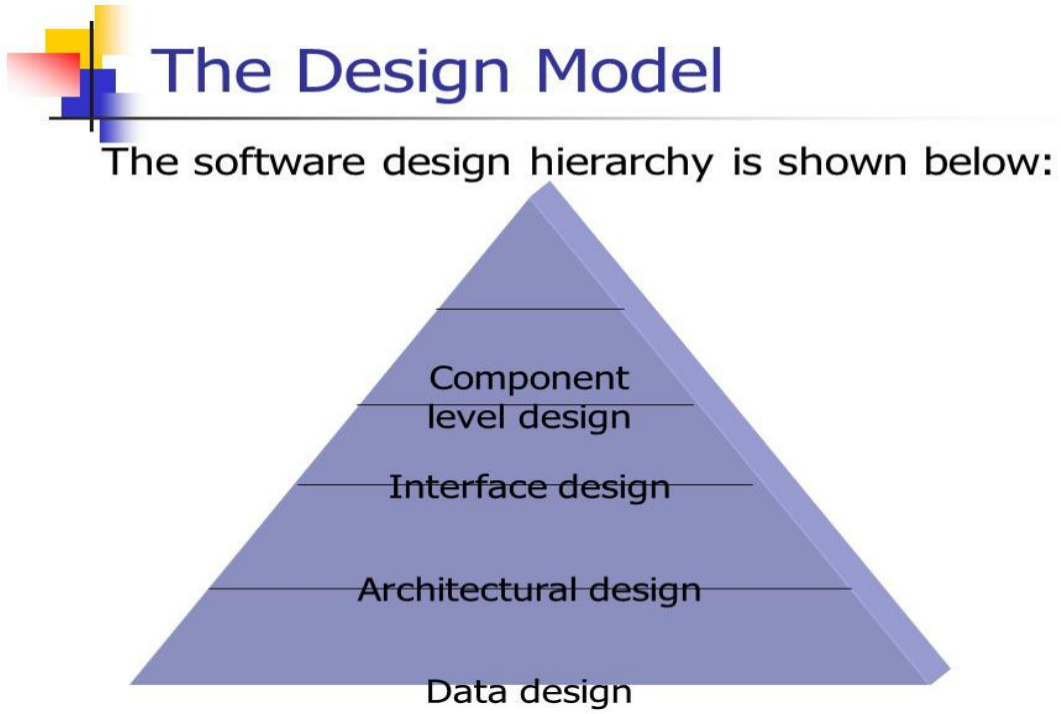
Design Modeling in Software Engineering



Design modeling in software engineering represents the features of the software that helps engineer to develop it effectively, the architecture, the user interface, and the component level detail.

Designing a model is an important phase and is a multi-process that represent the data structure, program structure, interface characteristic, and procedural details. It is mainly classified into four categories – Data design, architectural design, interface design, and component-level design.

- **Data design:** It represents the data objects and their interrelationship in an entity-relationship diagram. Entity-relationship consists of information required for each entity or data objects as well as it shows the relationship between these objects. It shows the structure of the data in terms of the tables. It shows three type of relationship – One to one, one to many, and many to many. In one to one relation, one entity is connected to another

entity. In one many relation, one Entity is connected to more than one entity. un many to many relations one entity is connected to more than one entity as well as other entity also connected with first entity using more than one entity.

- **Architectural design:** It defines the relationship between major structural elements of the software. It is about decomposing the system into interacting components. It is expressed as a block diagram defining an overview of the system structure – features of the components and how these components communicate with each other to share data. It defines the structure and properties of the component that are involved in the system and also the inter-relationship among these components.
- **User Interfaces design:** It represents how the Software communicates with the user i.e. the behavior of the system. It refers to the product where user interact with controls or displays of the product. For example, Military, vehicles, aircraft, audio equipment, computer peripherals are the areas where user interface design is implemented. UI design becomes efficient only after performing usability testing. This is done to test what works and what does not work as expected. Only after making the repair, the product is said to have an optimized interface.
- **Component level design:** It transforms the structural elements of the software architecture into a procedural description of software components. It is a perfect way to share a large amount of data. Components need not be concerned with how data is managed at a centralized level i.e. components need not worry about issues like backup and security of the data.

