

LAB REPORT

Submitted by

Aryan Gupta [RA201103010351]

Under the Guidance of

Dr. J. Rene Beulah

Assistant Professor, Department of Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING



SCHOOL OF COMPUTING
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR - 603203

JUNE 2022



SRM INSTITUTION OF SCIENCE AND TECHNOLOGY KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this lab report titled **Modern Archive** is the bonafide work done by **Aryan Gupta (RA2011003010351)** who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Dr. J. Rene Beulah
SEPM – Course Faculty
Assistant Professor, Department of Computing Technologies

ABSTRACT

No one can deny the role reading play in the life of students. Since digitization has taken over most of the components in school learning and higher education, learning through digital libraries is not a thing of the past anymore. The decline in visits to conventional libraries suggests that students prefer to access information and read content without visiting a library in person.

The Modern Archive is a digital library of Books in digital form. The purpose of this project is to let the resources reach more people by establishing an online environment for easily accessible knowledge. The material will be open sourced and user's suggestions will also be considered for future uploads in the archive.

Modern archive explores how, simply, anyone can get the materials they need. It maps the ubiquitous practice of photocopying and what are—in many cases—the more marginal ones of buying books, visiting libraries.

The objectives of modern archive are as follows –

- Online book reading.
- Material and resources uploading.
- Facility to download required material.

The combination of these modules into one website assures the perfect platform for free education and several other purposes which are: -

- Collection of Books of every category available to all for free.
- Easily accessible and better than carrying Physical books.
- Serve people who have difficulty interacting with physical books.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	iii
	LIST OF FIGURES	v
	LIST OF ABBREVIATIONS	vi
1	PROBLEM STATEMENT	1
2	STAKEHOLDERS & PROCESS MODELS	4
3	IDENTIFYING REQUIREMENTS	8
4	PROJECT PLAN & EFFORT	11
5	WORK BREAKDOWN STRUCTURE & RISK ANALYSIS	18
6	SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM	24
7	ENTITY RELATIONSHIP DIAGRAM	29
8	DATA FLOW DIAGRAM	32
9	SEQUENCE & COLLABORATION DIAGRAM	35
10	DEVELOPMENT OF TESTING FRAMEWORK/USER INTERFACE	38
11	TEST CASES	41
12	MANUAL TEST CASE AND REPORTING	44
13	ARCHITECTURE DESIGN/FRAMEWORK/IMPLEMENTATION	50
	CONCLUSION	52
	REFERENCES	53
	APPENDIX (CODE)	54

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
5.1	Work breakdown structure of Modern Archive	18
5.2	Timeline Gantt Chart of Modern Archive	19
5.3	SWOT-Analysis of Modern Archive	20
6.1	System Architecture of Modern Archive	22
6.2	Use Case Diagram of Modern Archive	23
6.3.1	Class Diagram of Modern Archive	24
6.3.2	Class Diagram of Modern Archive	24
7.1	Entity Relationship Diagram of Modern Archive	27
8.1	Data Flow Diagram of Modern Archive (Level-0)	29
8.2	Data Flow Diagram of Modern Archive (Level-1)	30
9.1	Sequence Diagram for User login	34
9.2	Collaboration Diagram	35

LIST OF ABBREVIATIONS

UML- Unified Modelling Language

EMS-Modern Archive

HR-Human Resource

HOD-Head of Department

WBS-Work Breakdown Structure

HTML-Hypertext Markup Language

CSS-Cascading Style Sheet

XML-Extensible Markup Language

MTTR-Mean Time to Repair

CAPTCHA-Completely Automated Public Turing test to tell Computers and Humans Apart

ID-Identity Document

UI-User Interface

IDE-Integrated Development Environment

UX-User Experience

API-Application Programming Interface

DB-Database

QR code-Quick Response code

DDOS-Distributed Denial of Service

SWOT-Strength Weakness Opportunities Threats

ER-Entity Relationship

DFD-Data Flow Diagram

QA-Quality Assurance



Department of Software Engineering and project management

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	1
Title of Experiment	To identify the Software Project, Create Business Case, Arrive at a Problem Statement
Name of the candidate	Aryan Gupta
Team Members	Hassan Ali, Aryan Gupta
Register Number	RA2011003010351
Date of Experiment	15-3-22

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	10	

Staff Signature with date

Aim

To Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the <title of the project>

Team Members:

S. No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Lead/Rep
2	RA2011003010357	Keshav	Member
3	RA2011003010351	Aryan	Member

Project Title: Modern Archive

Project Description:

The idea is to develop a shadow library - The online databases of readily available content that is normally obscured or otherwise not readily accessible. using the frontend support- HTML5, CSS, JavaScript and backend support – Node.js (if required) including the cloud database.

The Modern Archive is a digital library of Books in digital form. The purpose of this project is to let the resources reach more people by establishing an online environment for easily accessible knowledge. The material will be open sourced and user's suggestions will also be considered for future uploads in the archive.

Business Case

<Incorporate the Business Case template>

Result

Thus, the project team formed, the project is described, the business case was prepared and the problem statement was arrived.



Department of Computing Technologies

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	2
Title of Experiment	Identification of Process Methodology and Stakeholder Description
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	22-3-22

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	

	Total	10	
--	--------------	-----------	--

Staff Signature with date

Aim

To identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

Team Members:

SI No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep/Member
2	RA2011003010351	Aryan Gupta	Member
3	RA2011003010357	Keshav Agarwal	Member

Project Title:

Selection of Methodology

The methodology that we will be working with will be Iterative Methodology , since we know only the major requirements in our initial phase, new features can be developed during the transitional phase if required.

Incorporate information to below table regarding stakeholders of the project [Make use of below examples]

Stakeholder Name	Activity/ Area /Phase	Interest	Influence	Priority (High/ Medium/ Low)
Development team	Retain and upgrade	High	High	High

	skills. New feature experimentation, development and maintenance.			
Downloaders	Downloading the existing material available on the site.	Medium	Low	Low
Uploaders	Uploading the new material on the site after validation.	High	Medium	Medium

/ *

For Example

Stakeholder Name	Activity / Area / Phase	Interest	Influence	Priority (High / Medium/Low)
Regional Head of Sales & Marketing	Subscription using mobile App	High	High	1
Finance Account Receivable consultant	Multiple Currency Payment	High	Low	3

Interest and Influence matrix

Interest	Influence
High	High
Low	Low
Low	High
High	Low

Stakeholder	Interests	Estimated Project Impact	Estimated Priority
Owner	Achieve targets, Increase sales margin	High	1
Sponsor	Provides new market to expand ventures Negotiate funding for project Reviews changes to project environments.	Med	3
Team members	Demand incentives Retain and upgrade skills New product excitement	High	2
Project Manager	Lead the team in every aspect. Accountable for entire project scope, team, success & failure	High	2
Investors	Promoter of the investment, Provides necessary financial resources	Low	5
Resource Manager	Resource planning and allocation. Ensuring adequate resource according to project needs and budget.	Med	4
Suppliers	Ensuring feasible and realistic in every aspect Managing divergence from budgeted cost.	Med	6
End Users	Provides feedback	Low	7



Result

Thus the Project Methodology was identified and the stakeholders were described.



Department Of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	3
Title of Experiment	System, Functional and Non-Functional Requirements of the Project
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Ali Hassan
Register Number	RA2011003010351
Date of Experiment	28-3-22

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To identify the system, functional and non-functional requirements for the project.

Team Members:

S No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep/Member
2	RA2011003010351	Aryan Gupta	Member
3	RA2011003010357	Keshav Agarwal	Member

Project Title: Modern Archive

System Requirements: Basically, it's a e-library so it won't require any kind of special system.

Functional Requirements:

The website must display all kinds of resources available at that certain point.

The website must display all the key details of the particular resource.

The website must be organized with all the resources topicwise.

The website should allow the user to search the resource with keywords.

The website shall allow the user to edit a file if he finds a fault in it and brings a proof with it.

Non-Functional Requirements

Capacity - The website should support heavy traffic and multiple downloads and uploads simultaneously.

Must have -

The website should have a smooth ui with minimal bugs/errors encountered.

Could have -

The website may support upto 5000 concurrent sessions.

Availability -

Website will be available 24/7 all year long and the website will organize the database securely.
the system shall log in a customer within 10 second.

Result

Thus the requirements were identified and accordingly described.



Department of Computing Technologies

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	4
Title of Experiment	Prepare Project Plan based on scope, Calculate Project effort based on resources and Job roles and responsibilities
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	04/04/2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To Prepare Project Plan based on scope, Calculate Project effort based on resources, Find Job roles and responsibilities

Team Members:

Sl No	Register No	Name	Role
1	Ali Hassan	RA2011003010370	Lead
2	Keshav Agarwal	RA2011003010357	Member
3	Aryan Gupta	RA2011003010351	Member

Requirements

1. Project Management Plan

Describe the key issues driving the project. [Min 3 Focus Areas]

Integration Management	The model used for this project is Iterative Model. The project will be divided into 3 parts: First member will develop the frontend development of the the project. The second member will be designing the website layout and the third member will be doing the backend development.
Scope Management	Stakeholders: Manager, Investors, Project Team, Resource Manager, Employee, Administrator and Finance Manager. Project Objectives: The objective of the project is to set up a shadow library with the support of cloud database with an aim of providing the free learning material to everyone for free of charge . Schedule Objectives: Project is scheduled to be completed in 5 months Constraints: Lack of experience.
Schedule Management	Conception & Initiation: 1 week (approx.) Planning: 1 week (approx.) Execution: 3 months (approx.) Testing: 2 week (approx.) Deployment: 1 week (approx.)
Cost Management	Total Budget: Rs.50,000 (Rough estimate) Most of the technical tasks will be done on free platforms and resources. Effort: 10 hours/week distributed among all the team members.
Quality Management	The EMS will be tested for glitches and loading time. And the search algorithm will be optimized so that the navigation menu will be displayed in best possible manner.

