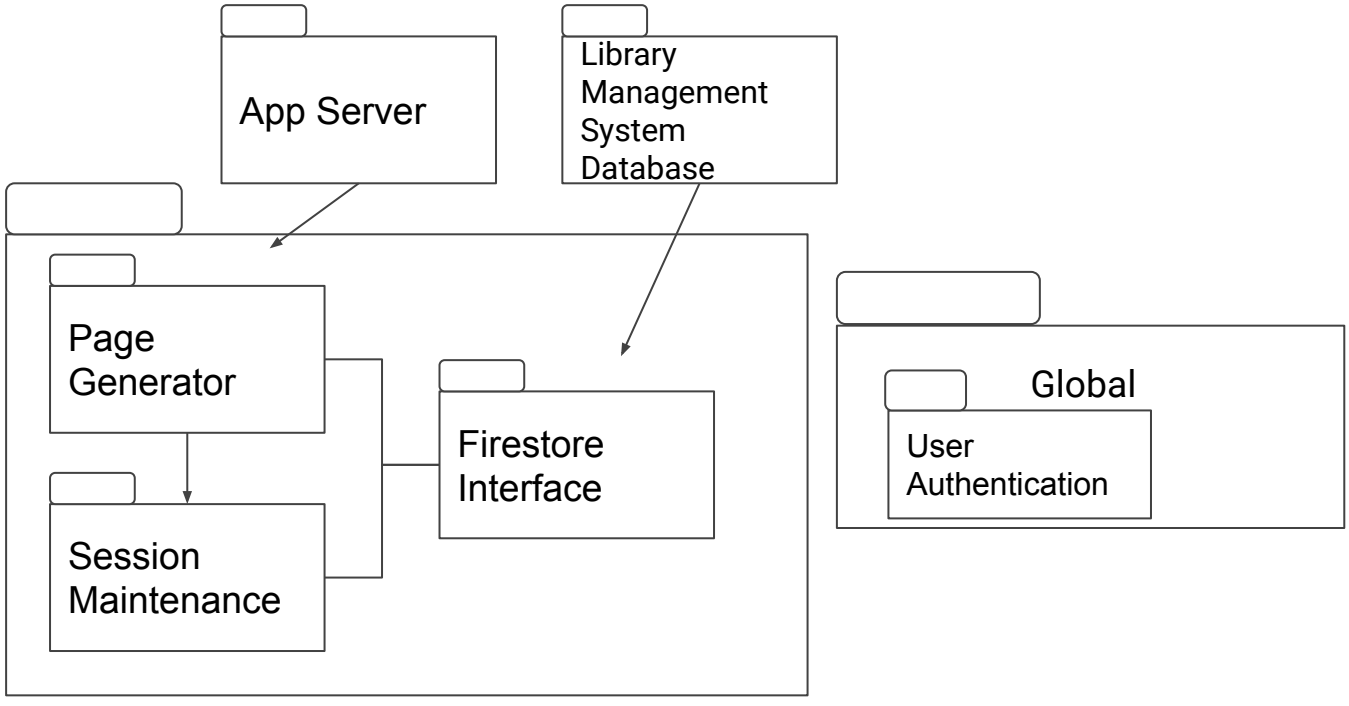
Diagram on next page



- **Deployment Diagram:** Theory provided in 4+1 Views



- **Package Diagram:**
  - Package diagrams are structural diagrams used to show the organization and arrangement of various model elements in the form of packages.
  - A package is a grouping of related UML elements, such as diagrams, documents, classes, or even other packages.



- **Components Diagram:**
  - The Component diagram of Library Management System which shows components, provided and required interfaces, ports, and relationships between the Issues, Student, Librarian, Member and Address.
  - This type of diagrams is used in Component-Based Development (CBD) to describe systems with Service-Oriented Architecture (SOA).
  - Library Management System UML component diagram, describes the organization and wiring of the physical components in a system