

eVidyaloka - Online Education Portal

To Educate - Anywhere - Anytime



Technical Report · January 2024 _ Developer – **Soniya Soni**

Use Case - **eVidyaloka**

A Non-Profit Organization wants to create a system to track books and AV equipment that Employees / Students / Volunteers can borrow from a library. Design and implement a system on the Salesforce platform to meet the requirements.

Description

The **eVidyaLoka** Library Management is simple, user-friendly, and can be easily integrated with your existing system. The library management system provides online and offline storage, automated backups, and easy upgrades to simplify and enhance the learning process.

- The library management system provides mobile access to search the library catalog, schedules, books and resources from anywhere, at any given time via Smartphones and tablets
- All records related to the book issued and return deadline can be managed by the authorized personnel. The system also allows to record the fine received from the reader in the same platform, which eases the work of reporting for the Library department.
- It eliminates paper-based processes and reduces operation costs and saves time of the library management. So, the library will become free of the number of registers and files.
- Maximize the performance of libraries with dynamic reports, charts and graphs to review and track the progress for better decision-making.

Objectives

The principle goal is to give a free environment with a proficient and intuitive interface to make the library administration framework less demanding and sorted out. Other goal are:

- Upgrading the present correspondence frameworks.
- Organization creates digital classrooms for children in remote Areas.
- Providing consistent learning experience
- Automating the whole procedure.
- Reducing Manual Error.

Implementation

- Creation of the required Data Model for above application
- Using LWC for creating user-friendly application which will show and implement all Use case of LWC