Resource Management	People: Project team will undertake the task of planning, building and documentation of the project. The developer team will build the front end and back end portions of the project Physical: A good database with necessary facilities will be required to store data.
Stakeholder	Development team, downloaders , uploaders.
Communication Management	In the Modern Archive, users and developers can communicate using either the inbuilt chat feature in case of any discrepancy.
Risk Management	Potential technical risks will be discussed in every meeting and managed accordingly and will provide security to prevent unauthorised access.
Procurement Management	Adhering to organization procurement process. Most of the procurement process will be online.

2. Estimation

2.1. Effort and Cost Estimation

Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
Frontend	E1R1A1T1	Layout of the login page	2	1,000
	E1R1A1T2	Layout of the registration page	4	2,000
	E1R1A1T3	Main EMS UI	40	20,000
	E1R1A1T4	Navigation Menu	4	2,000
Backend	E1R1A2T1	Database creation	20	10,000
	E1R1A2T2	Employee Details	10	5,000
	E1R1A2T3	Servers	20	10,000
	E1R1A2T4	Registration and login verification	4	2,000
Integration	E1R1A3T1	Integration of frontend and backend	10	5,000
Testing and quality assurance	E1R1A4T1	Check for glitches and optimize if required	6	3,000

Note: We used COCOMO 1 Model to calculate the cost and efforts.

Effort (hr)	Cost (INR)
1	500

2.2. Infrastructure/Resource Cost [CapEx]

Infrastructure Requirement	Qty	Cost per qty	Cost per item
IR1 (Laptops)1	3	80,000	2,40,000
IR2(Laptops)2	2	35000	70,000
WorkStation (Desk, Chair, Keyboard, Mouse)	6	5000	30,000
Router	2	1500	30,000

2.3. Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Developers, Content Curators, Marketing,	3	2,000,000	6,00,000
License	Operating System Database Electronic Giants Partnership IDE Domain	10	10000	1,00,000
Infrastructures	Server, Storage and Network Office Workspace	10	20000	2,00,000

3. Project Team Formation

Name	Role	Responsibilities
Aryan Gupta	UX Designer	Design the user experience
	Content Curator	Search and create effective course roadmap.
	Tester	Perform test cases and other checks.
Ali Hassan	Cloud Architect	Network and database development
	Backend Developer	Design, Develop and Unit Test Services/API/DB
	Technical Lead	Design end to end architecture.
	Project Manager	Manage the project
Keshav Agarwal	Frontend Developer	Develop user interface
	Key Business User	Provide clear business and user requirement.
	Business Analyst	Discuss and Documentation requirements.

4. Responsibility Assignment Matrix

RACI Matrix		Team Members		
Activity		Ali Hassan (Backend)	Keshav Agarwal (Frontend)	Aryan Gupta (UI/UX)
UX Designer	C/I	C/I		R
Technical Lead	R	A		I
Business Analyst	C/I	R		C/I
Key Business User (Product Owner)	R	C/I		C/I
Backend Developer	R	C/I		C/I
Content Curator	A	C/I		R
Project Manager	R	C/I		C/I
Frontend Developer	C/I	R		C/I
Tester	C/I	C/I		R
Cloud Architect	R	C/I		C/I

A	Accountable
R	Responsible
C	Consult
I	Inform

Reference

1. <https://www.pmi.org/>
2. <https://www.projectmanagement.com/>
3. <https://www.tpsgc-pwgsc.gc.ca/biens-property/sngp-npms/ti-it/ervcpgrm-dsfvpmteng.html>

Result: Thus, the Project Plan was documented successfully.



Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	5
Title of Experiment	Prepare Work breakdown structure, Timeline chart, Risk identification table
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	11-04-22

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	10	

Staff Signature with date

Aim

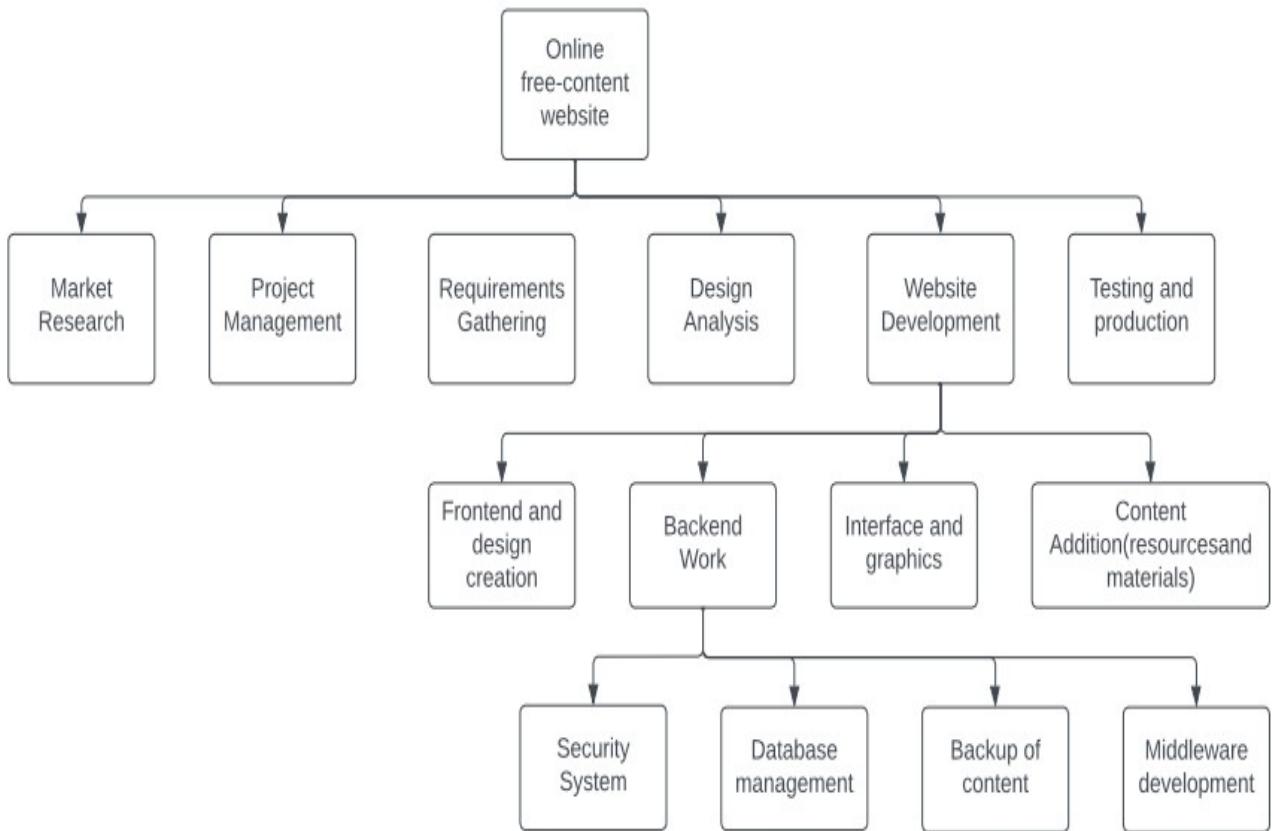
To Prepare Work breakdown structure, Timeline chart and Risk identification table

Team Members:

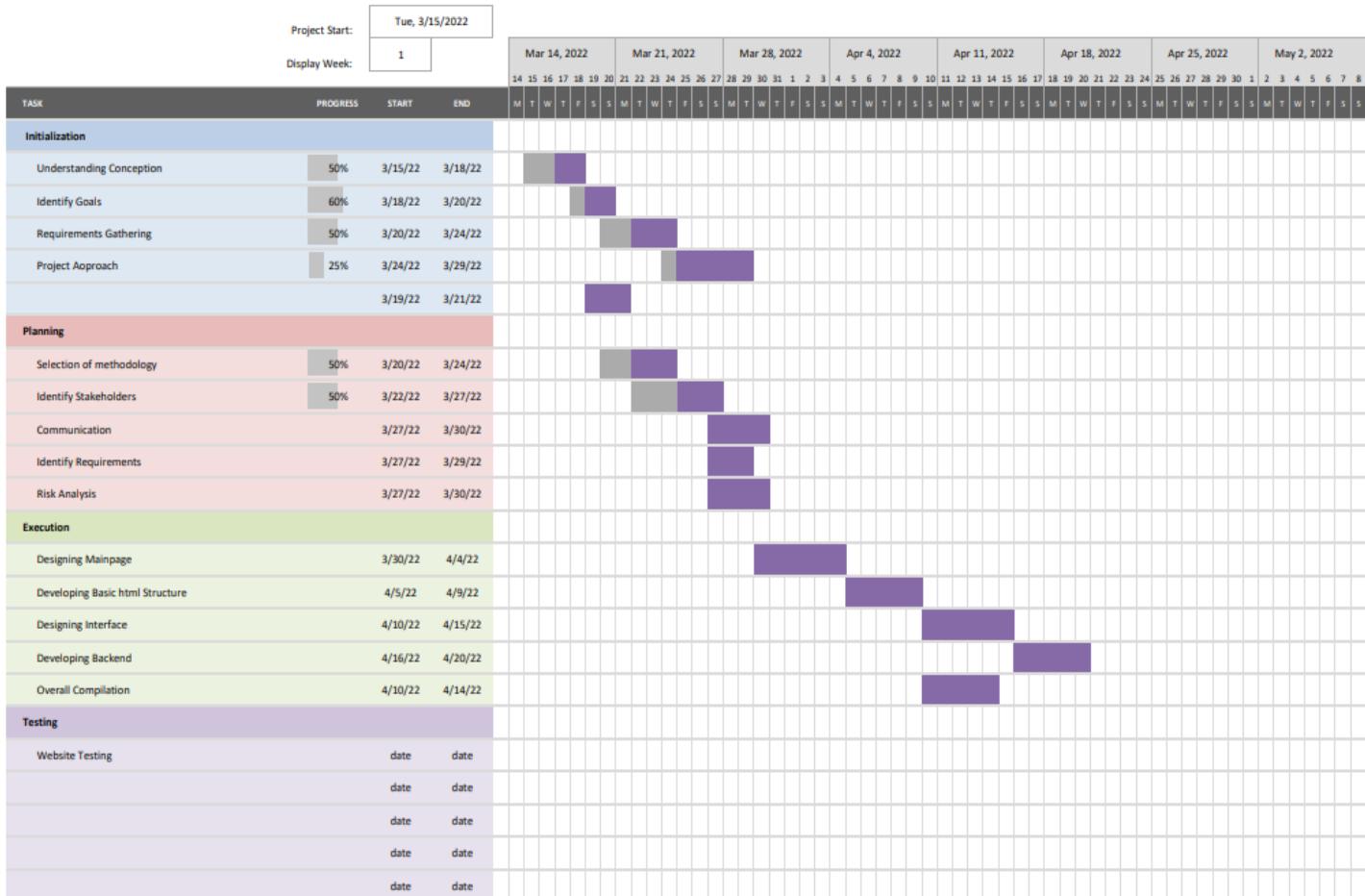
SI No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep
2	RA2011003010351	Aryan Gupta	Member

