LIBRARY MANAGEMENT SYSTEM

A PROJECT REPORT Submitted
partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

by

Deepali Jain
(University Roll No : 1900290149038)

Under the Supervision of
Ms. Neelam Rawat
Associate Professor

to the

Department of Computer Applications

Dr. A.P.J ABDUL KALAM TECHNICAL UNIVERSITY
LUCKNOW

JULY, 2021

SaaSnic Technologies

Connecting Clouds

Date: 09th Aug 2021

TO WHOM IT MAY CONCERN

This is to certify that **Ms. Deepali Jain** D/O **Manoj Jain,** a student of MCA, KIET Group of Institutions, Ghaziabad has successfully completed **five month's training** period at **SaaSnic Technologies Services Pvt Ltd, Noida**. She joined the company on **15th March 2021.**

We wish you all the best for your future.

Himani Adhikari
(Sr. HR)

Authorized Signature

Office Address : Block G-284, 2nd Floor, Sector 63, Noida, UP-201301
Tel.: 0120-4260370/ 71 | Email : info@saasnic.com | website : www.saasnic.com

CERTIFICATE

Certified that **Deepali Jain** (**1900290149038**) has carried out the project work presented in this project report entitled "**Library Management System**" for the award of Master of Computer Application from Dr. A.P.J. Abdul Kalam Technical University, Lucknow under my supervision. The report embodies result of original work, and studies are carried out by the student himself and the contents of the report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University.

Ms. Neelam Rawat **External Examiner**

Associate Professor

Dept. of Computer Applications

KIET Group of Institutions, Ghaziabad

Dr. Ajay Kumar Srivastava

Professor & Head

Department of Computer Applications

KIET Group of Institutions, Ghaziabad

Date:

LIBRARY MANAGEMENT SYSTEM

DEEPALI JAIN

ABSTRACT

Library management is a sub-discipline of institutional management that focuses on specific issues faced by libraries and library management professionals. Library management encompasses normal managerial tasks, as well as intellectual freedom and fundraising responsibilities. Issues faced in library management frequently overlap with those faced in managing non-profit organizations.

Library management system is a project which aims in developing a computerized system to maintain all the daily work of library. This project has many features which are generally not available in normal **library management systems**

Library Management System is designed & developed for a receipt and issuance of books in the **library** along with the student's details. The books received in the **library** are entered in Books Entry form and the new student is entered in the student entry form.

ACKNOWLEDGEMENT

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, **Ms. Neelam Rawat** for her guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Ajay Kumar Srivastava, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

DEEPALI JAIN

1900290149038

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. Project Description

One of the universities which is growing really fast has many students and faculty members. To attract doctorate, post doctorate students to roll into the university, management decided to buy periodicals, magazines, books and research related papers and expand library facilities. The current library management system has become obsolete and is not able to provide the necessary data to manage it effectively. It is becoming difficult for students to block magazines in advance, take more than 2 or 3 books, not able to search either through the title. Hence the management decided to implement a new library management system using SalesForce.com using APEX platform to have an interactive, scalable, flexible, reliable system.

- The main aim of this project is to design the application using Cloud computing applications branch out well beyond CRM. Salesforce.com customers are building more apps, and building better apps.

- The objective of the Library Management System is to handle the entire activity of a library.

- The software keeps track of all the information about the books and their complete details.

- Data are store on cloud where all the information will be stored safely and access anywhere, anytime by authorize person.

1.2. Project Scope

To attract doctorate, post doctorate students to roll into the university, management decided to buy periodicals, magazines, books and research related papers and expand library facilities. The current library management system has become obsolete and is not able to provide the necessary data to manage it effectively. It is becoming difficult for students to block magazines in advance, take more than 2 or 3 books, not able to search either through the title.

This project will keep track of all the books and library information. The software will be able to handle all the necessary information.

1.3. Hardware / Software used in the Project

Table 1.1 Hardware

Hardware	**Configuration**
Processor	Intel Pentium G2030 clocked at 3.0 GHz
RAM	4GB DDR4
Monitor	Dell Backlit 21” LED
Modem	Internet Connectivity
Keyboard	Dell Standard 102 Keys & Optical Mouse

Table 1.2 Software

Software	**Configuration**
Operating System	Windows XP /7/8/10
Software	Chrome, Microsoft Edge

1.3.1. Technology Description

SALESFORCE

Salesforce.com, Inc. is an American cloud-based software company headquartered in San Francisco, California. It provides customer-relationship management (CRM) service and also sells a complementary suite of enterprise applications focused on customer service, marketing automation, analytics, and application development.

Salesforce is the primary enterprise offering within the Salesforce platform. It provides companies with an interface for case management and task management, and a system for automatically routing and escalating important events. The Salesforce customer portal provides customers the ability to track their own cases, includes a social networking plug-in that enables the user to join the conversation about their company on social networking Web sites, provides analytical tools and other services including email alert, Google search, and access to customers' entitlement and contracts

Service Nature of Salesforce:

1. **Salesforce as SaaS (Software as a Service):** There is no need for installation, setup or downloads but you just have to log in and start using the software apps across the cloud. Isn't it more convenient and amazing? The answer is a big "YES" when using Salesforce CRM for your business.
2. **Salesforce as PaaS (Platform as a service):** Here, there is no need for a separate platform, but you can use code created by other developers to deploy apps. Obviously, you have to customize it as per your business needs, but it saves a lot of time and money as demanded by businesses today.
3. **Salesforce as IaaS (Infrastructure as a service):** Here, there is no need for installing any hardware or software program, but data and apps are stored securely on the cloud. Even you don't have to take the backup, but the cloud will take care of it automatically.

1.3.1.1 APEX

Apex is a proprietary programming language provided by the Force.com platform to developers similar to Java and C#. It is a strongly typed, object-oriented, case-insensitive programming language, following a dot-notation and curly-brackets syntax. Apex can be used to execute programmed functions during most processes on the

Force.com platform including custom buttons and links, event handlers on record insertion, update, or deletion, via scheduling, or via the custom controllers of Visualforce pages.

Due to the multitenant nature of the platform, the language has strictly imposed governor limitations to guard against any code monopolizing shared resources. Salesforce provides a series of asynchronous processing methods for Apex to allow developers to produce longer-running and more complex Apex code.

1.3.1.2 LIGHTNING

Salesforce made public the front end of its platform, called Lightning. This component-based framework is what the Salesforce mobile app is built on. Salesforce built on this framework in 2015 by releasing the Lightning Design System, an HTML style framework with default CSS styling built in. This framework allows customers to build their own components to either use in their internal instances or sell on the AppExchange.

The Salesforce Lightning App Builder is a tool for rapid application development of responsive web interfaces. This interface allows for different screens to be put together based on Lightning components. This can be used as layouts for records or specific applications.

Lightning Experience, released in 2016, is the new redesigned interface in Salesforce for processes enhancement. Since then all the apps available on AppExchange need to be Lightning and those built on Classic have to migrate to Lightning as Classic is not to be updated any more by Salesforce. The platform offers an option for developers to employ migration techniques to enable the new user interface and switch to Lightning.