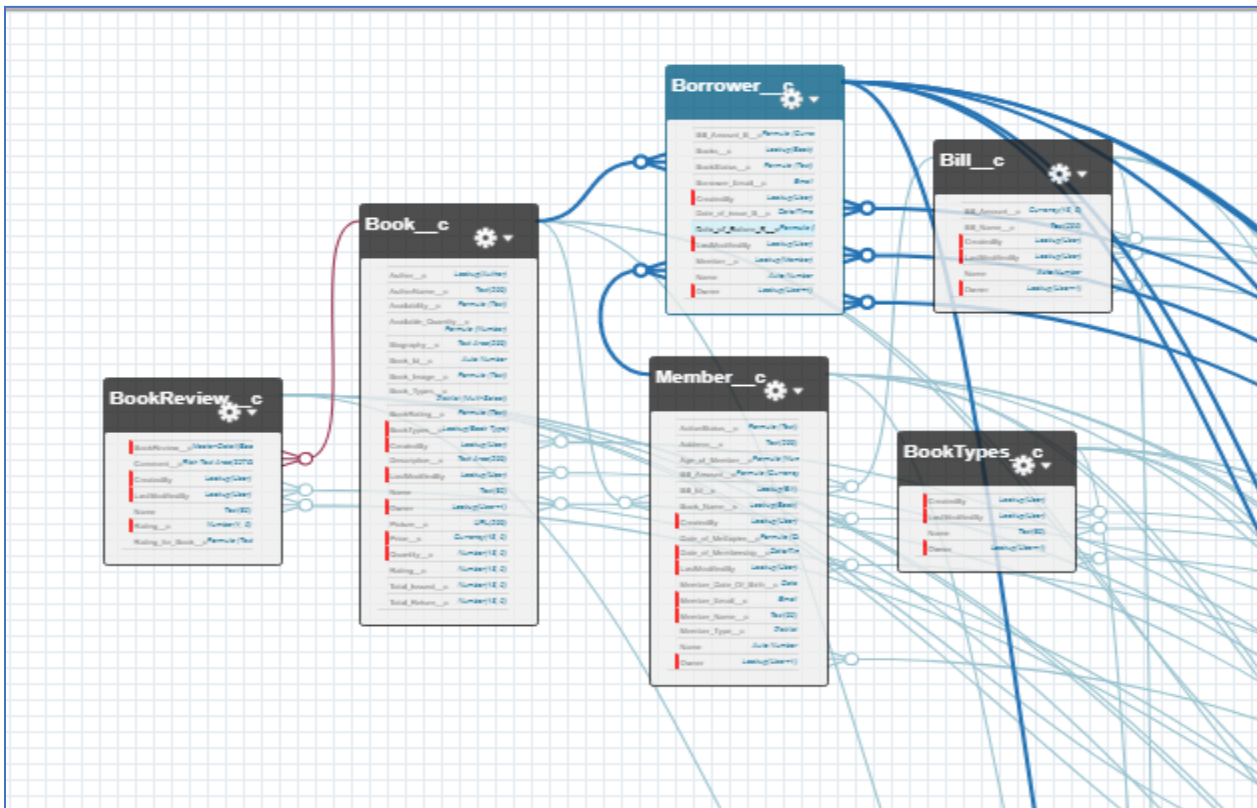
Custom Objects Required are-

- **Book** : Contain info about All Books
{ Book_Id ,Author Name , Biography ,Book Name, Description, Quantity ,Price, Available }
- **Borrower** : Contain info about All book Borrowers
{ borrower_name, borrower_id, Email}
- **Member** : Contain info about All Members associated with Library
{Name, Address, Member_id, Member_Type, Date of Membership, Date of Expire}
- **Bill** : Contain info about All Bills associated with Library and Members
{Bill_id, Member_id, Bill_Amount}
- **Book Review** : Contain info about Book Comments and ratings of books
{Book Review, Comment, Rating}
- **Book Types**: Contain info about Books and AVEquipments associated with Library
{Equipment Types , Book Name}

Field required are-

- All the details of the Member such as Name, Address, email, Member id.
- All the details of the book required.
- All the details if a bill is generated .
- All the details of the borrower are tracked.

Entity Diagram

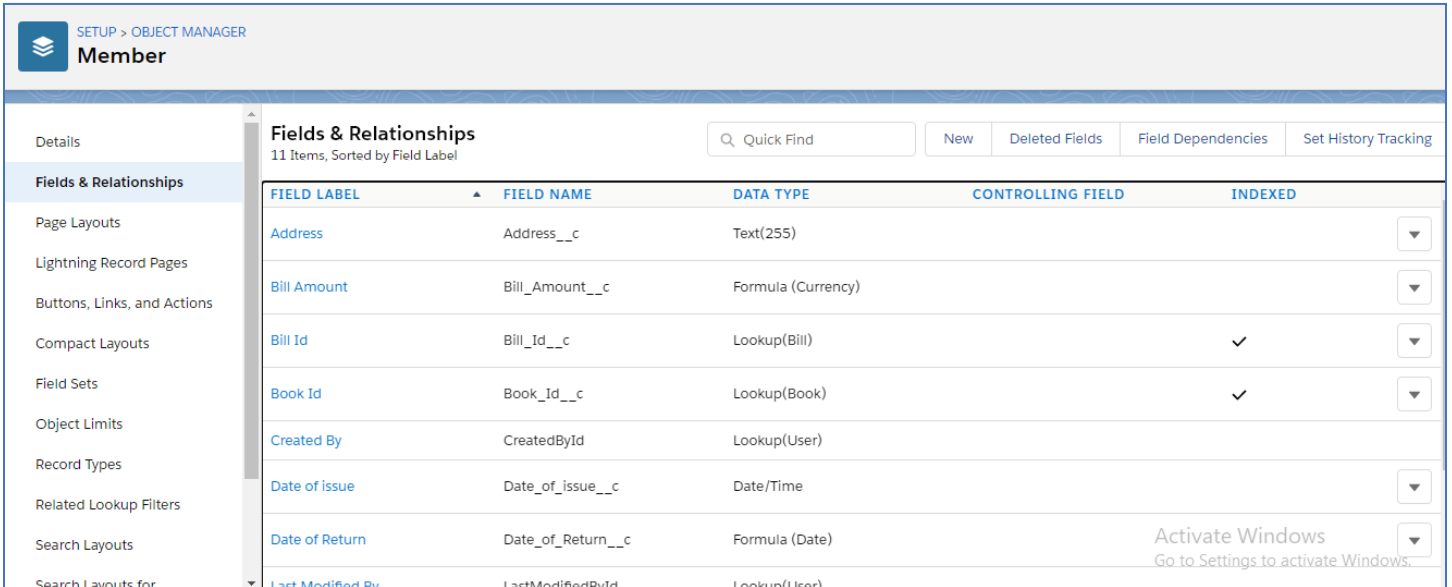


Book and BookReview:

This object has a master-detail relationship with **Book**. So one book can have many comments/ratings.