3	RA2011003010357	Keshav Agarwal	Member
---	-----------------	----------------	--------

Work Breakdown Structure



Timeline chart



Risk Identification Table

ID	Date raised	Risk description	Likelihood of the risk occurring	Impact if the risk occurs	Severity Rating based on impact & likelihood.	Owner Person who will manage the risk.	Mitigating action Actions to mitigate the risk e.g. reduce the likelihood.
1	20March22	Project purpose and need is not well-defined.	Medium	High	High	Project Sponsor	Complete a business case if not already provided and ensure purpose is well defined on Project Charter and PID.
2	22March22	Project design and deliverable definition is incomplete.	Low	High	High	Project Sponsor	Define the scope in detail via design workshops with input from subject matter experts.
3	25March22	Project schedule is not clearly defined or understood	Low	Medium	Medium	Project Manager	Hold scheduling workshops with the project team so they understand the plan and likelihood fo missed tasks is reduced.

Result:

Thus, the work breakdown structure with timeline chart and risk table were formulated successfully.



Department of Computing Technologies

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	6
Title of Experiment	Design a System Architecture, Use Case and Class Diagram
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	20-04-22

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	10	

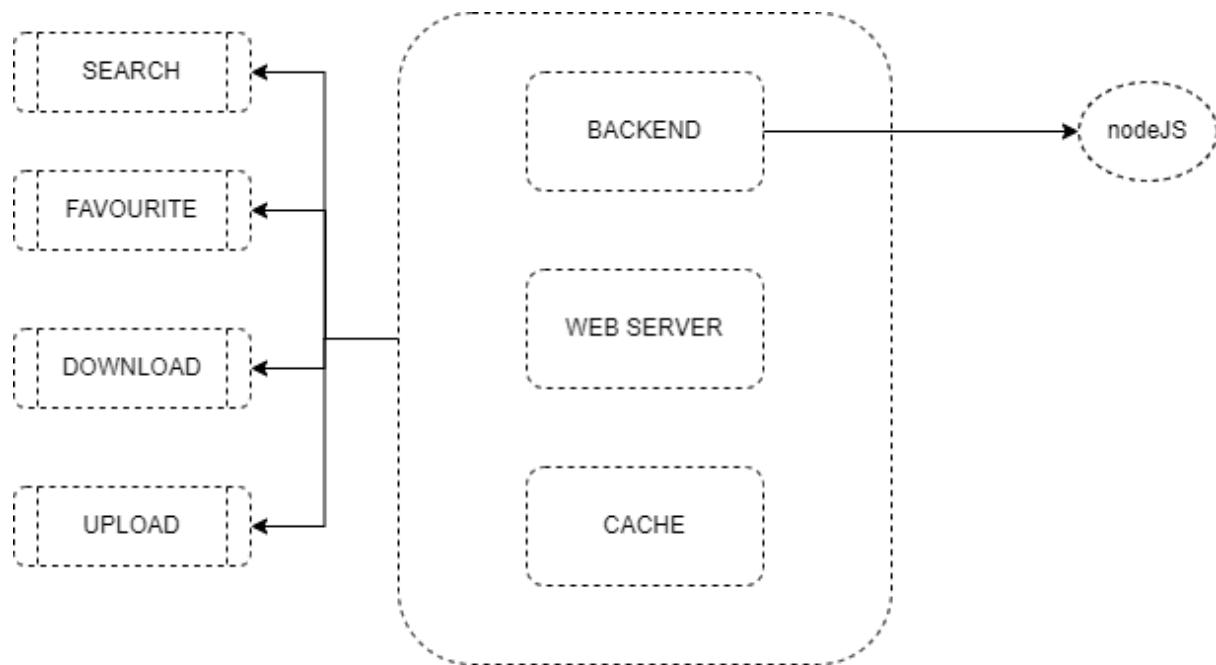
Staff Signature with date

Aim

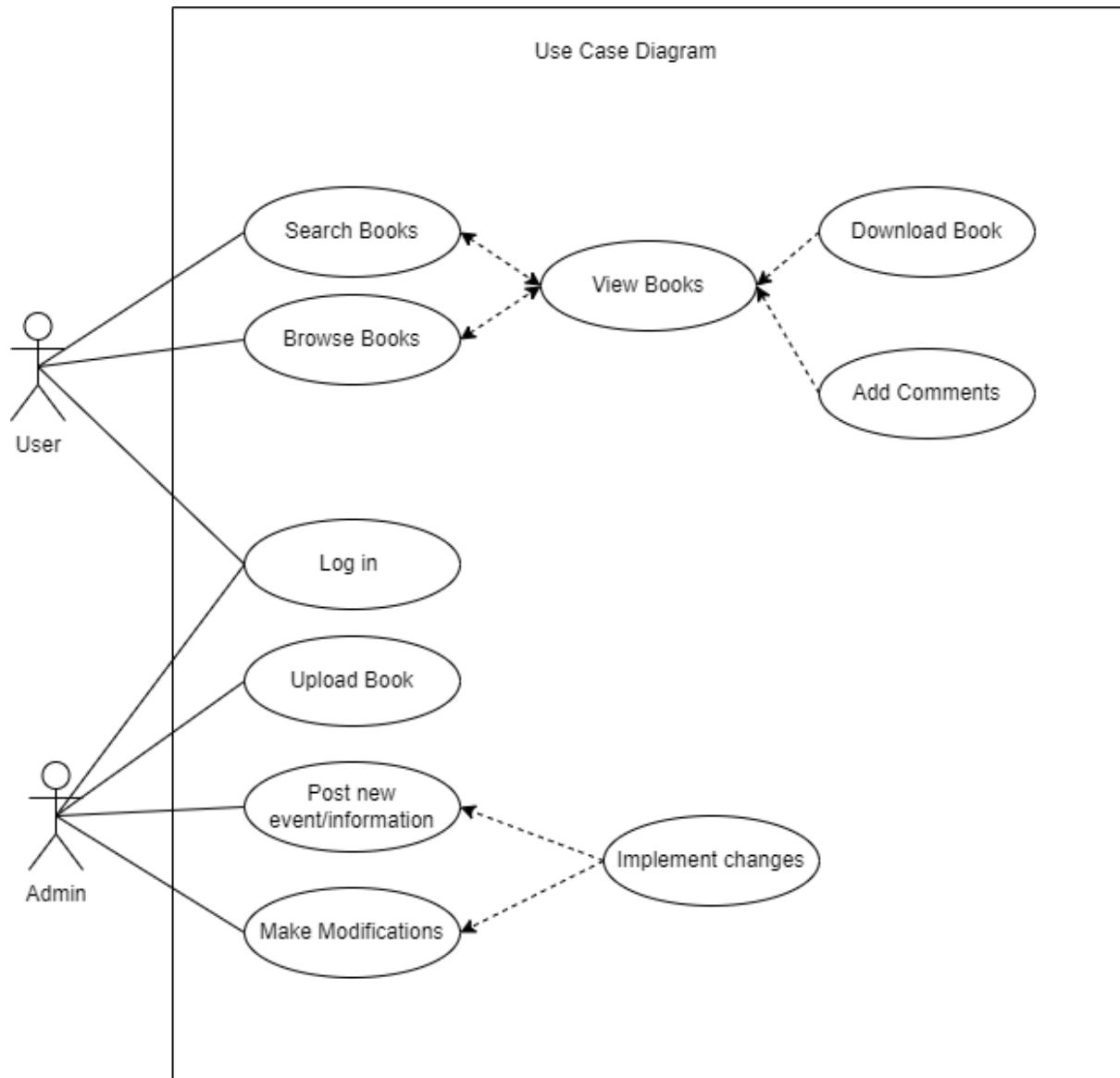
To Design a System Architecture, Use case and Class Diagram

