DATA FLOW DIAGRAMS

1. A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.

2. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

3. It shows how data enters and leaves the system, what changes the information, and where data is stored.

4. The objective of a DFD is to show the scope and boundaries of a system as a whole.

5. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system.

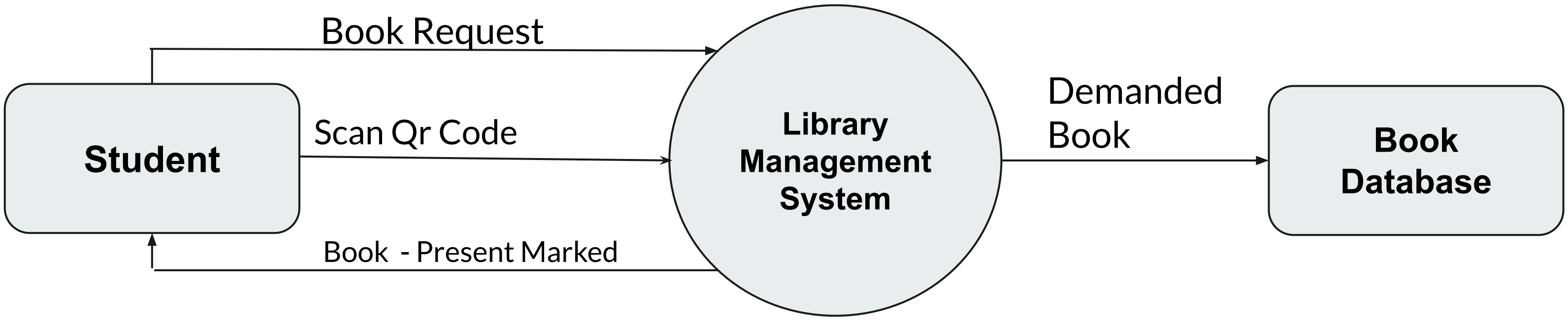
6. It can even be drawn to represent the system of different levels of abstraction.

7. Higher-level DFDs are partitioned into low levels-hacking more information and functional elements.

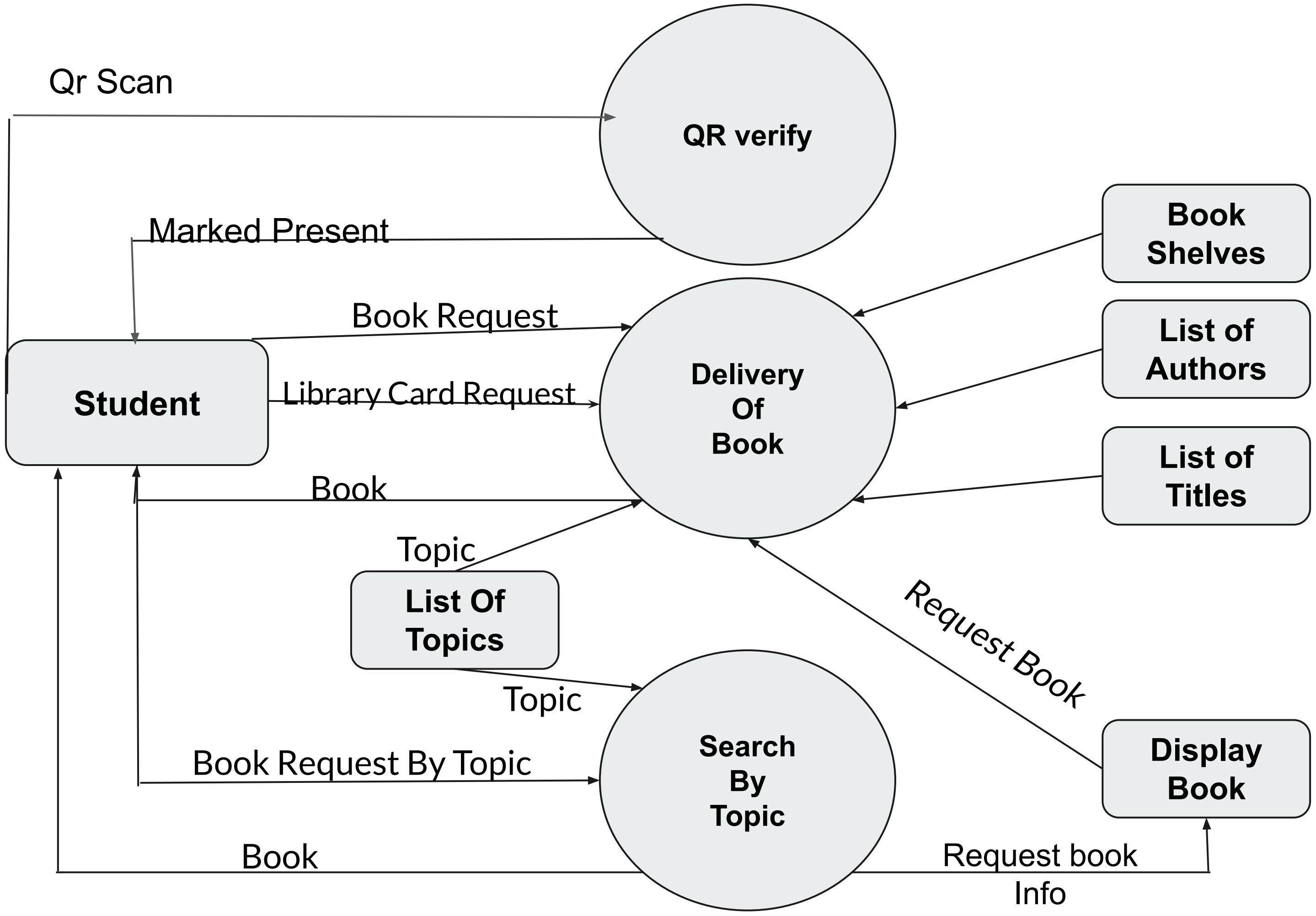
8. Levels in DFD are numbered 0, 1, 2 or beyond