List of the fields under the objects

a) Member



FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(255)		
Bill Amount	Bill_Amount__c	Formula (Currency)		
Bill Id	Bill_Id__c	Lookup(Bill)		✓
Book Id	Book_Id__c	Lookup(Book)		✓
Created By	CreatedById	Lookup(User)		
Date of issue	Date_of_issue__c	Date/Time		
Date of Return	Date_of_Return__c	Formula (Date)		
Last Modified By	LastModifiedById	Lookup(User)		

Here the Address is Text Area data type, Member id is a Auto Number so that we can generate the number automatically. Member type is been pick list as we have to provide the dropdown for student, faculty and Admin

Formula Fields for Member Object

1) **Bill Amount** : we have put a formula to calculate the bill.

Formula Used: $IF(TODAY() > Date_of_Return_c, (TODAY() - Date_of_Return_c) * 2, 0)$
Date of issue and Date of Return are date and formula (date) data type respectively.

2) **Date of Membership Expire** : we have put a formula to calculate the Date of Return.

Formula Used: $Date_of_MeExpire_c + 360$ (membership expire after 1 year)

3) **Age Of Member** : Calculated by a formula filed from Date of Birth of member

Formula Used: $YEAR(TODAY()) - YEAR(Memeber_Date_Of_Birth_c)$

4) **Member Active Status:** $IF((Date_of_MeExpire_c - Today()) < 0,$
 $IMAGE("/img/samples/color_red.gif", "red", 30, 30),$
 $IF((Date_of_MeExpire_c - Today()) < 10,$
 $IMAGE("/img/samples/color_yellow.gif", "yellow", 30, 30),$
 $IMAGE("/img/samples/color_green.gif", "green", 30, 30)$
 $))$

Validation Rules Used for Member Object

1) Date Of Membership should not be Saturday or Sunday

Member Validation Rule

[Back to Member](#)

Validation Rule Detail Edit Clone

Rule Name	Date_Of_Membership	Active
Error Condition Formula	CASE(MOD(DATEVALUE(Date_of_Membership__c) - DATE(1900, 1, 7), 7), 0, 0, 6, 0, 1)= 0	
Error Message	Date Of Membership should not be Saturday or Sunday	Error Location
Description	Date_Of_Membership should not be Saturday or Sunday	
Created By	<u>SONIA SONI</u> , 1/6/2021, 3:12 AM	Modified By

Edit Clone

2)Email field should contain valid Email Address

Member Validation Rule

[Back to Member](#)

Validation Rule Detail Edit Clone

Rule Name	Email_Rule
Error Condition Formula	NOT(REGEX(Member_Email__c, "[a-zA-Z0-9._-]+@[a-zA-Z]+.[a-zA-Z]{2,4}"))
Error Message	Please Enter a valid email Address
Description	Check if Email entered is in correct format
Created By	<u>SONIA SONI</u> , 1/6/2021, 10:05 PM

Edit Clone

5) Member Age should be between 16- 65 years

Member Validation Rule

[Back to Member](#)

Validation Rule Detail Edit Clone

Rule Name	Age_of_Member
Error Condition Formula	OR(Age_of_Member__c > 15, Age_of_Member__c < 65)
Error Message	Age of Member should lie between 16 -65
Description	Age of Member should be between 16-65
Created By	<u>SONIA SONI</u> , 1/10/2021, 11:20 PM

Edit Clone

In this project we have used data type Lookup to build the relationship between the two objects Bill and member, and similarly with the Books and Member.

b) Book

List of the fields under the object Book

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Author	Author__c	Lookup(Author)		✓
Author Name	Author_Name__c	Text(255)		
Availability	Availability__c	Formula (Text)		
Biography	Biography__c	Text Area(255)		
Book Name	Name	Text(80)		✓
Created By	CreatedById	Lookup(User)		
Description	Description__c	Text Area(255)		

Here the field Author and Title have Text area (255) Data Type. Book Id is Auto Number data type as we need to generate the automatic number for the books. Similarly we used data type as Currency (18,0) and Number(3,0) for Price and Quantity respectively.

Formula Fields

1) **Availability** : We have used a formula to check the availability for the books in the system.