1.3.1.3 JavaScript

JavaScript (often shortened to **JS**) is a lightweight, interpreted, object-oriented language with first-class functions, and is best known as the scripting language for Web pages, but it's used in many non-browser environments as well. It is a prototype-based, multi-paradigm scripting language that is dynamic, and supports object-oriented, imperative, and functional programming styles.

JavaScript runs on the client side of the web, which can be used to design / program how the web pages behave on the occurrence of an event. JavaScript is an easy to learn and also powerful scripting language, widely used for controlling web page behavior.

Contrary to popular misconception, **JavaScript is *not* "Interpreted Java"**. In a nutshell, JavaScript is a dynamic scripting language supporting prototype based object construction. The basic syntax is intentionally similar to both Java and C++ to reduce the number of new concepts required to learn the language. Language constructs, such as if statements, for and while loops, and switch and try ... catch blocks function the same as in these languages (or nearly so).

JavaScript can function as both a procedural and an object oriented language. Objects are created programmatically in JavaScript, by attaching methods and properties to otherwise empty objects **at run time**, as opposed to the syntactic class definitions common in compiled languages like C++ and Java. Once an object has been constructed it can be used as a blueprint (or prototype) for creating similar objects.

JavaScript's dynamic capabilities include runtime object construction, variable parameter lists, function variables, dynamic script creation (via eval), object introspection (via for ... in), and source code recovery (JavaScript programs can decompile function bodies back into their source text).

1.3.1.4. SOQL

SOQL is Salesforce Object Query Language that is highly similar to SQL (Structured Query Language). With the help of SOQL, you can always search the organizational data wisely. It can be combined with APEX, Visualforce, or Force.com

IDE too. While writing a query using SOQL, you should use SELECT command and a lot of other conditions too.

The SOQL can be used when you have the basic idea of objects and related data. It can be used to retrieve data from one or multiple objects that are connected together based on requirements. It helps in getting the total count and sort queries as well. Further, it can be combined with SOSL (Salesforce Object Search Language) to search the organizational data when they are not sure about objects.

SOSL is programmed in such a way to work on the basis of the search index. To improve the performance of searching, you should combine SOQL and SOSL together because searching work is done better by SOSL as compared to the SOQL.

CHAPTER 2

LITRATURE REVIEW

Salesforce is one of the main assets of any organisation. As salesforce is controlled by a control system, it is important to know what kind of impact a salesforce control system has on the salesforce. This paper contributes to the salesforce control literature by providing a comprehensive review based on research published in the area of salesforce control system in the last 43 years (1975-2018). [1]

Salesforce performance is extremely important for the organisation as it has a direct impact on the effectiveness of that organisation. A number of factors have an impact on salesforce performance, such as the salesforce characteristics, sales territory design and salesforce control systems. [2]

Sales literature suggests that salesforce control systems are likely to have a strong impact on sales territory design. Therefore, this study makes an attempt to analyse the impact of two types of salesforce control system, behaviour-based and outcome-based salesforce control systems, on sales territory design and also the impact of sales territory design on salesforce performance and sales organisation effectiveness. [3]

The paper is based on a review of studies published between 1987 and 2018, and reveals that a behaviour-based control system is more positively related to satisfaction with sales territory design than an outcome-based salesforce control system. [4]

The objectives of this review are to examine the chronological trends in the area of salesforce control system research in terms of research focus, research methods, research type, geographical region, etc. Most of the studies included in our review have concluded that the salesforce control systems (behaviour-based and outcome-based) have different impact on salesforce characteristics, sales territory design, salesforce performance and sales organisation effectiveness. The study has important implications for sales organisations. [5]

Information technology, comprises both hardware and software. Hardware includes, but is not limited to, desktop, laptop and handheld devices. SFA software applications offer a range of functionality. [6]

Customer Relationship Management (CRM) has become one of the most dynamic technology topics of the millennium. CRM is not a concept that is really new but rather due to current development and advances in information and enterprise software technology, it has assumed practical importance. The root of CRM is relationship marketing, which has the objective of improving the long-term profitability of customers by moving away from product-centric marketing. [7]

CRM was invented because the customers differ in their preferences and purchasing habits. If all customers were alike, there will be link need for CRM. As a result. understanding customer drivers and customer profitability. firms can better tailor their offerings to maximize the overall value of their customer portfolio. [8]

CRM generally is an enterprise-focused endeavour encompassing all departments in a business. He further explains that, in addition to customer service. CRM would also include, manufacturing, product testing, assembling as well as purchasing, and billing, and human resource, marketing, sales and engineering. [9]

There are various ways through which a customer can approach the business. This interaction is direct with company and its employees. The junction where this interaction happens is called touch point. Usually transactions like sale, payment. information seeking, queries, suggestions. and complaints happen at these operational touch points. That is why it is also called front office CRM. [10]

Salesforce is a cloud based system it does not need any software installation on your part. All you have to do is signup for a free trial and get started. The free trial account provides nearly all features which you need to learn to understand the basics of Salesforce platform. Let us now discuss the steps to get slimed with the Salesforce environment. [11]

Sales Cloud part of the Salesforce.com platform which is focused on enhancing the effectiveness oldie sales team of an organization and hence increases the amount of sales. h stands unique when compared to other sales methods as it provides both the account information of the customer as well as the information gathered from the social platforms about the product and customer. This helps in judging the potential of a sales lead and closing the sales faster. [12]

Salesforce is by far the largest and most widely-known CRM. This impassive level of success is well-deserved due to the many incredible advantages and extensive resources that Salesforce provides for its users. There are a great many advantages to using this CRM. and different companies may find some more useful and important than others. However, this article will briefly look at the top 5 advantages that have the widest range of applicability and effectiveness for the largest number of users. [13]

Salesforce has broad applicability for a wide range of different companies and businesses. The key to this successful approach is in its flexibility and customization potential. Salesforce integrates well with a host of different business models because it is able to provide report and analytics that are tailored to the specific needs of its users.[14]

The main purpose of this review paper is to establish the basic concepts of SALESFORCE. Salesforce is easy to learn and growing technology in the marketing and every data is stored into the cloud. So you can access anywhere and any device. Also for learning this technology you don't need to have the coding knowledge. [15]

CHAPTER 3

REQUIREMENTS

Considerations for the Library Management System:

Some of the requirements are:

1. To have list of different types of books and have the capability to search either through title or author or book name.
2. Track all the pending requests for books and get the status of the book.
3. Take the requests from students and faculty members and prioritize in buying books.
4. Give proper security accesses to Librarian so that pending requests are not edited and deleted by everyone.
5. Inform library member when the book requested arrives.
6. Include Reports to generate defaulters list, money collected every month.