9. As the level increases the DFD becomes more detailed and more complex.

- **Level 0: All Modules:**
  - It’s designed to be an abstraction view, showing the system as a single process with its relationship to external entities.
  - It represents the entire system as a single bubble with input and output data indicated by incoming/outgoing arrows.
  - It is also known as context diagrams.
  - Level 0 consists of overview of the flow of how the application will work from the client side.
  - The below given figure indicates that the student may request for book, scan qr code and in return may get the book issued if it is available with the librarian and get marked present at the library.

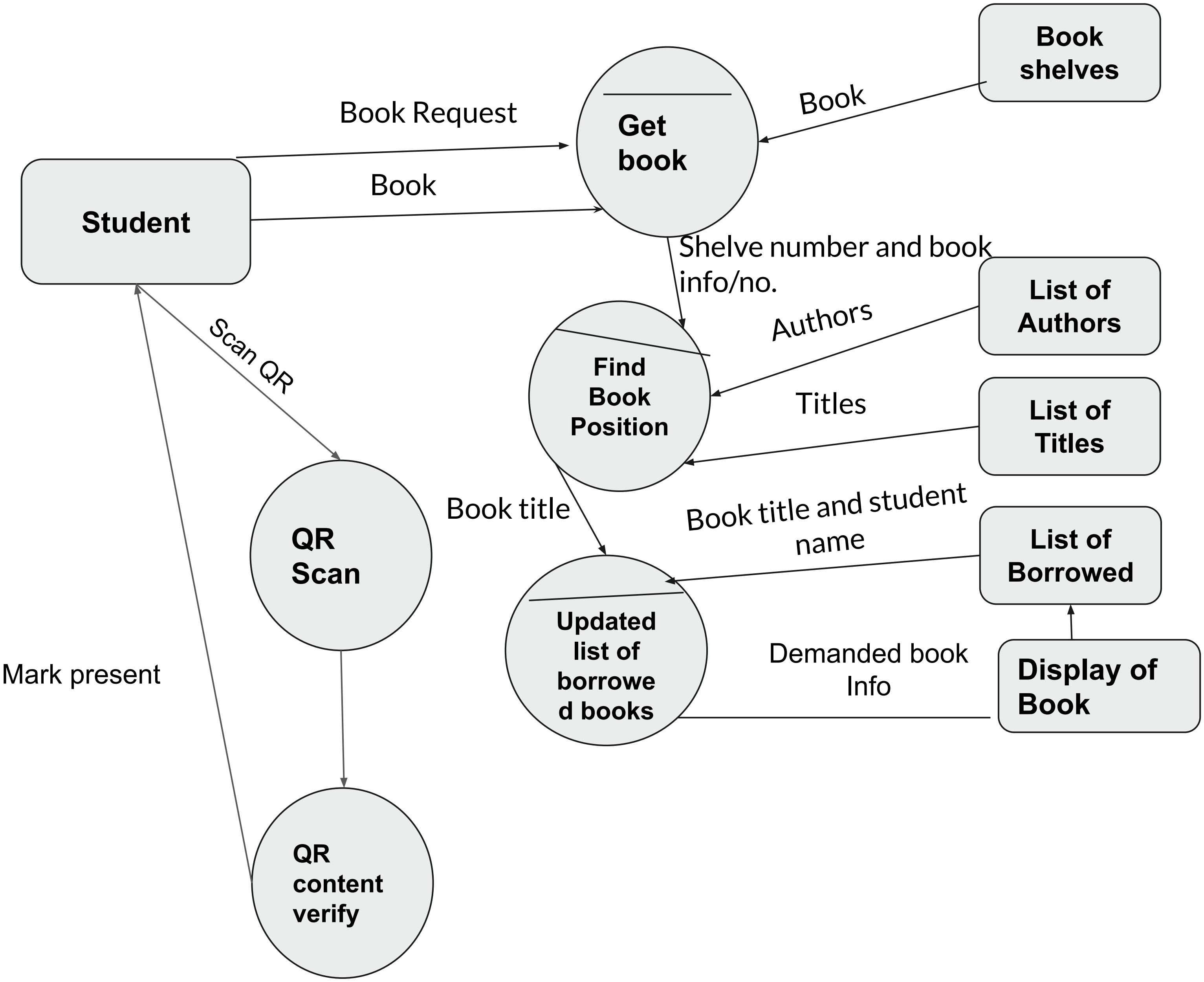


- **Level 1: All Functions:**
    - The context diagram is decomposed into multiple bubbles/processes
    - In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses.
    - In this we try to show the important process that need to be performed, or are performed by the user, hence more detailed than level 0 diagram.
    - List of authors, List of Titles, List of Topics, the bookshelves from which books can be located are some information that is required for these processes.
    - **Data store** is used to represent this type of information.
    - In level 1, the **Book Request** is present in more detailed diagrams
        - A user can request the book in many a several ways, here we have shown **Search By Topics.**
        - On search the application creates a **pull request** to the backend for the book with **query=Topic Name**
        - The book is then fetched from the backend and is displayed to the user.
    - **QR Scan:**
        - In this user scans the qr code present at the library using the inbuilt scanner.