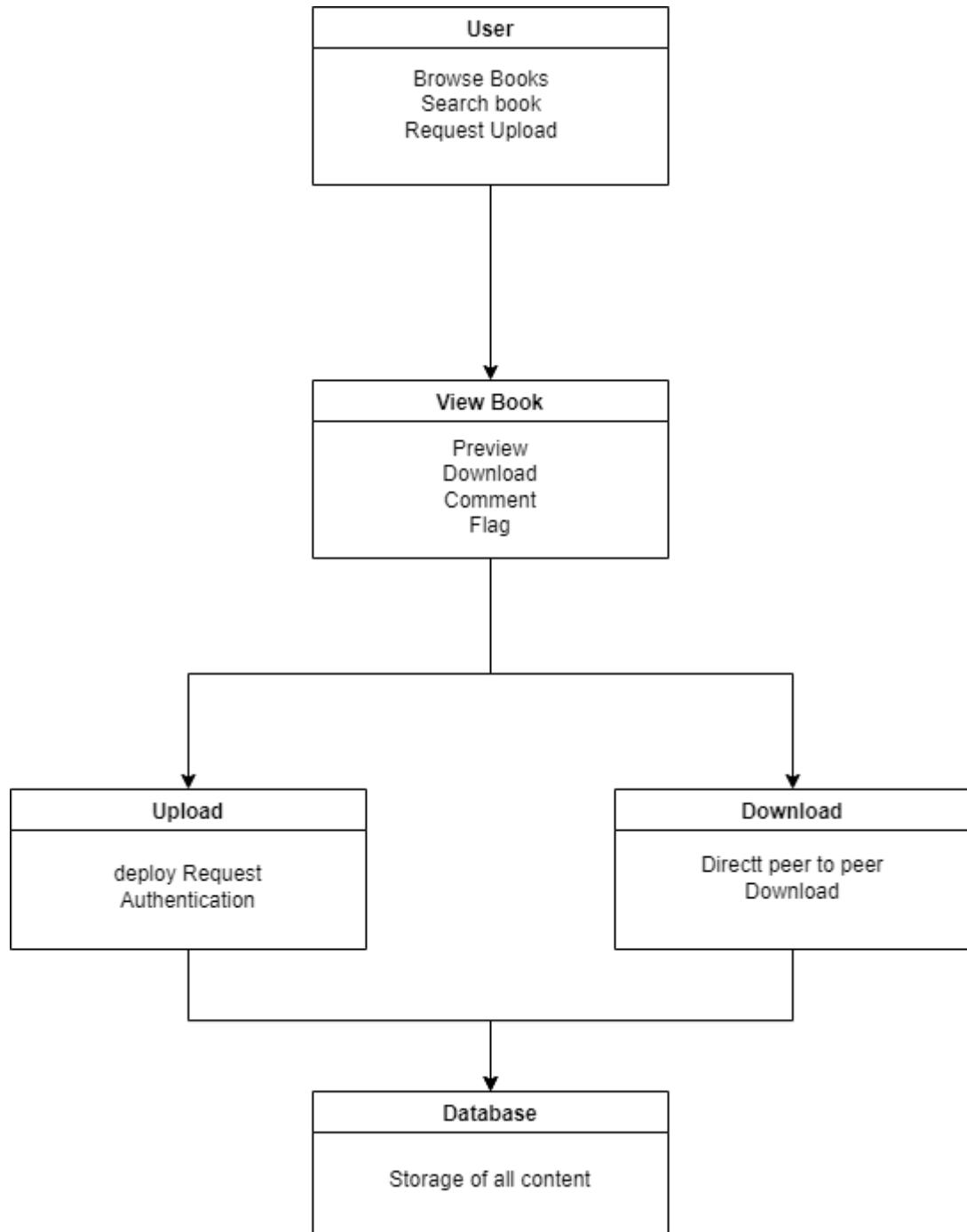
Team Members:

SI No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep
2	RA2011003010351	Aryan Gupta	Member
3	RA2011003010351	Keshav Agarwal	Member



SYSTEM ARCHITECTURE
DIAGRAM





Class Diagram

Result:

Thus, the system architecture, use case and class diagram created successfully.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	7
Title of Experiment	Design a Entity relationship diagram
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	27-04-2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

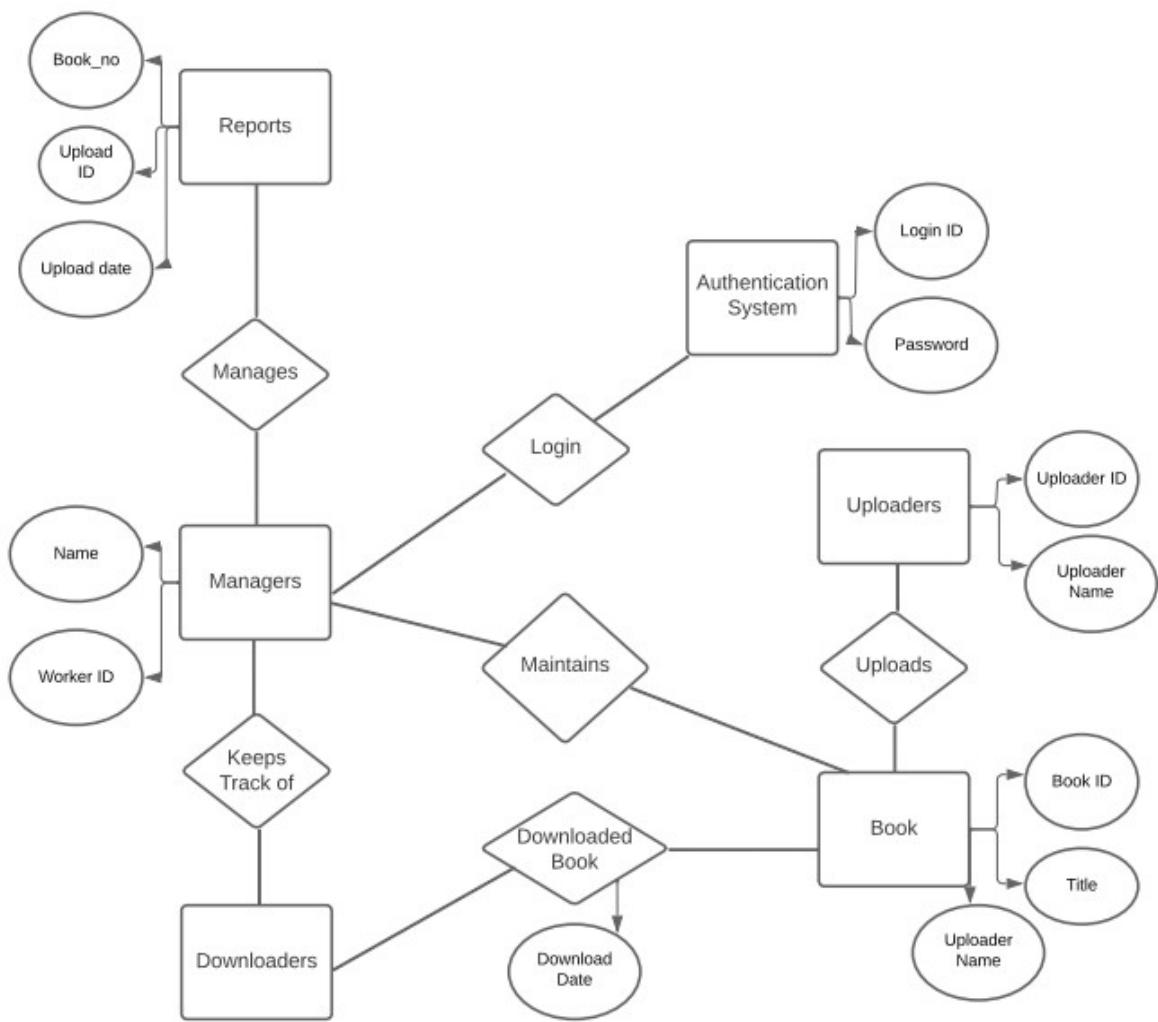
Aim

To create the Entity Relationship Diagram

Team Members:

S No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep
2	RA2011003010357	Keshav Agarwal	Member
3	RA2011003010351	Aryan Gupta	Member

ER Diagram:



Result:

Thus, the entity relationship diagram was created successfully.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	8
Title of Experiment	Develop a Data Flow Diagram (Process-Up to Level 1)
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	05-05-2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