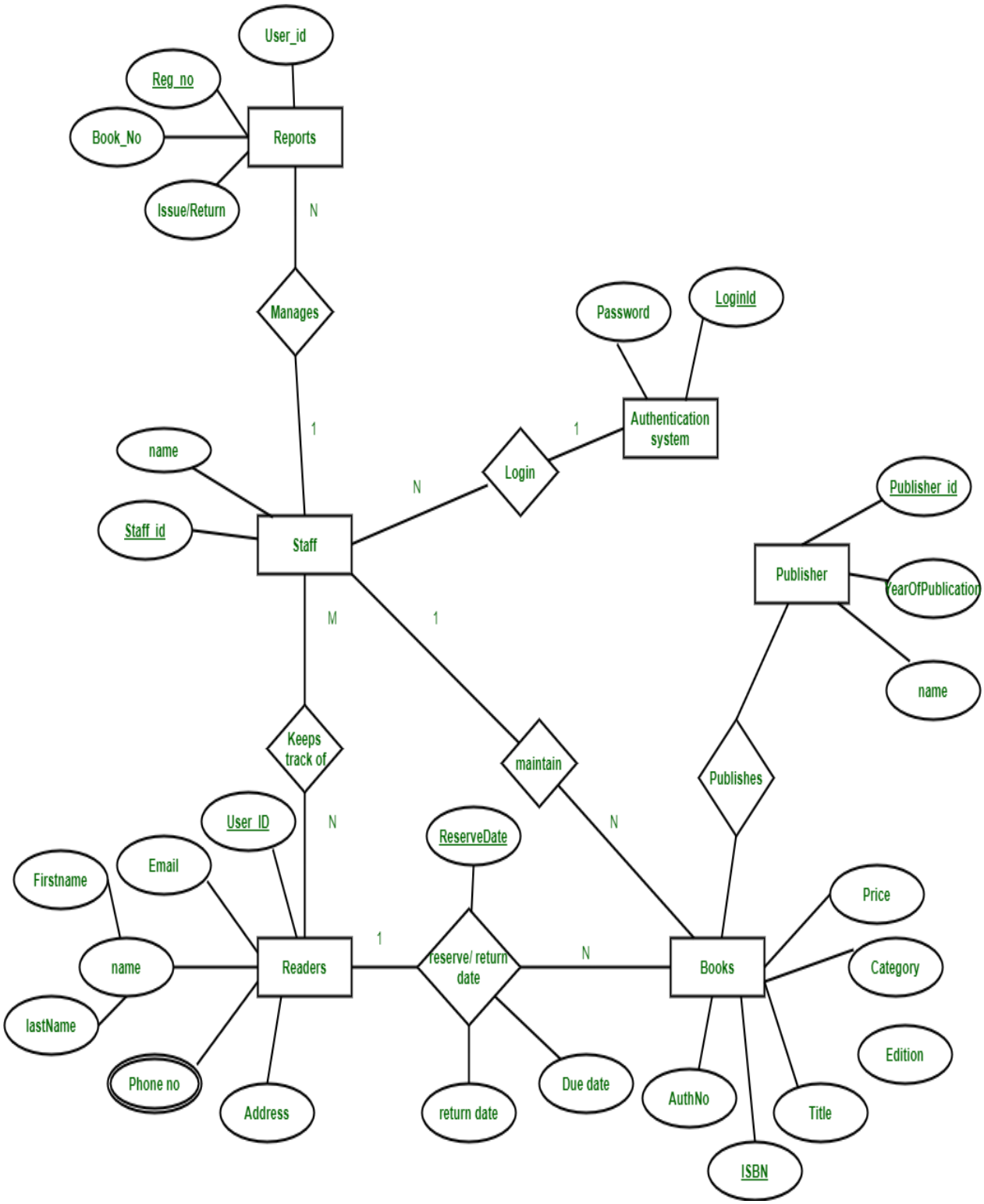
ER Diagram is known as Entity-Relationship Diagram, it is used to analyze the structure of the Database. It shows relationships between entities and their attributes. An ER Model provides a means of communication.
The Library Management System database keeps track of readers with the following considerations –

- The system keeps track of the staff with a single point authentication system comprising login Id and password.
- Staff maintains the book catalog with its ISBN, Book title, price(in INR), category(novel, general, story), edition, author Number and details.
- A publisher has publisher Id, Year when the book was published, and name of the book.
- Readers are registered with their user_id, email, name (first name, last name), Phone no (multiple entries allowed), communication address. The staff keeps track of readers.
- Readers can return/reserve books that stamps with issue date and return date. If not returned within the prescribed time period, it may have a due date too.
- Staff also generate reports that has readers id, registration no of report, book no and return/issue info.

Symbols Used in ER Model

ER Model is used to model the logical view of the system from a data perspective which consists of these symbols:

- **Rectangles:** Rectangles represent Entities in the ER Model.
- **Ellipses:** Ellipses represent Attributes in the ER Model.
- **Diamond:** Diamonds represent Relationships among Entities.
- **Lines:** Lines represent attributes to entities and entity sets with other relationship types.
- **Double Ellipse:** Double Ellipses represent Multi-Valued Attributes.
- **Double Rectangle:** Double Rectangle represents a Weak Entity.



ER Diagram of Library Management System

This Library ER diagram illustrates key information about the Library, including entities such as staff, readers, books, publishers, reports, and authentication system. It allows for understanding the relationships between entities.

Entities and their Attributes –

- **Book Entity :** It has authno, isbn number, title, edition, category, price. ISBN is the Primary Key for Book Entity.
- **Reader Entity :** It has UserId, Email, address, phone no, name. Name is composite attribute of firstname and lastname. Phone no is multi valued attribute. UserId is the Primary Key for Readers entity.
- **Publisher Entity :** It has PublisherId, Year of publication, name. PublisherID is the Primary Key.
- **Authentication System Entity :** It has LoginId and password with LoginID as Primary Key.
- **Reports Entity :** It has UserId, Reg_no, Book_no, Issue/Return date. Reg_no is the Primary Key of reports entity.
- **Staff Entity :** It has name and staff_id with staff_id as Primary Key.
- **Reserve/Return Relationship Set :** It has three attributes: Reserve date, Due date, Return date.

Relationships between Entities –

- A reader can reserve N books but one book can be reserved by only one reader. The relationship 1:N.
- A publisher can publish many books but a book is published by only one publisher. The relationship 1:N.
- Staff keeps track of readers. The relationship is M:N.
- Staff maintains multiple reports. The relationship 1:N.
- Staff maintains multiple Books. The relationship 1:N.
- Authentication system provides login to multiple staffs. The relation is 1:N.