2.1 Create the following Custom Objects

2.1.1 Books – This object holds all the information related to a book.

Data Type	Field Label	Other Values	Remarks
Text	Book_ID		Generated – Format B-{00000}
Date	Date of arrival		Date on which Library got it – Mandatory – Can't be greater than today.
Text	Author	Length:50	Mandatory
Picklist	Type	Magazines Periodicals Books WhitePapers	
Name	Text		Name of the book (Mandatory)
Currency or Number	Price of the book		In Dollar terms (Mandatory) and make sure that it is greater than Zero
Text Area(Long)	Description	32,000	Visible 3 lines

Fig. 2.1.1

2.1.2 Members – This object holds all the information related to a library Member.

Data Type	Field Label	Other Values	Remarks
Text	First Name	Length:50	Member's First Name – Mandatory
Text	Last Name	Length:50	Member's Last Name – Mandatory
Dependent picklist	Member Type	Student or faculty	Mandatory
Dependent picklist	Category		Beginner, Sophomore, Junior, Senior for students

			Computer Science, Mechanical, Electronics for faculty
Date	DOB		
Number	Borrowed_Books		Total number of books borrowed
Text	Email_Address		Should take the email address
Text	Member_ ID		Generated – Format M-{00000}

Fig. 2.1.2

2.1.3 Borrowed Books: This has one to many relationships between Library Member and books. Contains the data of books borrowed by the library member.

Data Type	Field Label	Other Values	Remarks
Text	Book_ID		Foreign Key from the BOOKS table (Books_ID).
Text	Member_ID		Foreign Key from the MEMBERs table (Member_ID)
Date	Borrowed Date		Borrowed Date can't be changed once the record is created
Date	Return Date		Borrowed Date + 15. Once borrowed date is created, can't be changed
Date	Actual Returned Date		Actual Return Date can't be less than Borrowed Date

Fig. 2.1.3

1.On Books, Members screen – on the related Borrowed Books screen need to show Member who borrowed the book (Member First Name) and also the BOOKID and the Author of the book.

2. On clicking BOOKS tab, please show BOOKID, Author, Book Name. On borrowing book screen, show which book one is borrowing like book name and the author, who is borrowing. Two look up fields to be created on that screen.

3. On clicking Members tab, please show Member First Name, Last Name.

4. Both are required fields.

5. Once the record is saved, need to show the author of the book on the saved screen.

6. In the look up screen show both BOOK_ID, AUTHOR and DESCRIPTION of the book.

7. Increment Member.Borrowed_Books for that member by 1 whenever Borrowed Book new record is created.

8. Member can't borrow more than 4 books.

9. Arrange the fields in two column fashion from top down in the same sequence.

10.Whenever Book is returned – the borrowed_books counter in Member table to be reduced by 1.

11. Book date arrived can't be changed once the record is created.

12.Try to create a time trigger which should create a task or send an email whenever a record is created and whose due date in case of task should be Borrowed Date + 10.

13.Book Author name can't be changed once the book is borrowed.

14. Combination of MemberID and BOOKID should be unique in Borrowed Book table.

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities.

2.2. Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:
Does the existing technology sufficient for the suggested one?
Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology. Through the technology may become obsolete after some period of time, due to the fact that never version of same software supports older versions, the system may still be used. So, there are minimal constraints involved with this project. The system has been developed using Java the project is technically feasible for development.

2.3. Operational Feasibility

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following:

¬ Is there sufficient support for the management from the users?

¬Will the system be used and work properly if it is being developed and implemented?

¬Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So, there is no question of resistance from the users that can undermine the possible application benefits.

2.4. Economic Feasibility

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

The costs conduct a full system investigation.

The cost of the hardware and software.

The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also, all the resources are already available, it gives an indication of the system is economically possible for development.

2.4. Behavioral Feasibility

An estimate should be made of how strong a reaction the user staff is likely to have towards the development of a computerized system. It is common knowledge that computer installation have something to do with Turnover, Transfers and changes in employee Job Status. Normal human psychology of human beings indicate that people are resistant to change and computers are known to facilitate change. Any project formulations should consider this factor also. Before the development of the Project titled "Delhi Metro", the need to study the feasibility of the successful execution of the project was felt and thus the following factors are considered for a Feasibility Study.

1. Need Analysis.
2. Provide the users information pertaining to the preceding requirement.

CHAPTER 3

DATABASE MANAGEMENT

A Database in Salesforce is defined as the organized collection of objects where each object contains some information. Data is stored in the form of database tables for people, things, contacts, etc that are important for any project in the future. Each database has a set of certain rows and columns where information is stored in the form of fields and records. It helps produce database systems

1. That meet the requirements of the users
2. Have high performance.

One such important concept in Oracle is the Salesforce database class. Database classes can specify either you want to continue the execution or not if some error is encountered. You can also add one Boolean parameter to the Salesforce database class to make it more functional. The same task can be completed using DML statements but you are not free to keep a check on query execution.

Another important concept is the Salesforce database architecture. It has the multi-tenancy architecture where clouds are used to share resources reliably and securely. The multitenant Salesforce database architecture has a huge impact on application delivery and the infrastructure. In simple words, we can say that Salesforce database architecture is highly similar to the architecture of relational database systems.

3.1 Schema Builder

Fig 3.1 Shows the relationship between Custom Objects.