Formula used: IF (Quantity__c > 0,"Available","Not Available")

- 2) **Book Image** : We have used a formula to get Image
Formula used: IMAGE(Picture__c , Name)

Page layout for the objects and fields for the object Book

The screenshot shows a web interface for a library management system. At the top, there is a navigation bar with 'Library Managememe...' and several menu items: 'Books', 'Recommendations', 'Members', 'Bills', 'Reports', and 'Dashboards'. A search bar is also present. Below the navigation, the main content area displays the details for a book titled 'Flashy Superhero'. The details are organized into two columns:

Book Name Flashy Superhero	Owner SONIA SONI
Description This exciting story is all about being fast. Really fast.	Author Name Iris West
Genre Humor	Quantity 10
Biography Iris West is a fictional character, a supporting character appearing in American comic books published by DC Comics	Availability Available
Price \$25	
Created By SONIA SONI, 1/4/2021, 11:15 PM	Last Modified By SONIA SONI, 1/5/2021, 10:56 PM

Validation Rules on Book Object

- 1) **Book Name** field should be only contain Alphabets

The screenshot shows a configuration page for a validation rule. The title is 'Book Validation Rule' with a 'Back to Book' link. Below the title, there is a 'Validation Rule Detail' section with 'Edit' and 'Clone' buttons. The details are as follows:

Rule Name	BookName
Error Condition Formula	NOT(REGEX(Name, "[a-z A-Z]*\$"))
Error Message	Book Name should contain only alphabets
Description	Book Name should contain only alphabets
Created By	SONIA SONI, 1/9/2021, 9:03 PM

At the bottom of the detail section, there are 'Edit' and 'Clone' buttons.

2) Quantity of Books should be greater than Issued Books

Book Validation Rule
[Back to Book](#)

Validation Rule Detail Edit Clone

Rule Name	Book_1
Error Condition Formula	Quantity__c + Total_Issued__c + Total_Return__c > Quantity__c
Error Message	Book Issue cannot be greater than Total Quantity of Books
Description	
Created By	SONIA SONI, 1/6/2021, 2:59 AM

Edit Clone

Trigger on Book Object

This Trigger will fire for Before and After Event on Book Object for following conditions.

- 1) When Quantity = 0 : Since Book Quantity is zero cannot issue book at this time.
- 2) When Total Issue is > then Quantity : Book Issue cannot be greater than Quantity of Books
- 3) When Total Return > then Issued : Book Return cannot be greater than total Books Issued



c) **Bill**
 List of the fields under the object Bill

SETUP > OBJECT MANAGER
Bill

Details

Fields & Relationships 6 Items, Sorted by Field Label Q Quick Find New Deleted Fields Field Dependencies Set

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Bill Amount	Bill_Amount__c	Currency(18, 0)		
Bill ID	Name	Auto Number		✓
Bill Name	Bill_Name__c	Text(255)		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓

Page Layouts
 Lightning Record Pages
 Buttons, Links, and Actions
 Compact Layouts
 Field Sets
 Object Limits
 Record Types
 Related Lookup Filters

d) **Borrower**: Shows Relation between Book and Member Object

Formula Fields for Borrower Object

1) **Bill Amount** : we have put an formula to calculate the bill.

Formula Used: IF(TODAY(> Date_of_Return_B__c ,(TODAY()- Date_of_Return_B__c)*2, 0)
Date of issue and Date of Return are date and formula (date) data type respectively.

2) **Cross Object Formula field** :

Formula Used: IF (Books__r.Available_Quantity__c > 0,"Borrow Now","Not Available")
This will not allow User to Borrow Books when Quantity of Book is Zero

Borrower Validation Rules

1) **Date of Issue cannot be Saturday or sunday**

Borrower Validation Rule

[Back to Borrower](#)

Validation Rule Detail

Edit Clone

Rule Name	Date_Of_Issue
Error Condition Formula	CASE(MOD(DATEVALUE(Date_of_issue_B__c) - DATE(1900, 1, 7), 7), 0, 0, 6, 0, 1)= 0
Error Message	Date Of Issue should not be Saturday or Sunday
Description	Date_Of_Issue should not be Saturday or Sunday
Created By	<u>SONIA SONI</u> , 1/10/2021, 10:43 PM

Edit Clone

2) **Data of Return should be weekdays**

Borrower Validation Rule

[Back to Borrower](#)

Validation Rule Detail

Edit Clone

Rule Name	Date_Of_Return
Error Condition Formula	CASE(MOD(Date_of_Return_B__c - DATE(1900, 1, 7), 7), 0, 0, 6, 0, 1) = 0
Error Message	Date must be a weekday.
Description	Date of Return Cannot be Saturday or Sunday
Created By	<u>SONIA SONI</u> , 1/10/2021, 10:46 PM

Edit Clone

3) Trigger Validation Rule on Borrower Object

```
if(Trigger.isBefore && bolssue.BookStatus__c == 'Not Available') {  
    bolssue.addError('Since Book Quantity is zero cannot issue book at this time')
```

e) BookReview Object Fileds

BookReview__c : **Master Detail Relationship** Between Book and BookReview
Coz BookReview cannot be created until book is there.