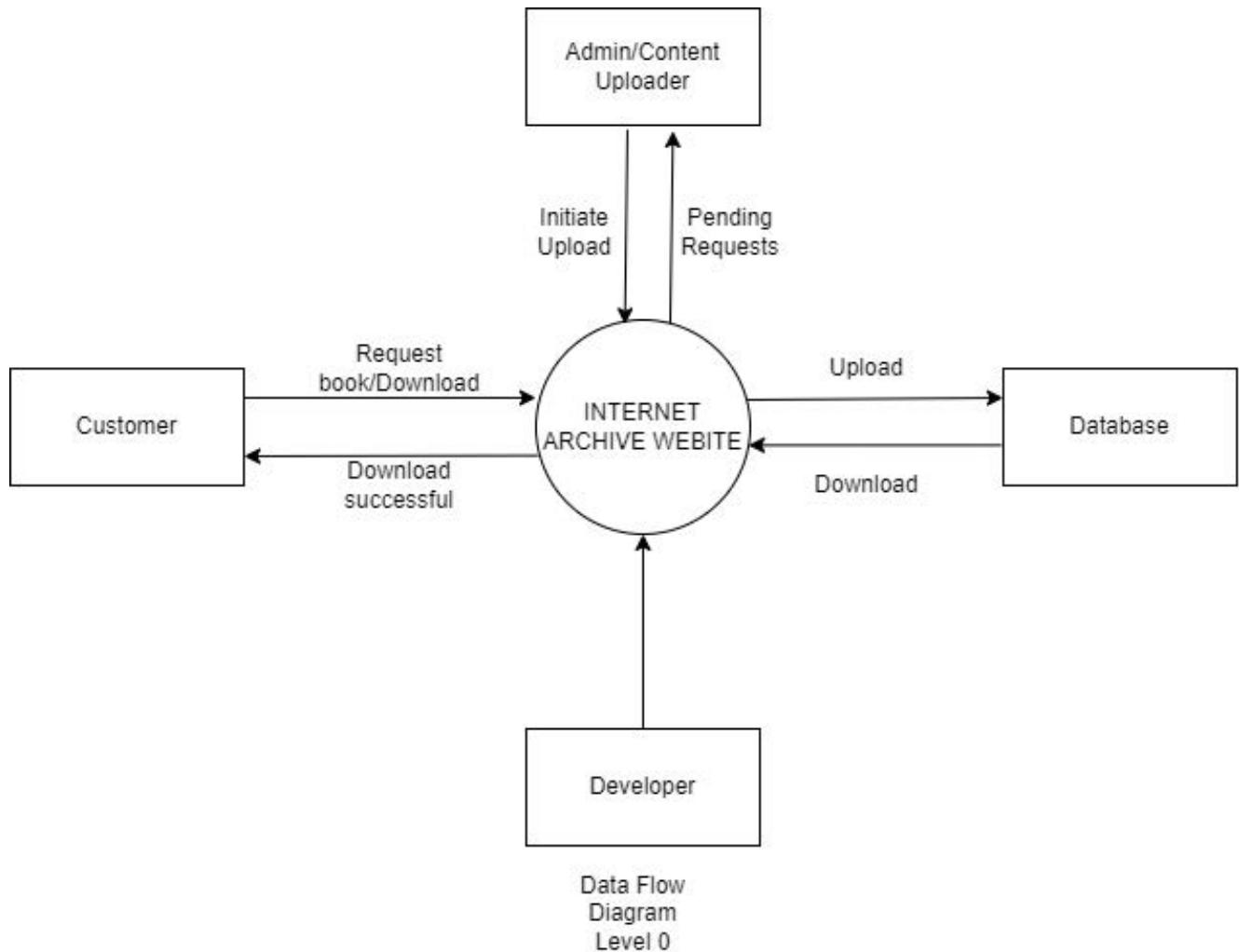
Staff Signature with date

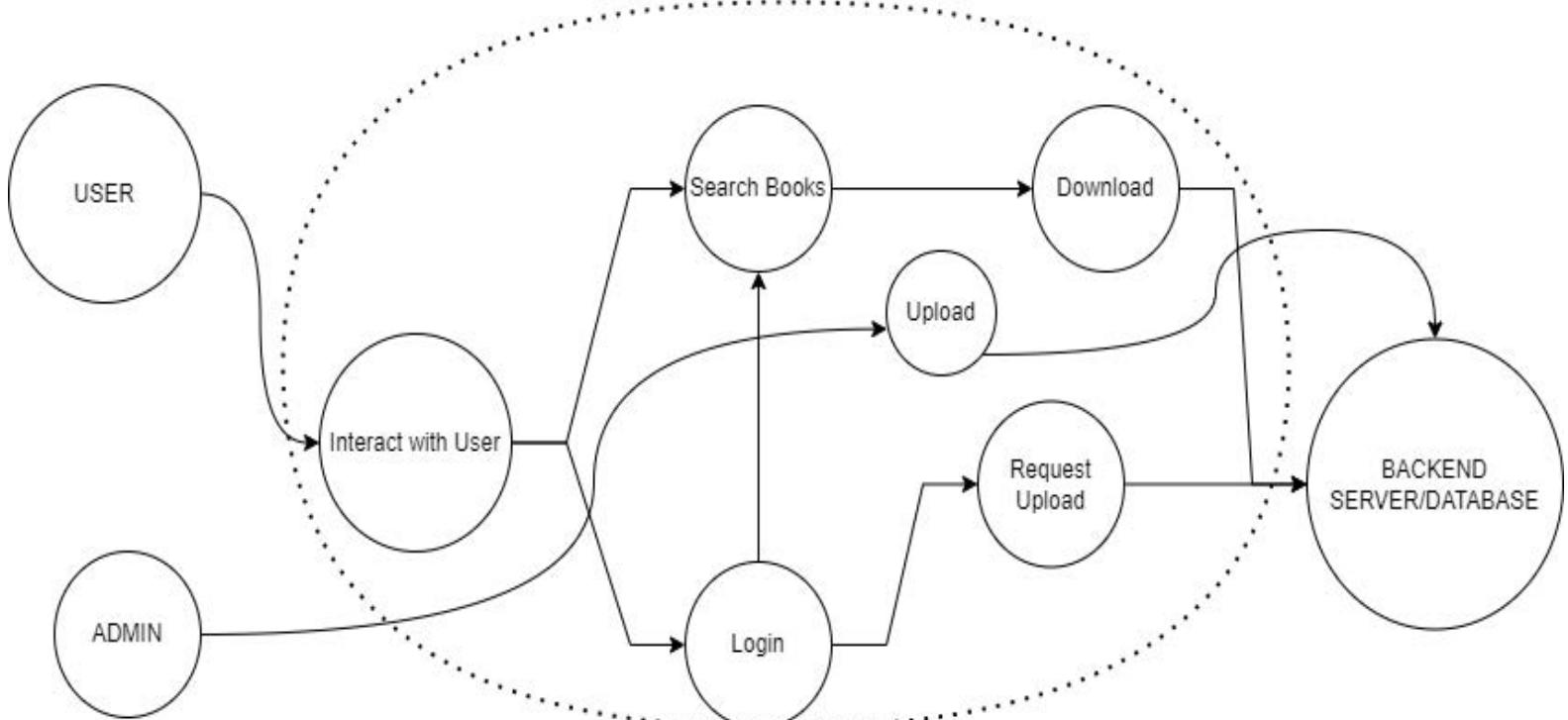
Aim

To develop the data flow diagram up to level 1 for the <project name>

Team Members:

S No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep
2	RA2011003010351	Aryan Gupta	Member
3	RA2011003010357	Keshav Agarwal	Member





Level 1

Result:

Thus, the data flow diagrams have been created for the <project name>.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	9
Title of Experiment	Design a Sequence and Collaboration Diagram
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Number	RA2011003010351
Date of Experiment	12-05-22