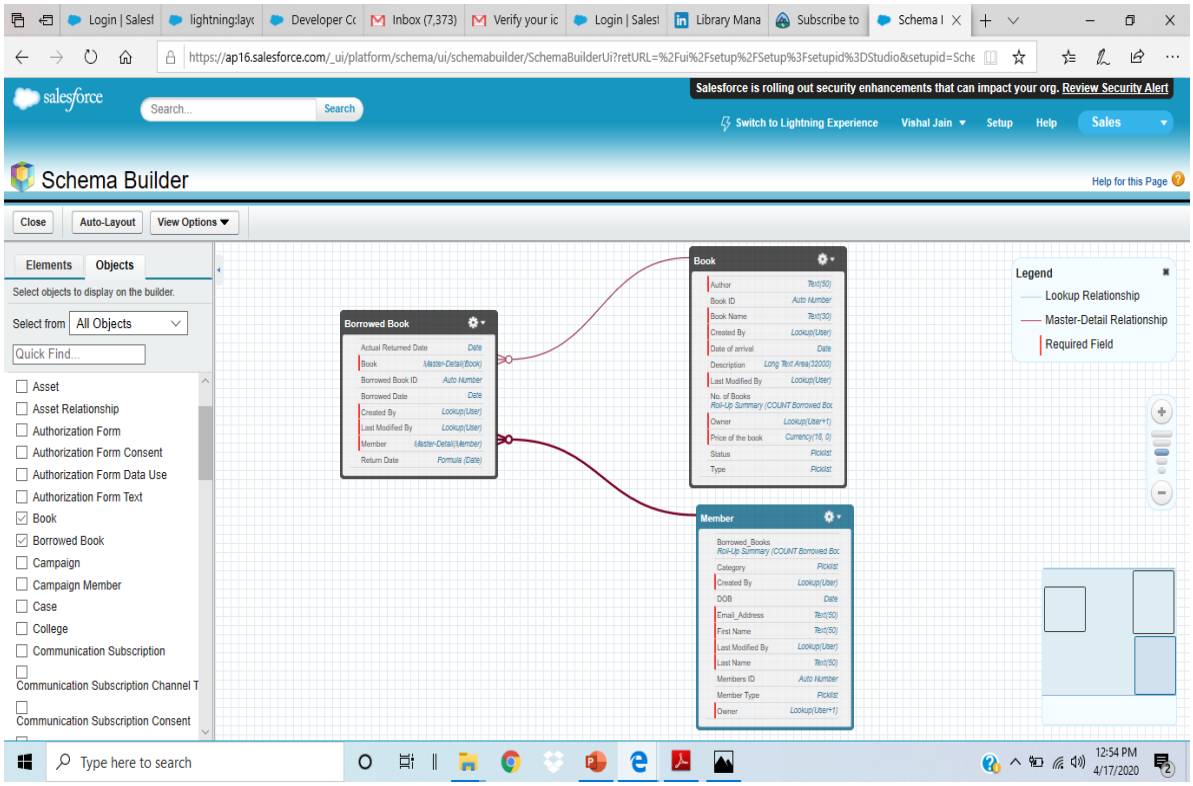

Fig. 3.1

3.2. Flow Graphs

3.2.1 Zero Level DFD

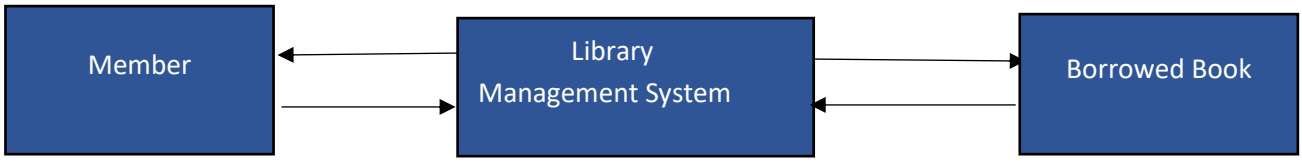

Fig. 3.2.1 Zero Level DFD

3.2.2 One Level DFD

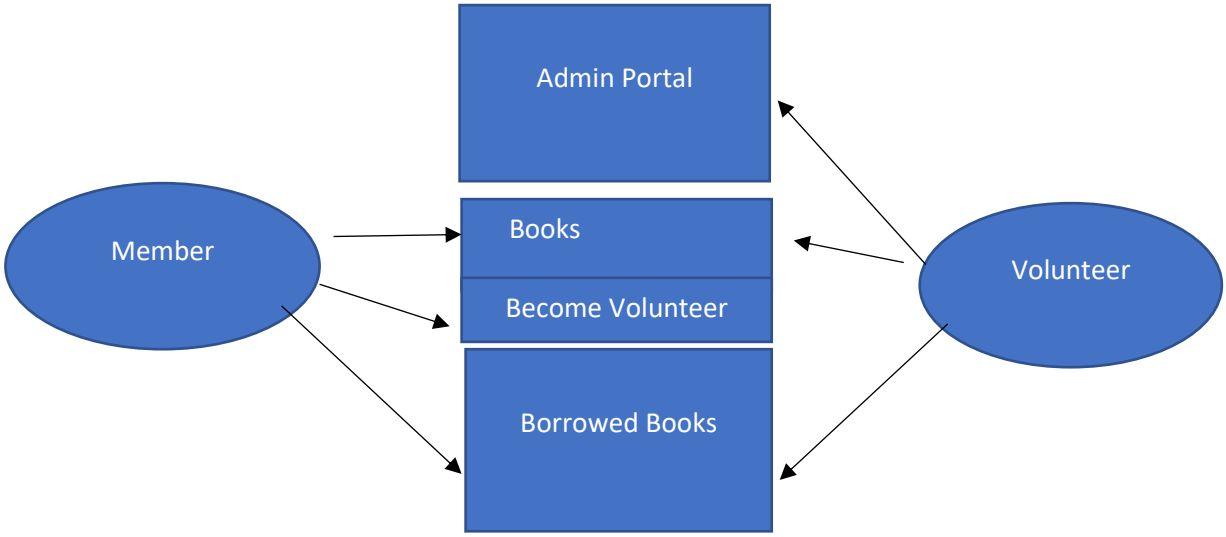

Fig. 3.2.2 One Level DFD

3.2.3 Two Level DFD

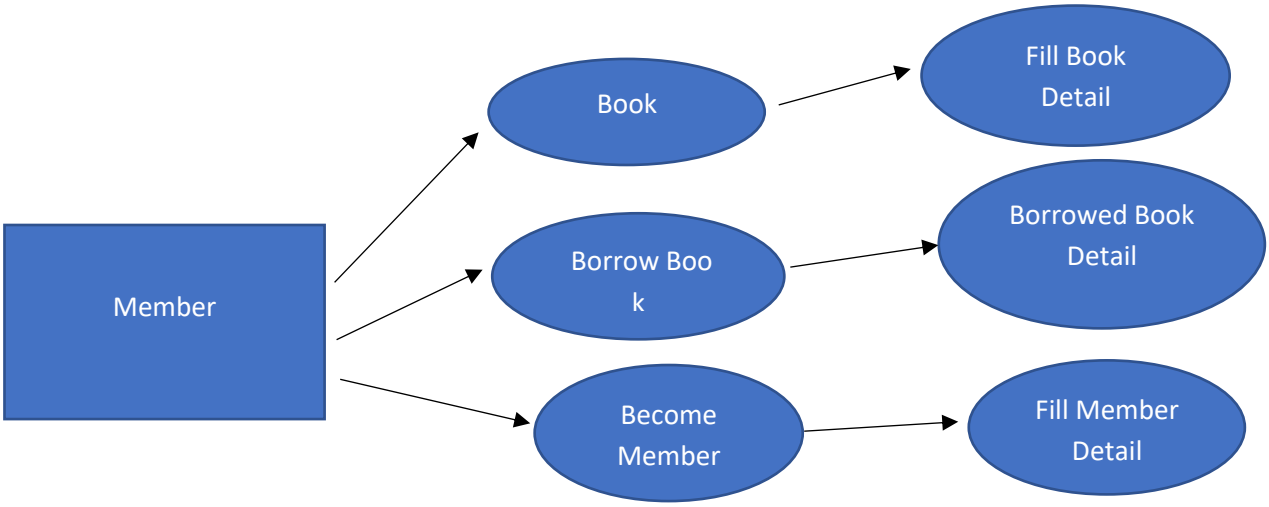

Fig. 3.2.3 Two Level DFD

CHAPTER 4

OBJECT INTERFACES

4.1. Book Object

Fig. 4.1 It show all the recent book added with shows Book Id, Book Name and Author name.