Custom Tabs: We have provide different color tabs for our objects as shown in the below snapshot.

Declarative Tools Used

Process builder

Sample Email used in the project: We have used the following two sample Email templates;

Unfiled Public Classic Email Templates [Help for this Page](#)

Classic Email Template Availability [\[Expand \]](#)

Folder: Unfiled Public Classic Email Templates [Create New Folder](#)

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other

Action	Email Template Name ↑	Template Type	Available For Use	Description	Author	Last Modified Date
Edit Del	Books are Due for Re-Order	Text	✓	Books are Due for Re-Order when Quantity is zero	SSONI	1/5/2021
Edit Del	Marketing: Product Inquiry Response	Text	✓	Standard email response to website product inquiries	SSONI	1/4/2021
Edit Del	Reminder for Re-Issue of books	Text	✓	Reminder for Re-Issue of Books	SSONI	1/6/2021
Edit Del	Sales: New Customer Email	Text	✓	Email to new customers	SSONI	1/4/2021
Edit Del	SUPPORT: Self-Service New Comment	Text	✓	Sample email template that can be sent to your Self-Service customers to notify	SSONI	1/4/2021

a) **Books overdue for Re-Order :** This email will be generated and delivered to the members if their books quantity is zero and need to Re-Order.

Email Template [Send Test and Verify Merge Fields](#)

Subject | Books- {!Book__c.Name} -are Due for Re-Order as Quantity is zero


Plain Text Preview

Hi {!Member__c.Name} {!Member__c.Member_Type__c}

This is to inform you that following Books {!Book__c.Name}{!Book__c.Book_Id__c} need to be Re-Order as these are more in Demand !!

Email Alert received by above action:

Books- Watership Down -are Due for Re-Order as Quantity is zero Inbox x

 **SONIA SONI** via z29syhq31jz26i.5g-2vfbaea2.ap24.bnc.salesforce.com 1:49 PM ()
to me, sworldsam@gmail.com ▾

Hi

This is to inform you that following Books Watership Down12 need to be Re-Order as these are more in Demand !!

b) Reminder Email for Re-issue for the book: This email will be generated and send to the borrowers if they have the books pending and the books are needed to be Re-issued otherwise Charges will be applied as penalty.

Email Template Send Test and Verify Merge Fields

Subject | Reminder for Re-Issue of Books

Plain Text Preview


Hi `{!Borrower__c.Name}` `{!Borrower__c.MemberId__c}` `{!Borrower__c.Member__c}`

It's a Gentle Reminder that your Books are due for Re-Issue , Kindly Ressue it or return it before Date of Return `{!Borrower__c.Date_of_Return_B__c}`.
Or else you will be charged 50 Rupee per day as a penalty .

Happy Learning

Thanks You.

Reminder for Re-Issue of Books

 **SONIA SONI** via 8jcmktdbj20qs1.5g-2vfbaea2.ap24.bnc.salesforce.com 2:05 P ()
to me, sworldsam@gmail.com ▾

Hi Fana
Member Type [Student] ,

It's a Gentle Reminder that your Books are due for Re-Issue , Kindly Ressue it or return it before Date of Return 1/10/2021.
Or else you will be charged 50 Rupee per day as a penalty .

Happy Learning

Thanks You.

c) **Reminder Email for Renew of Membership** : This email will be generated and send to the members regarding Renew of membership otherwise Charges will be applied.

Email Template Send Test and Verify Merge Fields

Subject | Reminder Email for Renewals of membership

Plain Text Preview

Hi {!Member__c.Member_Name__c} {!Member__c.Member_Type__c}

It's a Gentle Reminder that your Membership is due for Renew , Kindly Renew it before {!Member__c.Date_of_MeExpire__c}
Or else your Membership will get Expired and charges will be Applicable.