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

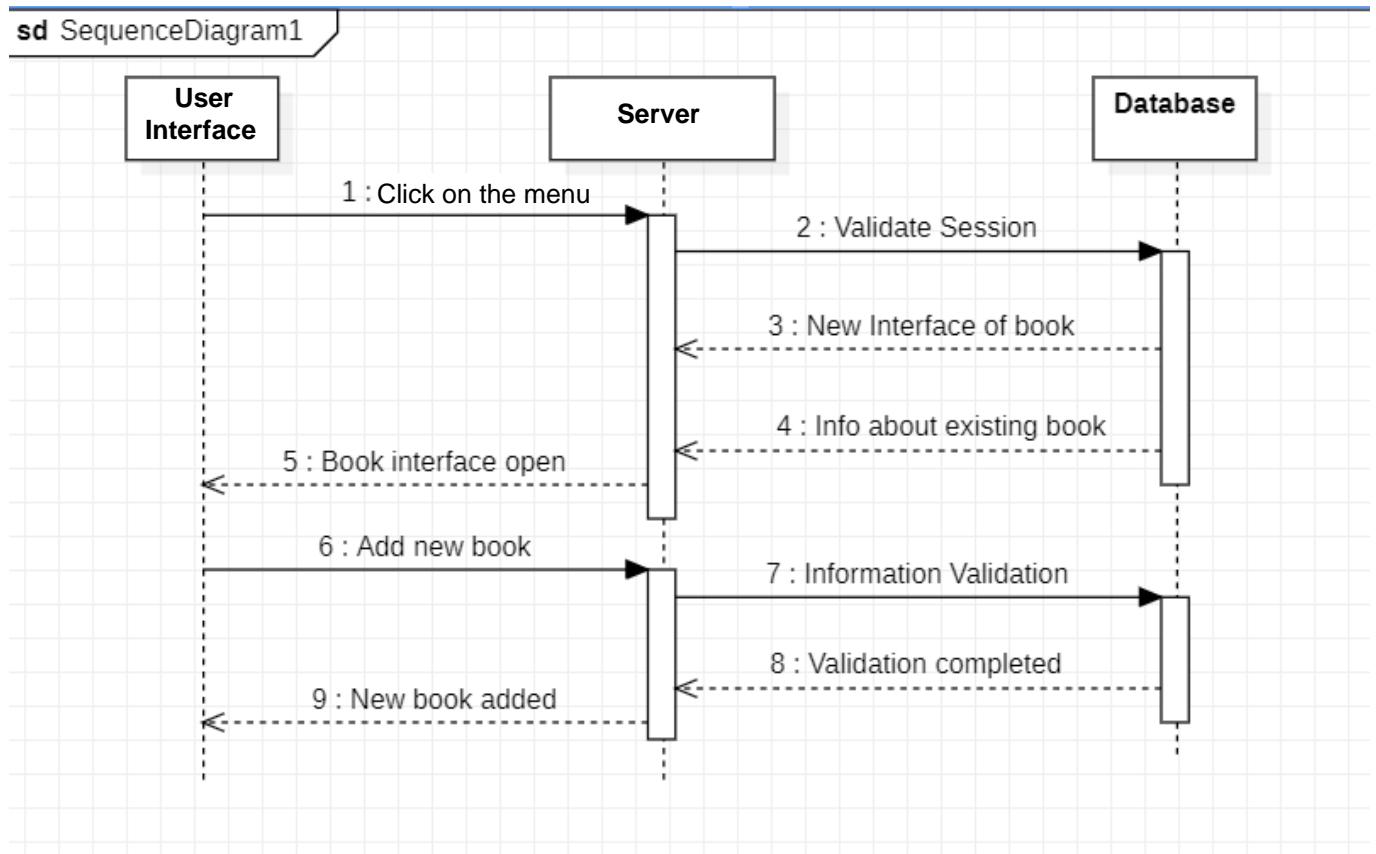
Staff Signature with date

Aim

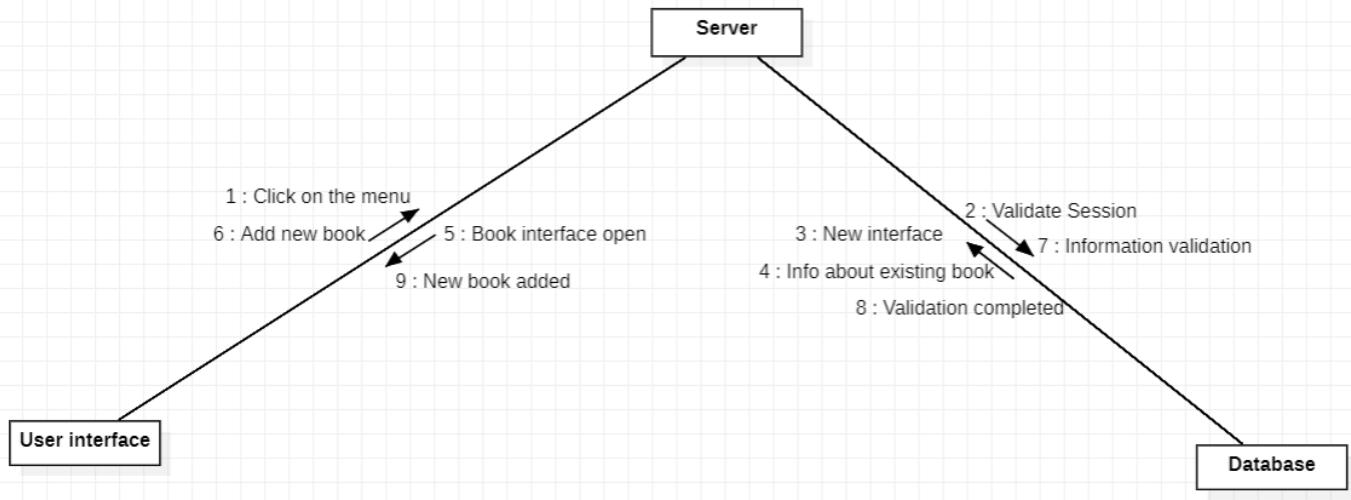
To create the sequence and collaboration diagram for the <project name>

Team Members:

S No	Register No	Name	Role
1	Ali Hassan	RA2011003010370	Rep/Member
2	Keshav Agarwal	RA2011003010357	Member
3	Aryan Gupta	RA2011003010351	Member



Collaboration Diagram:



Result:

Thus, the sequence and collaboration diagrams were created for the Modern Archive.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	10
Title of Experiment	Develop a Testing Framework/User Interface
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Ali hassan
Register Number	RA2011003010351
Date of Experiment	9-05-22

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To develop the testing framework and/or user interface framework for the <project name>

Team Members:

S No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep/Member
2	RA2011003010357	Keshav agarwal	Member
3	RA2011003010351	Aryan Gupta	Member

Executive Summary

Both functional and non-functional requirements have been tested with the help of automated and manual tools depending upon the required functionality.

Scopes Of Testing

Functional:

Test Area	Input	Methodology	Tools Required
Homepage Design Feel	-	Manual	Post Review
Login Module	Username and Password	Manual	-
Responsiveness	UI/UX	Automated	Maze, usertesting.com
Searching	Items	Manual	-
Payment Options/Security	Cards and UPI	Manual	Stripe
Logistics	Order	Manual	Time Record

Non-Functional:

Test Area	Testing Method	Tools
Usability	Automated	UserFeel
Security	Automated	SSL
Performance	Automated	LoadNinja
Scalability	Manual	

Result:

Thus, the testing framework/user interface framework has been created for the ModernArchive.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	11
Title of Experiment	Test Cases
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Ali hassan
Register Number	RA2011003010357
Date of Experiment	25-05-22

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To develop the test cases manual for the <project name>

Team Members:

S No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Rep
2	RA2011003010351	Aryan Gupta	Member
3	RA2011003010357	Keshav Agarwal	Member

Test Case

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
	Verify User Login	Accept Valid Password	1. User enters username 2. Enter the password 3. Click Login button	User details should be verified and home page should be shown	Home page shown when correct details entered	Pass	success
	Searching of Items	Valid search results form given input	1.User enters a key word or phrase into search bar 2.according to given key word, related books displayed	Key words should be cross referenced with database and relevant products showed	Items related to key words showed	pass	success

Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
	Scalability	Expanding number of users to 2000 at one time	1. 2000 users enter usernames and passwords 2. all users search for varied books	Website handles all users simultaneously with no lag and buffering	1% of users experienced 404 errors and 2% experienced slow response time	Pass	Can be improved

Result:

Thus, the test case manual has been created for the Modern Archive.



DEPT. Of Computer Science Engineering

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	12
Title of Experiment	<i>Manual Testing with report</i>
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Ali Hassan
Register Number	RA2011003010351
Date of Experiment	02.06.22

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Manual Testing	5	
2	Report	5	
Total		10	

Staff Signature with date

Aim To conduct manual test using Test cases and prepare test report for the project

Team Members:

Sl No	Register No	Name	Role
1	RA2011003010370	Ali Hassan	Lead
2	RA2011003010357	Keshav Agarwal	Member
3	RA2011003010357	Aryan Gupta	Member

1. Executive Summary

The project is tested using mostly manual testing with some automated testing. Manual testing before every deployment ensures that all modules are working as expected. This includes acceptance of proper user input as well as error handling or invalid user input. The Manual test cases are done by manually checking and running all the test cases individually by the creator.

2. Test Plan

2.1. Scope of Testing

In the scope of testing, we have defined what areas of a customer's product are supposed to get tested, what functionalities to focus on, what bug types the customer is interested in, and what areas or features should not be tested by any means. Understanding the scope of a test is crucial for our project.

Functional: Functional testing is a quality assurance (QA) process and a type of black-box testing that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (unlike white-box testing). Functional testing happens in the source code, where the system is tested against functional requirements and specifications.

Typically, functional testing includes:

1. the identification of functions that software is supposed to do
2. data input and entry
3. the execution of the test case
4. an analysis of the actual results

Non-Functional: NON-FUNCTIONAL TESTING is defined as a type of Software testing to check non-functional aspects (performance, usability, reliability, etc) of a software application. It is designed to test the readiness of a system as per non-functional parameters which are never addressed by functional testing.

2.2. Types of Testing, Methodology, Tools

Category	Methodology	Tools Required
Documentation Testing	Manual	Excel Template
Functional Requirements	Manual	Excel Template
Unit Testing	Manual	Fully functional website
Integration Testing	Manual	Fully functional website
Compatibility Testing	Manual	Fully functional website
Performance Testing	Manual	Fully functional website
Load Testing	Manual	Fully functional website

2.3. Test Deliverables

Test Case Documentation was completed. Defect Logs were recorded and Test Report was generated.

3.1. Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
T01	Unit Testing	1. Valid login credentials 2. Downloading functionality 3. Uploading Functionality 4. Sign out functionality	1. Launch the application 2. Navigate the login page 3. Enter valid username 4. Enter valid password 5. Click on login button 6. Search for required content. 7. Download or upload the content 8. Sign-out.	1. An account should be created with the information provided by the user. 2. The user should be able to search the required content. 3. The user should be able to download or upload content.	1) User was able to create an account and login into the website. 2) The user was able to search the required content. 3) The user was	Pass	Test is successful.

				the content. 4. The user should be able to sign out	able to download and upload the content. 4) The user was able to sign-out.		
T02	Integration Testing	To check if all the web pages are hyperlinked to each other	1. Open the Website 2. Check whether all the links are working or not	All links working properly	All the links were working perfectly And we were able to jump from one page to another	Pass	Test is successful.
T03	Smoke Testing	Download or Upload the content.	1. Search the content to be download ed or uploaded. 2. Complete validation. 3. Click on download or upload button.	User should be able to download or upload the desired content.	User was able to download or upload the desired content.	Pass	Test is successful.

3.2. Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Speed Test	Mean Response Time	1. User enters the URL 2. User waits for the web page to load completely	Web page should load with the mean response time of less than 1second	Response time was around 1second	Pass	Test is successful
2	Load Test	Website Should	1. Open Website and record the	Failed Test cases	Failed Test cases should	Pass	Test is successful

		not crash while loading	Number of test cases in which the web site fails to load completely on a stable metered connection. 2. The sample space is 1000	should be less than 10 (1%)	be less than 6 (0.6%)		
3	Compatibility testing	Website should be mobile responsive	1. User should be able to open our website in his/her mobile 2. Website is mobile compatible	Website is mobile compatible	Website was found to be mobile compatible	Pass	Test is successful

3. Defect Log

Requirement #	Defect ID #	Defect Description	Assignee	Status
M1R1	D01	While sign in/ sign up, if the user forgets to fill anyone of the detail, it shows an error that the field is empty.	Ali Hassan	Defect is rectified
M1R1	D02	Multiple copies of the content is downloaded in the system , even though the user requested a single instance of the content.	Aryan Gupta	Defect is rectified

4. Test Report

Testing was successful. Obstacles were presented to the stakeholders and were further looked into. Obstacles were removed.

