**Requirements :**

Functional Requirements - What must it do?

The system must …

* Register users
* enable users to upload/book items
* allow users to select the needed items from the listed items via checkbox
* allow users to include a description and contact information while uploading an item.
* provide users with the ability to add chosen items to the cart
* facilitate users to contact the uploader with the mentioned number or e-mail
* enable users to review and rate items.

Non-functional Requirements - How should it do it?

The system should be…

* available 24/7
* Secured to prevent data theft
* Smooth, so that it does not cause inconvenience to the user and does not take more than 3 seconds to load.

**Use Cases :**

Title : TakeItFree

Primary Actor : User

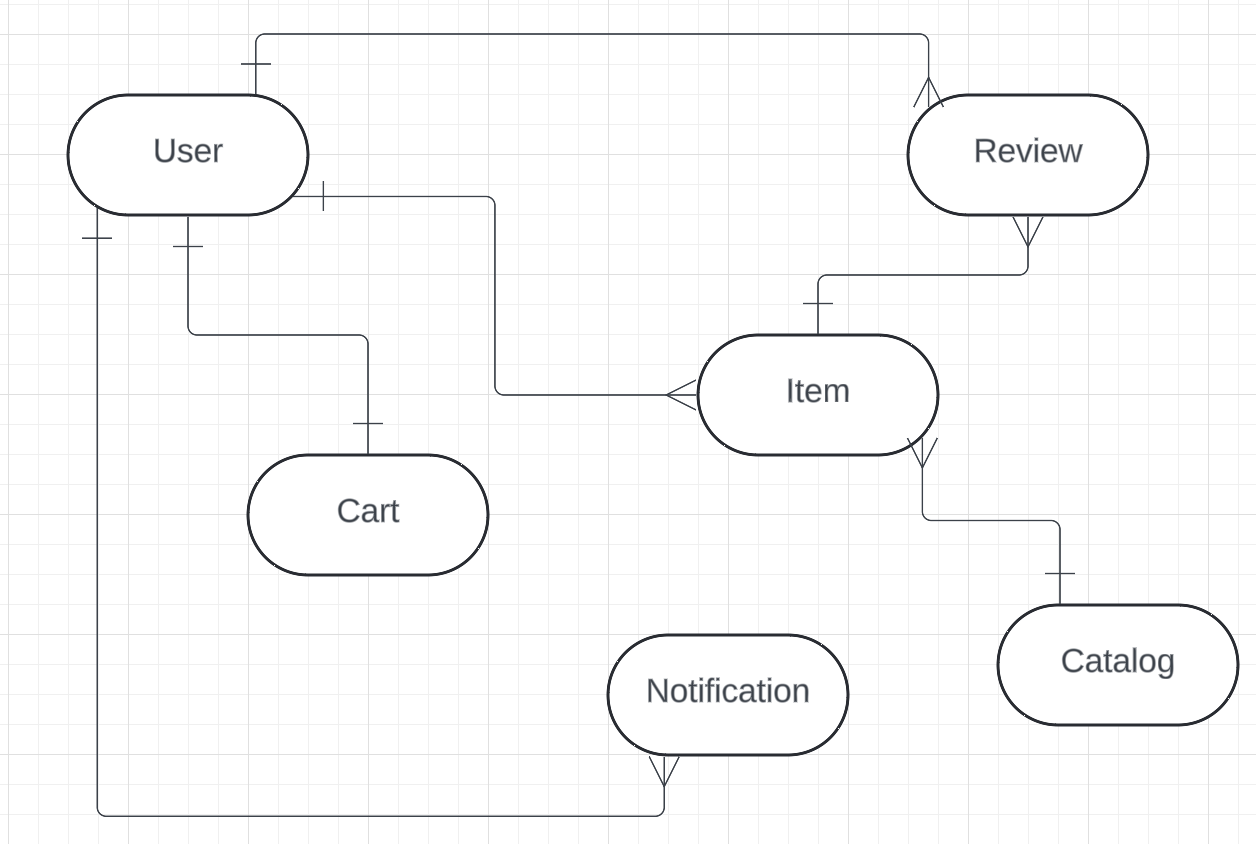
Success Scenario : The user registers with the help of the registration field. The system allows the user to choose whether to upload or book an item. Uploader adds information about the item along with photos. Booker marks needed item from the listed item with checkbox. If the marked item exists, the system opens the catalog, if not system will notify via notification. The user puts the desired item in the cart and contacts the uploader with the mentioned number or e-mail. The final Step that system allow users is reviewing and rating.

Identifying the objects :

* User Booker
* Registration field Checkbox
* System Catalog
* Item Notification
* Uploader Cart
* Information Number
* Photos E-mail

Identifying classes:

* User
* Item
* Cart
* Review
* Notification
* Catalog



**Relationships between Classes :**

1. User and Item: One-to-Many Relationship (one user can book/upload many items)
2. User and Cart: One-to-One Relationship (one user can have just only one cart)
3. User and Notification: One-to-Many Relationship (One user can receive many notifications)
4. User and Review: One-to-Many Relationship (one user can write many reviews)
5. Catalog and Item: One-to-Many Relationship (catalog contains many items)
6. Item and Review: One-to-Many Relationship (one item can have many reviews)