        - Then if the content in qr code matches with the one present in backend to verify, user is marked present in the library and the date & time of capturing along with username gets stored in firestore model..



- **Level 2 All Parameters:**
  - 2-level DFD goes one step deeper into parts of 1-level DFD.
  - It can be used to plan or record the specific/necessary detail about the system's functioning
  - In Level-2 Diagram we try to elaborate processes associated with the backend like fetching the book to display from backend.
  - As you can see, when a user requests for the display of the book:
  - The First process begins to find the position of the book in the backend.
  - We can search the book in the **BookShelves** depending on the request provided by the user, it may be topic name, authors name,etc.
  - After finding the book, we display it to the user, wherein he/she can get the option to borrow the book.
  - If user borrows the book,
    - We then have to update the borrow list and update the frontend as well.
    - We have to mark **bookIssued()** boolean to true, in the backend to mention that the book is borrowed and is not displayed on the **Book Shelves.**
  - Else, we may just stop displaying the book after a stipulated time.
  - Along with that as we scan the qr code, we get the content present in it and then it verifies if it matches it marks as present else keeps scanning, and pops an alert that wrong qr code scanned.

Diagram on the next Page


Book shelves
Book Request
Get book
Book
Book
Student
Shelve number and book info/no.
List of Authors
Authors
Scan QR
Find Book Position
Titles
List of Titles
Book title
Book title and student name
List of Borrowed
QR Scan
Updated list of borrowed books
Demanded book Info
Display of Book
Mark present
QR content verify