Category	Progress Against Plan	Status
Functional Testing	Green	Completed

Non-Functional Testing	Green	Completed
------------------------	-------	-----------

Functional	Test Case Coverage (%)	Status
Module 1	100%	Completed
Module 2	100%	Completed
Module 3	100%	Completed

Result: Thus, the software test conducted and documented the report successfully



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	13
Title of Experiment	Provide the details of Architecture Design/Framework/Implementation
Name of the candidate	Aryan Gupta
Team Members	Keshav Agarwal, Hassan Ali
Register Numbers	RA2011003010351
Date of Experiment	09-06-22

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

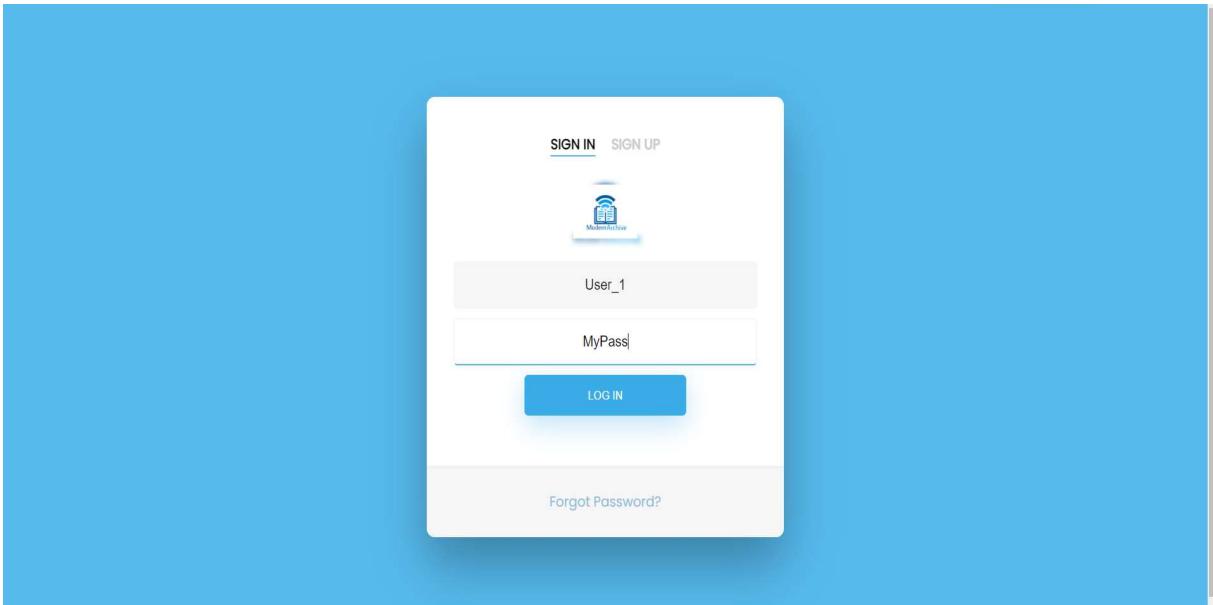
Aim

To provide the details of architectural design/framework/implementation

Team Members:

S No	Register No	Name	Role
1	Ali Hassan	RA2011003010370	Rep/Member
2	Keshav Agarwal	RA2011003010357	Member
3	Aryan Gupta	RA2011003010351	Member

Login Page.



Source code:

Html:

```
<html>
<body>
<link rel="stylesheet" href="login.css">
<div class="wrapper fadeInDown">
<div id="formContent">

    <h2 class="active"> Sign In </h2>
    <h2 class="inactive underlineHover">Sign Up </h2>

    <div class="fadeIn first">
        
    </div>

    <form>
        <input type="text" id="login" class="fadeIn second" name="login" placeholder="login">
        <input type="text" id="password" class="fadeIn third" name="login" placeholder="password">
        <input type="submit" class="fadeIn fourth" value="Log In">
    </form>

    <div id="formFooter">
        <a class="underlineHover" href="#">Forgot Password?</a>
    </div>
</div>
```

```
</div>
</div>
</div>
</body>
</html>
```

CSS:

```
@import url('https://fonts.googleapis.com/css?family=Poppins');
html {
    background-color: #56baed;
}
body {
    font-family: "Poppins", sans-serif;
    height: 100vh;
}
a {
    color: #92badd;
    display: inline-block;
    text-decoration: none;
    font-weight: 400;
}
h2 {
    text-align: center;
    font-size: 16px;
    font-weight: 600;
    text-transform: uppercase;
    display: inline-block;
    margin: 40px 8px 10px 8px;
    color: #cccccc;
}
.wrapper {
    display: flex;
    align-items: center;
    flex-direction: column;
    justify-content: center;
    width: 100%;
    min-height: 100%;
    padding: 20px;
}
#formContent {
    -webkit-border-radius: 10px 10px 10px 10px;
```

```
border-radius: 10px 10px 10px 10px;
background: #fff;
padding: 30px;
width: 90%;
max-width: 450px;
position: relative;
padding: 0px;
-webkit-box-shadow: 0 30px 60px 0 rgba(0, 0, 0, 0.3);
box-shadow: 0 30px 60px 0 rgba(0, 0, 0, 0.3);
text-align: center;
}
#formFooter {
background-color: #f6f6f6;
border-top: 1px solid #dce8f1;
padding: 25px;
text-align: center;
-webkit-border-radius: 0 0 10px 10px;
border-radius: 0 0 10px 10px;
}
h2.inactive {
color: #cccccc;
}
h2.active {
color: #0d0d0d;
border-bottom: 2px solid #5fbae9;
}

```

```
input[type=button],
input[type=submit],
input[type=reset] {
background-color: #56baed;
border: none;
color: white;
padding: 15px 80px;
text-align: center;
text-decoration: none;
display: inline-block;
text-transform: uppercase;
font-size: 13px;
-webkit-box-shadow: 0 10px 30px 0 rgba(95, 186, 233, 0.4);
box-shadow: 0 10px 30px 0 rgba(95, 186, 233, 0.4);
```

```
-webkit-border-radius: 5px 5px 5px 5px;
border-radius: 5px 5px 5px 5px;
margin: 5px 20px 40px 20px;
-webkit-transition: all 0.3s ease-in-out;
-moz-transition: all 0.3s ease-in-out;
-ms-transition: all 0.3s ease-in-out;
-o-transition: all 0.3s ease-in-out;
transition: all 0.3s ease-in-out;

}

input[type=button]:hover,
input[type=submit]:hover,
input[type=reset]:hover {
    background-color: #39ace7;
}

input[type=button]:active,
input[type=submit]:active,
input[type=reset]:active {
    -moz-transform: scale(0.95);
    -webkit-transform: scale(0.95);
    -o-transform: scale(0.95);
    -ms-transform: scale(0.95);
    transform: scale(0.95);
}

input[type=text] {
    background-color: #f6f6f6;
    border: none;
    color: #0d0d0d;
    padding: 15px 32px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
    font-size: 16px;
    margin: 5px;
    width: 85%;
    border: 2px solid #f6f6f6;
    -webkit-transition: all 0.5s ease-in-out;
    -moz-transition: all 0.5s ease-in-out;
    -ms-transition: all 0.5s ease-in-out;
    -o-transition: all 0.5s ease-in-out;
    transition: all 0.5s ease-in-out;
    -webkit-border-radius: 5px 5px 5px 5px;
```

```
border-radius: 5px 5px 5px 5px;  
}  
input[type=text]:focus {  
    background-color: #fff;  
    border-bottom: 2px solid #5fbae9;  
}  
input[type=text]:placeholder {  
    color: #cccccc;  
  
.fadeInDown {  
    -webkit-animation-name: fadeInDown;  
    animation-name: fadeInDown;  
    -webkit-animation-duration: 1s;  
    animation-duration: 1s;  
    -webkit-animation-fill-mode: both;  
    animation-fill-mode: both;  
}  
@-webkit-keyframes fadeInDown {  
    0% {  
        opacity: 0;  
        -webkit-transform: translate3d(0, -100%, 0);  
        transform: translate3d(0, -100%, 0);  
    }  
    100% {  
        opacity: 1;  
        -webkit-transform: none;  
        transform: none;  
    }  
}  
@keyframes fadeInDown {  
    0% {  
        opacity: 0;  
        -webkit-transform: translate3d(0, -100%, 0);  
        transform: translate3d(0, -100%, 0);  
    }  
    100% {  
        opacity: 1;  
        -webkit-transform: none;  
        transform: none;  
    }  
}
```

```
@-webkit-keyframes fadeIn {
  from {
    opacity: 0;
  }
  to {
    opacity: 1;
  }
}
@-moz-keyframes fadeIn {
  from {
    opacity: 0;
  }
  to {
    opacity: 1;
  }
}
@keyframes fadeIn {
  from {
    opacity: 0;
  }
  to {
    opacity: 1;
  }
}
.fadeIn {
  opacity: 0;
  -webkit-animation: fadeIn ease-in 1;
  -moz-animation: fadeIn ease-in 1;
  animation: fadeIn ease-in 1;
  -webkit-animation-fill-mode: forwards;
  -moz-animation-fill-mode: forwards;
  animation-fill-mode: forwards;
  -webkit-animation-duration: 1s;
  -moz-animation-duration: 1s;
  animation-duration: 1s;
}
.fadeIn.first {
  -webkit-animation-delay: 0.4s;
  -moz-animation-delay: 0.4s;
  animation-delay: 0.4s;
```

```
}

.fadeIn.second {
    -webkit-animation-delay: 0.6s;
    -moz-animation-delay: 0.6s;
    animation-delay: 0.6s;
}

.fadeIn.third {
    -webkit-animation-delay: 0.8s;
    -moz-animation-delay: 0.8s;
    animation-delay: 0.8s;
}

.fadeIn.fourth {
    -webkit-animation-delay: 1s;
    -moz-animation-delay: 1s;
    animation-delay: 1s;
}
```

```
.underlineHover:after {
    display: block;
    left: 0;
    bottom: -10px;
    width: 0;
    height: 2px;
    background-color: #56baed;
    content: "";
    transition: width 0.2s;
}

.underlineHover:hover {
    color: #0d0d0d;
}

.underlineHover:hover:after {
    width: 100%;
}

*:focus {
    outline: none;
}

#icon {
    width: 60%;
}

* {
    box-sizing: border-box;
```

}

User Interface.