Happy Learning

Thanks You.

Email Alert received by above action:

SONIA SONI via 4beyzvm33oqqld.5g-2vfbaea2.ap24.bnc.salesforce.com
to soniasalesforce9@gmail.com, me ▾

Hi Seema Faculty

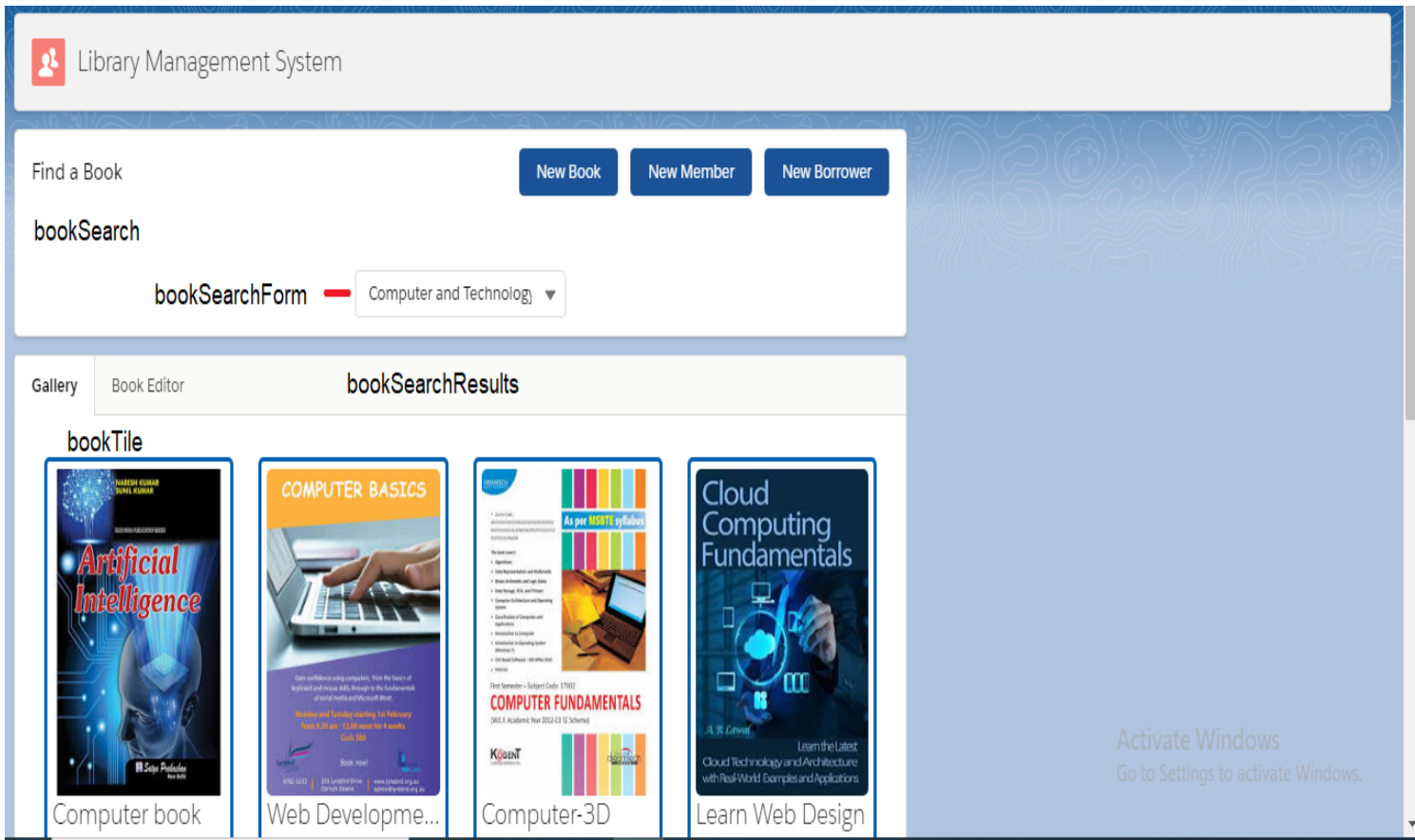
It's a Gentle Reminder that your Membership is due for Renew , Kindly Renew it before 1/7/2021
Or else your Membership will get Expired and charges will be Applicable.