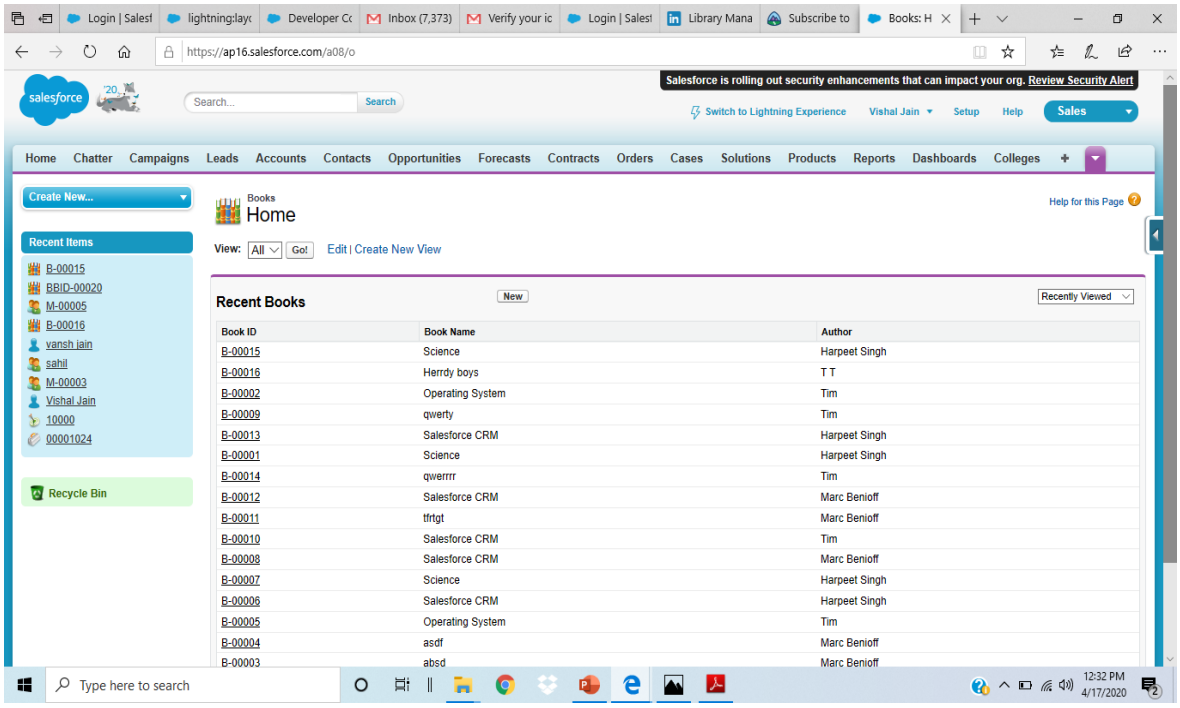

Fig. 4.1 Book Object Shows Recent Books Added

Fig. 4.1.1. In this snapshot we can see to add New Book and which type of book it is and price of it.

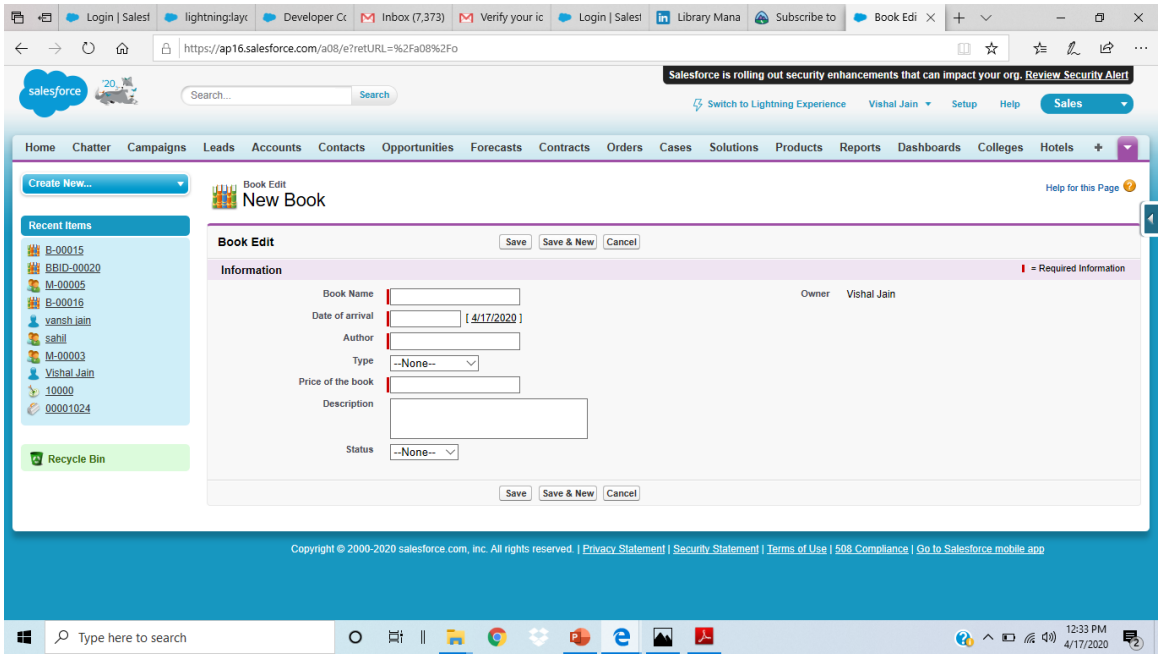

Figure 4.1.1. Adding New Book Interface

Fig. 4.1.3. This shows the book detail like how many books, book Id, number of Books etc. and show Book are Borrow by who's borrower.