The screenshot shows a web application for managing books. At the top, there's a navigation bar with 'Modern Archive' and 'Home' on the left, and a search bar on the right. A green banner at the top indicates a success message: 'Success: Your book has been successfully downloaded'. Below the banner, the main content area features the 'Modern Archive' logo with a blue icon of a book and a signal. There are input fields for 'Name' (Book Name) and 'Author', and a dropdown menu for 'Type' with options: Fiction (selected), Computer Programming, and Classics. Below these are two buttons: 'Download Book' (green) and 'Upload Book' (red). The section titled 'Your books' displays a table with one row, showing a book titled 'Coding with c++' by Sumita Arora, categorized under 'programming'. The table has columns for 'Name', 'Author', and 'Type'.

Source code:

Html:

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
        integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cYiJTQUhcWr7x9JvoRxT2MZw1T"
        crossorigin="anonymous">
  <title>Welcome to Modern Archive</title>
</head>
<body>
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand" href="#">Modern Archive
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
    aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
```

```

<li class="nav-item active">
    <a class="nav-link" href="#">Home <span class="sr-only">(current)</span></a>
</li>
</ul>
<form class="form-inline my-2 my-lg-0">
    <input class="form-control mr-sm-2" id="searchTxt" type="search" placeholder="Search"
        aria-label="Search">
    <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>
</form>
</div>
</nav>
<div id="message"></div>
<div class="container my-3">
    <span></span>
    <h1>Modern Archive </h1>

    <hr>
    <form id="libraryForm">
        <div class="form-group row">
            <label for="bookName" class="col-sm-2 col-form-label">Name</label>
            <div class="col-sm-10">
                <input type="text" class="form-control" id="bookName" placeholder="Book Name">
            </div>
        </div>
        <div class="form-group row">
            <label for="Author" class="col-sm-2 col-form-label">Author</label>
            <div class="col-sm-10">
                <input type="text" class="form-control" id="author" placeholder="Author">
            </div>
        </div>
        <fieldset class="form-group">
            <div class="row">
                <legend class="col-form-label col-sm-2 pt-0">Type</legend>
                <div class="col-sm-10">
                    <div class="form-check">
                        <input class="form-check-input" type="radio" name="type" id="fiction" value="fiction"
                            checked>
                        <label class="form-check-label" for="fiction">

```

```

        Fiction
    </label>
</div>
<div class="form-check">
    <input class="form-check-input" type="radio" name="type" id="programming"
        value="programming">
    <label class="form-check-label" for="programming">
        Computer Programming
    </label>
</div>
<div class="form-check disabled">
    <input class="form-check-input" type="radio" name="type" id="cooking" value="cooking">
    <label class="form-check-label" for="cooking">
        Classics
    </label>
</div>
</div>
</div>
</fieldset>
<div class="form-group row">
    <div class="col-sm-10">
        <span><button style="background-color: green;" style='margin-right:20px' type="submit" class="btn btn-primary">Download Book</button>
            <button style="background-color:red;" type="submit" class="btn btn-primary">Upload Book</button>
        </span>
    </div>
</div>
</div>
</div>
</form>
<div id="table">
    <h1>Your books</h1>
    <table class="table table-striped">
        <thead>
            <tr>
                <th scope="col">Name</th>
                <th scope="col">Author</th>
                <th scope="col">Type</th>
            </tr>
        </thead>
        <tbody id='tableBody'></tbody>
    </table>

```

```

</div>
</div>
<script src="indexes6.js"></script>
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIly6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>

```

JavaScript(E6):

```

console.log("This is ES6 version of Project Modern Archive");
class Book {
  constructor(name, author, type) {
    this.name = name;
    this.author = author;
    this.type = type;
  }
}
class Display {
  add(book) {
    console.log("Adding to UI");
    let tableBody = document.getElementById("tableBody");
    let uiString = `<tr>
      <td>${book.name}</td>
      <td>${book.author}</td>
      <td>${book.type}</td>
    </tr>`;
    tableBody.innerHTML += uiString;
  }
  clear() {
    let libraryForm = document.getElementById("libraryForm");
    libraryForm.reset();
  }
  validate(book) {
    if (book.name.length < 2 || book.author.length < 2) {

```

```

    return false;
} else {
    return true;
}
}

show(type, displayMessage) {
    let message = document.getElementById("message");
    let boldText;
    if (type === "success") {
        boldText = "Success";
    } else {
        boldText = "Error!";
    }
    message.innerHTML = `<div class="alert alert-$\{type\} alert-dismissible fade show" role="alert">
        <strong>$\{boldText\}</strong> $\{displayMessage\}
        <button type="button" class="close" data-dismiss="alert" aria-label="Close">
            <span aria-hidden="true">\&times;</span>
        </button>
    </div>`;
    setTimeout(function () {
        message.innerHTML = "";
    }, 5000);
}
}

let libraryForm = document.getElementById("libraryForm");
libraryForm.addEventListener("submit", libraryFormSubmit);
function libraryFormSubmit(e) {
    console.log("You have submitted library form");
    let name = document.getElementById("bookName").value;
    let author = document.getElementById("author").value;
    let type;
    let fiction = document.getElementById("fiction");
    let programming = document.getElementById("programming");
    let cooking = document.getElementById("cooking");
    if (fiction.checked) {
        type = fiction.value;
    } else if (programming.checked) {
        type = programming.value;
    } else if (cooking.checked) {
        type = cooking.value;
    }
}

```

```

let book = new Book(name, author, type);
console.log(book);

let display = new Display();
if (display.validate(book)) {
    display.add(book);
    display.clear();
    display.show("success", "Your book has been successfully downloaded");
} else {
    // Show error to the user
    display.show("danger", "Sorry you cannot download this book");
}
e.preventDefault();
}

```

Backend.

Table: Modern_Archive

No.	Title	Author	Subject	Publisher
1	Data Structure	Lipschute	DS	McGraw
2	DOS Guide	NORTRON	OS	PHI
3	Turbo C ++	Robort Lafore	Prog	Galgotia
4	Dbase Dummies	Palmer	DBMS	PustakM
5	Mastering Windows	Cowart	OS	BPB
6	Computer Studies	French	FND	Galgotia
7	COBOL	Stern	Prog	John W
8	Guide Network	Freed	.NET	Zpress
9	Basic for Beginners	Norton	Prog	BPB
10	Advanced Pascal	Schildt	Prog	McGraw

Source code:

```

const express = require("express");
const mysql = require("mysql");
const DATABASE = require("./utilities/createDB");
const TABLES = require("./utilities/createTables");
const cred = require("./utilities/credentials");
class Modern_Archive {
    constructor(port, app) {
        this.port = port;
        this.app = app;
        this.app.use(express.json());
    }
}

```

```

this.temp = 0;
new DATABASE().initDB();

new TABLES().initTable();
this.db = mysql.createConnection({
  ...cred,
  database: "Modern_Archive",
});
}

get() {

this.app.get("/api/getBooks", (req, res) => {
  let sql = `SELECT * FROM book`;
  this.db.query(sql, (err, result) => {
    if (err) console.log(err);
    else console.log("Successfully extracted books");
    res.send(result);
  });
});

this.app.post("/api/Download", (req, res) => {
  let sql = [
    `INSERT INTO DOWNLOAD(idUser, idBook) VALUES (${req.body.sid}, ${req.body.id})`,
    `Update BOOK SET count = count - 1 WHERE id = ${req.body.id}`,
  ];
  for (let i = 0; i < sql.length; i++) {
    this.db.query(sql[i], (err, result) => {
      if (err) {
        console.log("Couldn't Download");
        this.temp = 1;
      } else console.log("Successfully Downloaded");
    });
    if (this.temp) break;
  }
});

this.app.get("/api/getIssues/:sid", (req, res) => {
  let sql = `SELECT book.name, book.author as sname\
    FROM book, User\
    where Download.idUser ='${req.params.sid}' and book.id = Download.idBook and User.id =

```

```

`${req.params.sid}`;

this.db.query(sql, (err, result) => {
  if (err) console.log(err);
  else console.log("Successfully extracted issues");
  res.send(result);
});

});

this.app.post("/api/return", (req, res) => {
  let sql = [
    `SELECT deadline from Upload\`  

     WHERE idBook = ${req.body.id} and idUser = ${req.body.sid}\`,  

    `DELETE FROM Download where idUser = ${req.body.sid} and idBook = ${req.body.id}\`,  

    `UPDATE BOOK SET count = count + 1 WHERE id = ${req.body.id}\`,  

  ];
  for (let i = 0; i < sql.length; i++) {
    this.db.query(sql[i], (err, result) => {
      if (err) {
        console.log("Couldn't return");
      }
    }
  }

  else if (i == 0) {
    var d1 = new Date(result[0].deadline);
    var d2 = new Date();
    const timeDiff = d2 - d1;
    const daysDiff = Math.ceil(timeDiff / (1000 * 60 * 60 * 24));
    if (daysDiff > 0) {
      this.db.query(
        `UPDATE User SET Limit = Limit + ${  

          (daysDiff - 1) * 10  

        } WHERE id = ${req.body.sid}\`,  

        (err, result) => {
          if (err) console.log(err);
          else console.log("Limit Updated Succesfully");
        }
      );
    }
  });
}
);
});

```

```

this.app.get("/api/Users/:id", (req, res) => {
  let sql = `SELECT User.name\
    FROM User\
    where Download.idBook = '${req.params.id}' and User.id = Download.idUser`;
  this.db.query(sql, (err, result) => {
    if (err) console.log("Couldn't get issues");
    else console.log("Successfully extracted issues");
    res.send(result);
  });
});

listen() {
  this.app.listen(this.port, (err) => {
    if (err) console.log(err);
    else console.log(`Server Started On ${this.port}`);
  });
}

let Modern_Archive = new Modern_Archive(3001, express());
Modern_Archive.get();
Modern_Archive.listen();

```

Result:

Thus, the details of architectural design/framework/implementation along with the screenshots were provided.