Happy Learning

Thanks You.

Features:

1. Use of LWC in Application



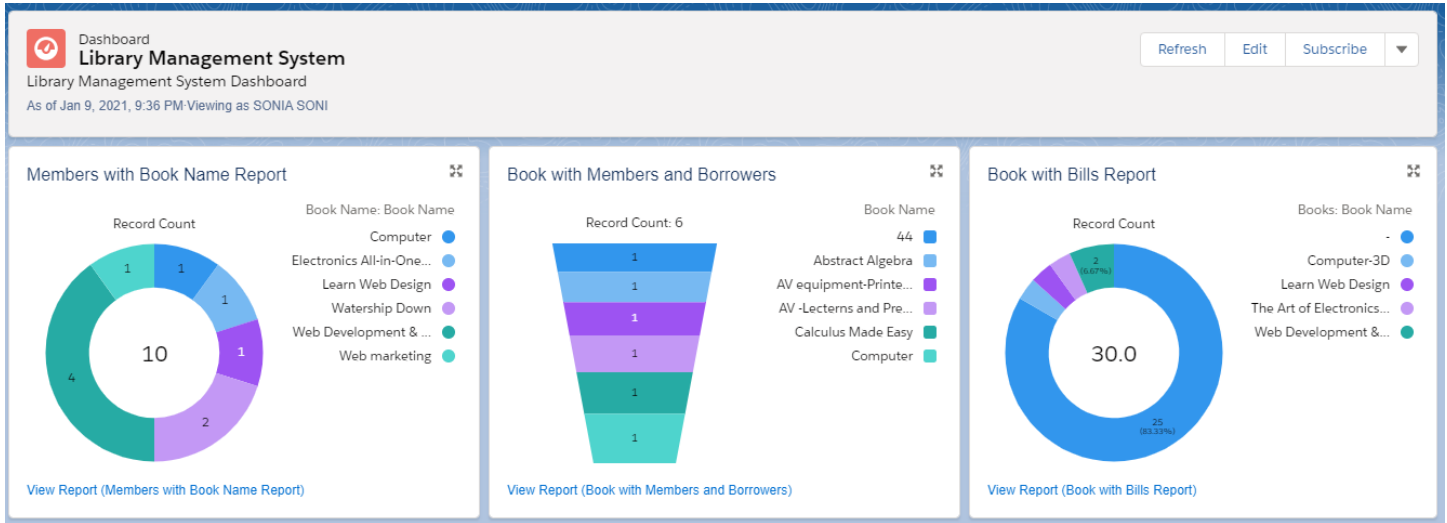
2. Report: To create the report we have to follow the process click on New Report-> select report type -> Then chose the desire report type. I have created a few Report types for this project which are as mentioned below.

1) Book with Members: It draws the graph between the books with the members and displays that the number of books that each member have.

2) Books with Bills : Here in the report the maximum numbers of books issued to Members

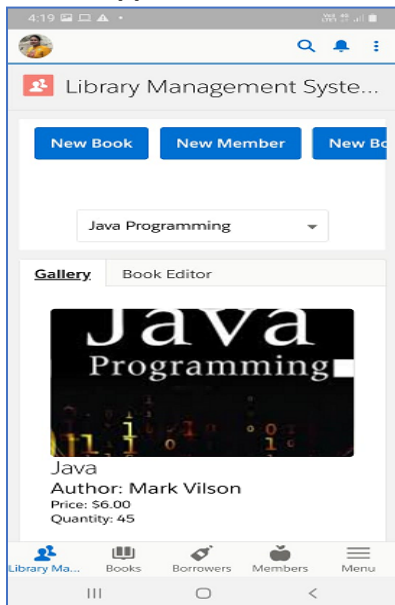
3) Books versus Member and Bill: This plot shows the comparison between Books issued by the Members and the bill generated by each member on the books.

3. Dashboard: At last goes the dashboard the place the explanatory Also business choices would be produced. It provides for you an acceptable slice dream for what may be going on around the firm. You might screen each Furthermore each single part for your firm. This Dashboard comprises all the three reports that are explained earlier and it will give a live track of the data that has been flowing through the system. All the three reports have been collided and presented in this form of the dashboard.