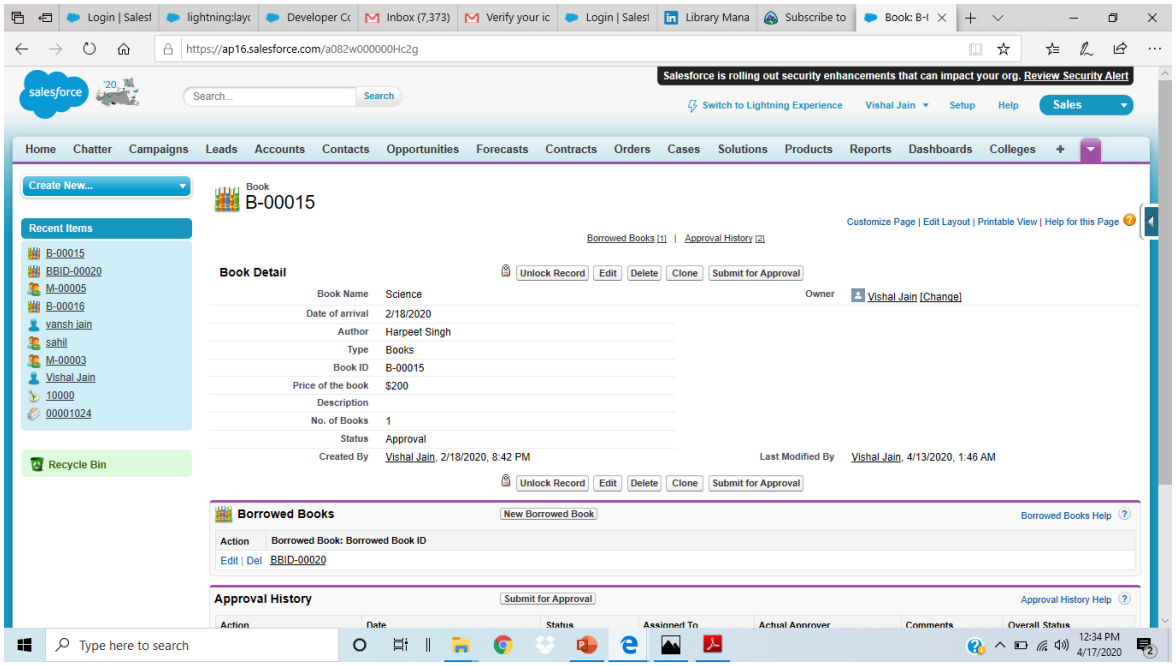

Fig. 4.1.3 Book Detail

Fig 4.2. Member object

It shows all the recent member with member Id, First name, last name and member type.

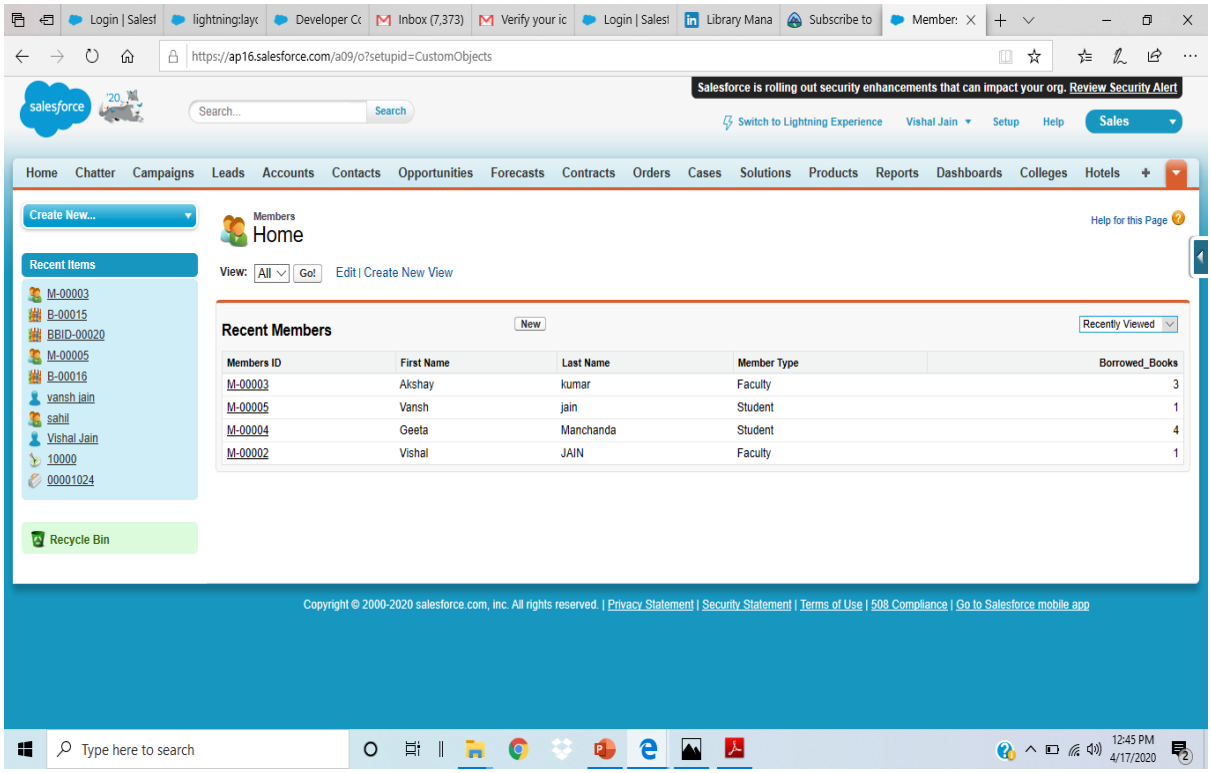

Fig. 4.2. Recent Member

Fig 4.2.1. In this figure new member are added by filled their Details in respective fields,

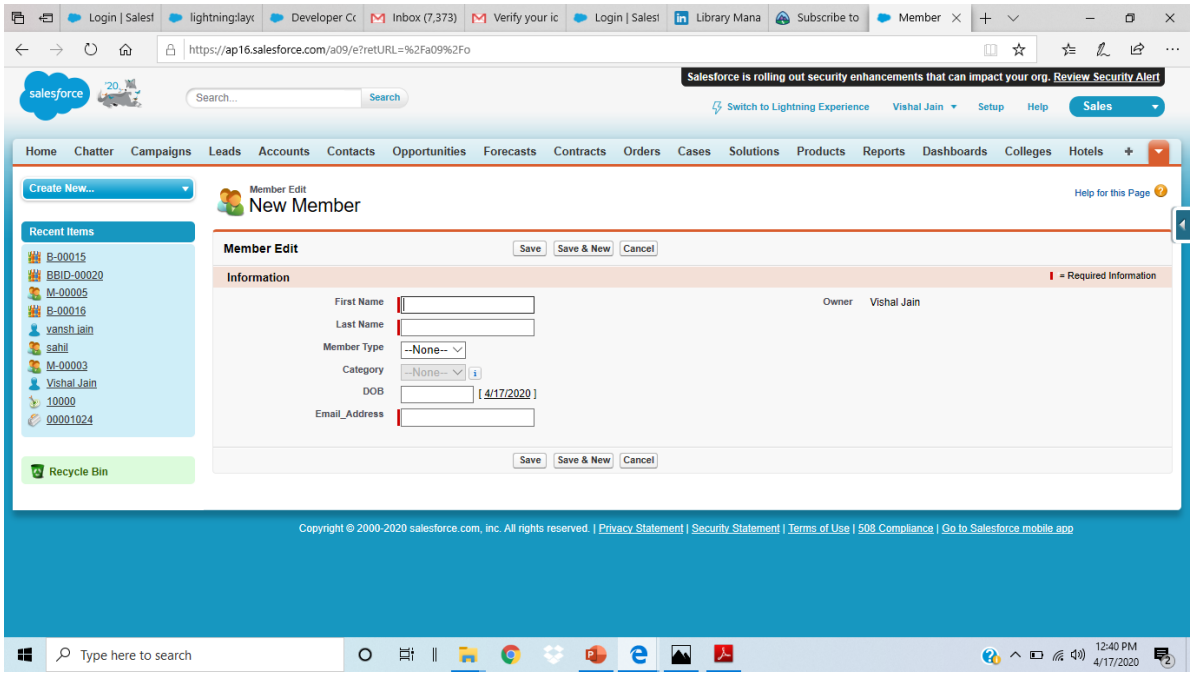

Fig. 4.2.1 Add Member

Fig. 4.3 BORROWED OBJECT

This object shows when person wants to borrow a book firstly it became a member then it can borrow a book by choose by Lookup view which contain by Book id , member id .