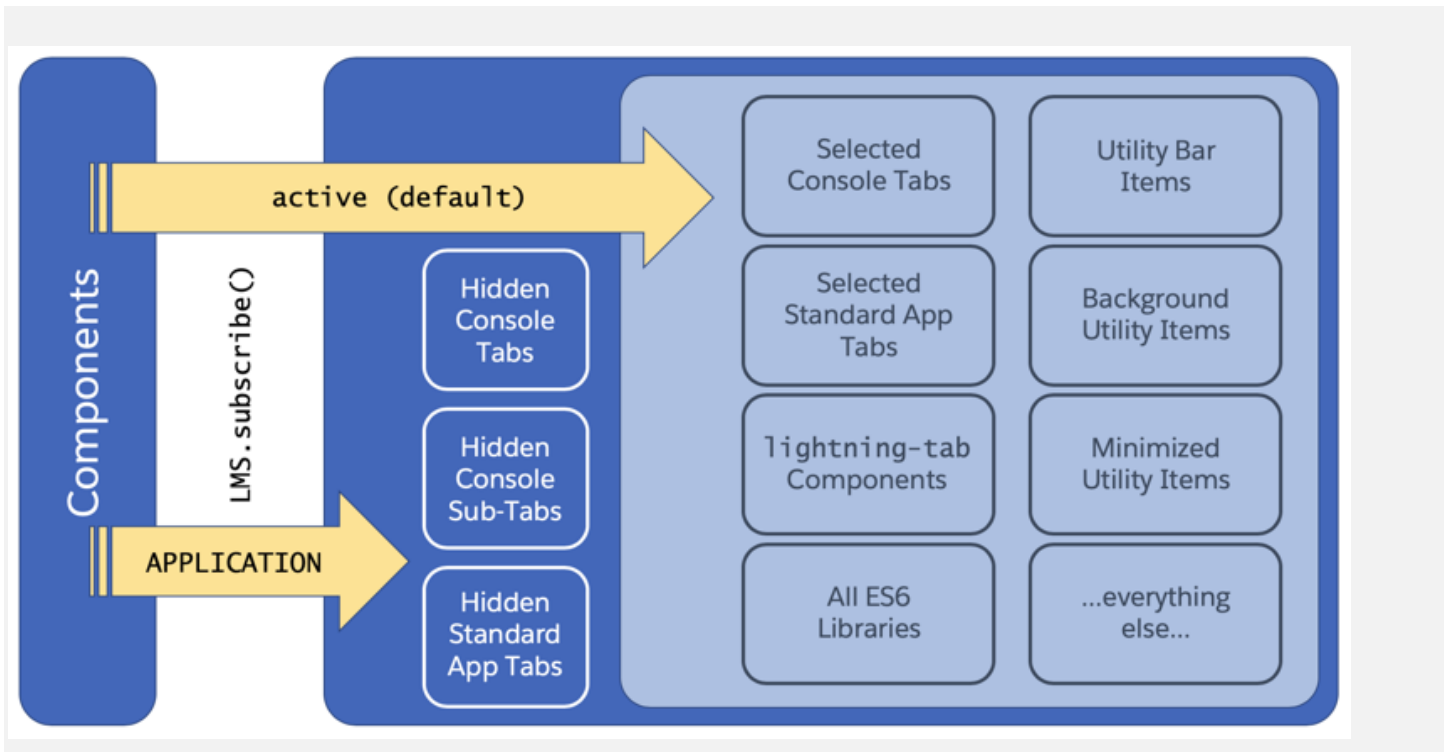
4. Mobile App-LMS - Access to library catalog, schedules, books, Dashboard, Reports and resources from anywhere, at any given time via Smartphones and tablets gives user freedom to quick Learning from Anywhere and Anytime.

Mobile App-LMS



5. Use of Lightning message service

The Lightning message service now lets you limit scope to the active tab only (default) or include the entire application.



To specify active scope, you don't need to include the scope parameter, since it's the default behavior. To specify scope for the entire application, continue to use the scope parameter with a value of APPLICATION.

Key Things learned and implemented

- Building reusable LWC components inside Lightning App Builder
- Lightning Message Service (LMS)
- Lightning Data Service (LDS)
- Communication between components (Parent to Child and Child to Parent)
- Apex Classes and Triggers
- Lightning Datatable, Layout, Tabset, Spinner, custom Labels
- Utilizing External Javascript
- Navigation between pages
- *Mild* Lightning Design System (SLDS)
- Automation Flows , process builder