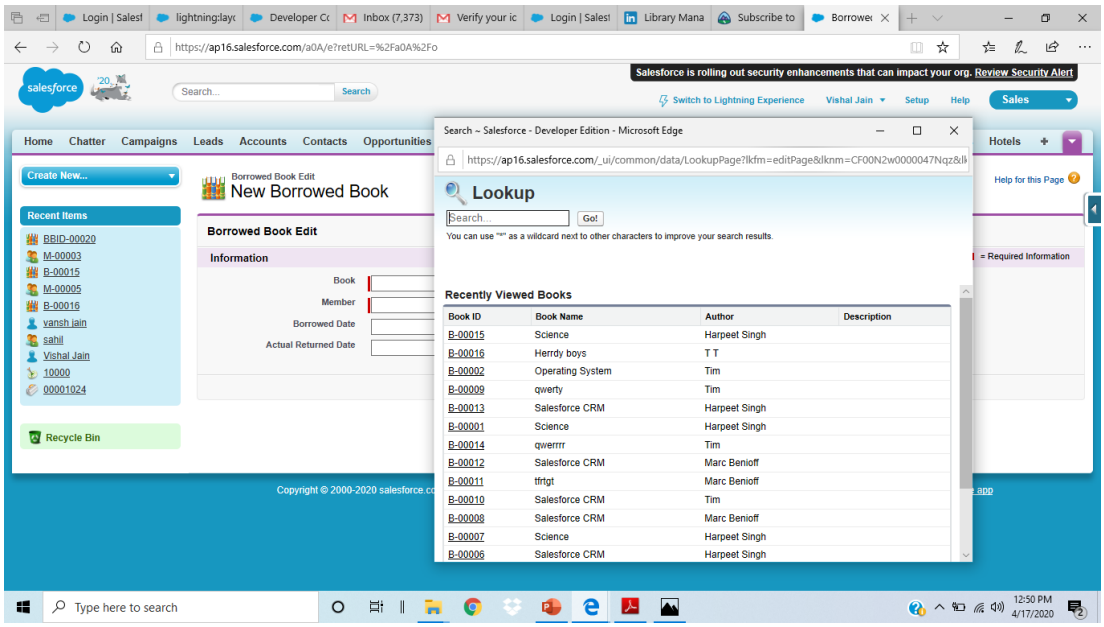

Fig. 4.3 Add Borrowed Member

CHAPTER 5

Extend Salesforce with Clicks, Not Code

Ready to go beyond the basics of Salesforce administration? Want to customize your org, push its boundaries, and enhance its functionality? You can do that and so much more without writing a single line of code. All you need is your mouse and a sense of adventure. Enhance your objects, data, and fields, customize your org's look and feel, augment your business processes, create websites, and even create apps—all using point-and-click tools.

The Force.com platform provides an enormous amount of functionality and flexibility, all of which is driven by underlying metadata. Force.com metadata is a collection of attributes that describe most components of data and applications that operate on the platform. Metadata describes the data structures in your environment, the declarative functionality implemented on the platform—even the applications you build on the platform.

To understand the power and reach of metadata, simply access a data record held in Force.com. When you access the record in your Force.com environment, the platform uses metadata to understand the structure of the record, the user interfaces defined for the record, the applications that use the user interface, even the security on that data and application.

5.1 App Setup Menu

The App Setup menu is where you will spend the most time as a developer. These menus provide access to pages that let you create and configure Force.com components and services.

Once you create a custom object, you can edit the definition of the object via the Custom Object page.

The Custom Object page provides links for adding custom fields, validations to enforce data integrity rules, database triggers, and custom buttons or links to the object's page layouts. You can also modify the attributes of standard fields, buttons, links or layouts for both the page and search dialogs, as well as add new page layouts or assign record types.

As an example, when you add a new field to a custom object, a wizard walks you through a number of steps, including:

- Selecting a field type.
- Giving the field a label, name, help text, a default value, and potentially other attributes, such as the length of the field or whether a value is required.
- Assigning security settings to the field.
- Adding the field to existing page layouts.
- Depending on the type of field in focus, the wizard may include other pages for other relevant metadata attributes.

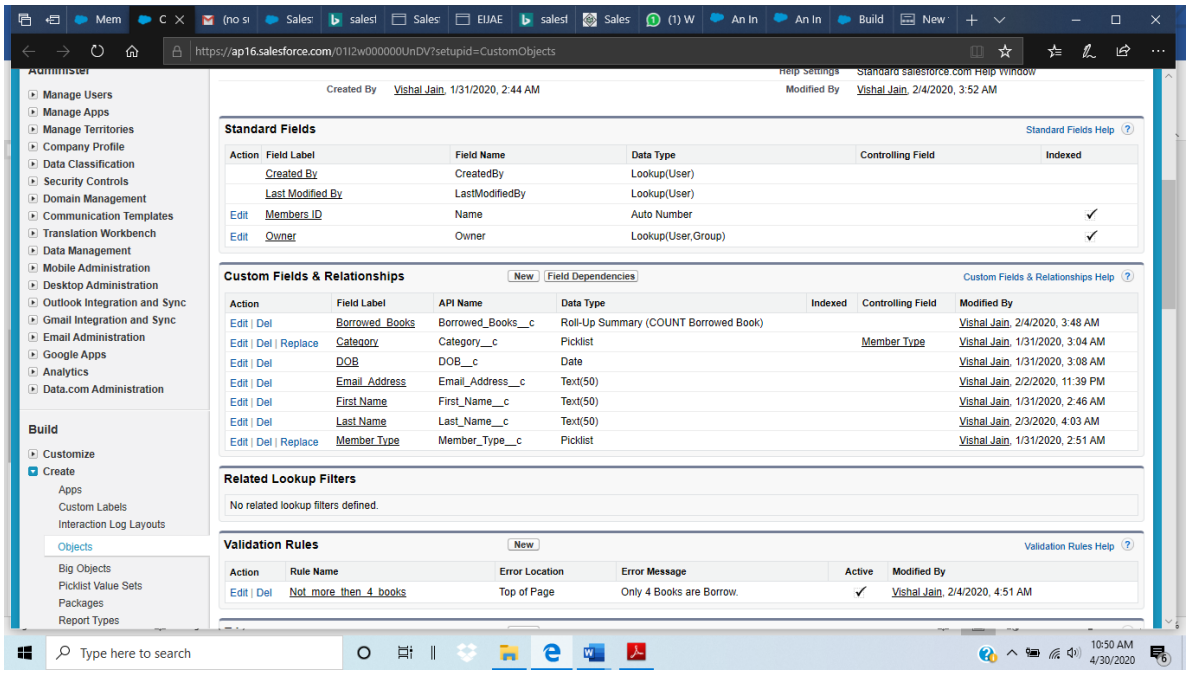

Fig 5.1

CHAPTER 6

TESTING

SalesForce testing requires the use of complex **test** methodologies as most of the features in **SalesForce** are built-in features that are customizable. When an issue is observed, the tester needs to make sure that he is **testing** the code that has been customized rather than **testing** the built-in **salesforce** functionality

SalesForce is built on a platform development language named as APEX. The language provides built-in unit test cases for developers to test their own code. The standard rule of SalesForce requires a developer to achieve 75% of code coverage with unit test cases.

From a tester's perspective, we should always aim for 100% code coverage within each test cycle.

Salesforce Testing Process

The salesforce testing process would be the same as that of testing a normal web-based application. However, a tester needs to have a clear perspective of the customizable features that are being built so that during the testing process, a tester can focus on those features alone rather than the built-in Salesforce features.

Testing of Salesforce applications requires a production like an environment called **SandBox.** Developers and testers need to use the Sandbox environment for each of their purposes.

Once the code is deployed in a Sandbox environment and approved to be ready for release, the code would be moved into production from the Sandbox environment. It is assumed that the tester has the basic knowledge of all the terms used in Salesforce before proceeding with testing.

Salesforce Testing Tips

Salesforce testing must include the following features:

- Testing must include UI testing, functional testing, regression testing, integration testing, system testing and system integration testing.
- Automation testing can also be enforced on SalesForce using tools like HP Unified Functional Testing (UFT) and Selenium.
- A tester needs to be cautious during UI testing as most of the web pages on the Salesforce platform are Visual Force pages. The dynamic nature of visual force pages need to be paid special attention as all the elements of a webpage may not be loaded at one go.
- Testers need to create functional flows including positive and negative flows to cover the entire functionality of an application.
- Workflows using various user roles must be constructed and tested.
- Test cases need to be documented using a test management tool like HP ALM.
- Test Data needs to be prepared for validating the reports functionality.

Roles and Responsibilities of a Salesforce Tester

Testers involved in SalesForce are often referred to as **'Quality Engineers'** as opposed to **'Quality Assurance'** people as SalesForce testing requires the testers to build complex test frameworks, understanding the functionality of an application in depth and the ability to work with the developers and project stakeholders.

Please note that some of the default functionalities provided by SalesForce cannot be removed, although your organization may not use them. Testers need to ignore the default functionality and focus on the customized functionality built by the organization.

Given below are some of the major roles and responsibilities of a Salesforce tester.

- A tester needs to have clear communication with the development team, to understand the customizable features that are being built into SalesForce.

- The tester needs to coordinate with the business whenever required as the requirement document for Salesforce is usually complex to understand and requires a lot of effort to be understandable by the testers.
- The tester needs to make sure that the code coverage does not go below 75% as per the standard Salesforce rule.
- The tester needs to conduct role-based testing to ensure the consistency of data with various user roles.
- The tester needs to perform compatibility testing of the third-party applications integrated with SalesForce if any.
- A tester needs to be familiar with load testing tools such as JMeter to validate the complex flows that produce inconsistent results in Salesforce.
- A tester needs to be familiar with multiple application flows.

6.1 SalesForce Exploratory Testing

Exploratory Testing in Salesforce would involve the following best practices:

- Testing should involve validating the consistency of data across multiple screens.
- UI Testing must involve documented test cases as per the requirement document.
- Testing should involve negative test flows, such as deleting the default data generated and validating the behaviour of an application.
- Testing should involve user input validation on the form fields.
- Cross browser compatibility testing needs to be performed to ensure if the rendering of data is correct across multiple browsers.
- Testing must include Maximum length validation for each of the editable input fields along with the invalid data validation.
- Testing must also include error message validation when invalid data is passed onto the applications.
- Amount field validation on banking applications using Boundary Value Analysis technique needs to be performed with proper diligence.

- Reports and dashboard testing need to be paid special attention to various test data parameters.
- Testing should include the entire application flow, along with individual functional flows.
- Multiple permutations and combinations of functional flows can be tested for positive and negative testing.
- API testing for integrated third-party applications needs to be performed.
- Identify the default Salesforce functionalities that come in the way of customized features and coordinate with the developers.

6.2 SalesForce Test Automation

Automated functional testing of SalesForce is a challenging one as most of the web pages are dynamic in nature on the SalesForce platform. Hence, SalesForce demands automation testers to build robust automation framework to sustain in the future. Also, there can be frequent updates to the applications as they are on cloud applications.

Test Automation on Salesforce can be achieved using any of the following tools:

- Selenium web driver
- HP Unified Functional Testing (UFT)
- Test Frameworks, such as Cucumber
- Provar

6.3 Salesforce Load Testing

Load testing involves testing the behavior of an application under varying loads. SalesForce.com is a highly scalable platform built for handling a large number of users. Salesforce.com is tested by the platform developers themselves for performance bottlenecks.

However, load testing becomes essential when a newly introduced piece of code yields performance bottlenecks that have to be addressed. Load Testing

on Salesforce platform can be performed using performance testing tools such as HP LoadRunner and Apache JMeter.

6.4 Salesforce Security Testing

Security testing on the Salesforce platform is usually done by SalesForce development team. Before placing a request for a security test, it is best to review the ‘Application and Network Vulnerability Assessment Summaries’ provided by Salesforce.

After reviewing the summary, if a security test is still required, then a Security Assessment Test can be scheduled with the Salesforce team.

CHAPTER 7

REFERENCES

The following references have been used by me, during all the phases of the project:

1. Krutarth Soni (2017), “Roadmap to salesforce security governance & salesforce access management”
2. Radhika Gupta (2018), “Custom Application Development in Cloud Environment: Using Salesforce”
3. Aneta Poniszewska-Maranda (2017), “Use of Salesforce Platform for Building Real-Time Service Systems in Cloud”
4. L.N. Goslin (1997), “New product development: salesforce and cycle time”
5. Steve Fisher (2007), “The Architecture of the Apex Platform, salesforce.com's Platform for Building On-Demand Applications”
6. Jigar Patel (2016), “An approach to introduce basics of Salesforce.com: A cloud service provider”
7. Amitabh Saxena (2013), “Detecting SOQL-injection vulnerabilities in SalesForce applications”
8. Jigar Patel (2017), “An integration of salesforce.com with Twitter: A case of AppExchange”
9. Kush R. Varshney (2013), “Dose-response signal estimation and optimization for salesforce management”
10. S. Fisher (2006), “Service Computing: The AppExchange Platform”
11. Alexandre Chaves Da Silva (2015), “A Case Study Using Testing Technique for Software as a Service (SaaS)”
12. Anuradha S. Kanade (2013), “Choosing right database system: Row or column-store”
13. Ming-Chien Shan (2010), “Let's Walk Out of the Cloud”
14. Sohan Singh Yadav (2010) “CLOUD: A computing infrastructure on demand”
15. Kasiviswanadhan, S (1998), “A practical approach to library automation”, Automation in Libraries and Information Retrieval Units: RlLISAR Bulietin, vol.4, no.3