BOOK BANK

1. OBJECTIVE:

- To implement book bank system online.
- To provide the students with the information on books and scheduled times of book rental available.
- To ensure validity and security in the online book bank management system.

2. SCOPE OF THE PROJECT:

The book bank system project is a software tool created to help and access the students gather required information about the various books in the institution the project is web based interactive application.

Focus is laid solely on the book display, schedule, categories, syllabus and payment as per required.

3. PROJECT DESCRIPTION:

USER PERSONAL AND CHARACTERISTICS:

The target client for our software are students who lively are interested in rental of books be it either from engineering or medical genre in the institution. They can log into our website and browse the courses which they have aspiration to study. The student must register themselves online, which is free. The booking or reservation of the books is online, the detailed description about their book package and necessary terms and condition norms would be emailed to the students registered email address which he/she may provide during the time of registration.

PRODUCT PERSPECTIVE:

- This product requires a web browser which is capable of playing flash media.
- This program will not be dependent on any other software and is not a component of another program.
- This program does not require any new hardware.
4. REQUIREMENTS:

(a). FUNCTIONAL REQUIREMENTS:

A functional requirement defines a function of a software system on its component. A function is described as a set of input, the behavior and output.

i. A main menu including a brief help section.
ii. Login
iv. Displaying details.
vi. Logout.

1. REQUIREMENTS:

The system should have the requirements of the project. The developer should prepare the requirements of the project. The should prepare the requirements which are need for the software.

2. ANALYSIS:

Analyze the requirements whether it provides proper operations/output and performs the task.

3. DESIGN:

Project manager should design the layout of the project before going to implement time allocation, cost allocation and staff allocation will coming under design process.

4. IMPLEMENTATION:

After encomposing all the diagrams, we have to generate code for each and every diagrams i.e from usecase to deployment.

5. TESTING:

After implementing the diagram with domain language, we have to test the particular projects.

6. MAINTAINENCE:

The system should be easily updated. The system should utilize the interchangeable plugins software developed should maintain the cost and time schedule of the project.
(b). NON-FUNCTIONAL REQUIREMENTS:

Nonfunctional requirements define the needs in terms of performance, logical database requirements, design constraints, standard compliance, reliability, availability, security, maintainability and portability.

i. PERFORMANCE REQUIREMENTS:

Performance requirements define acceptable response times for system functionality.

- The total time for user interface screens will take no longer than two seconds.
- The login information shall be verified within the seconds.
- Queries shall results within five seconds.

ii. DESIGN CONSTRAINTS:

The software shall be a standard system running in a windows environment. The system shall be developed using rational enterprise suite and oracle 10i database.

iii. RELIABILITY:

Specify the factors required to establish the required reliability of the software system at time of delivery.

iv. AVAILABILITY:

The system should have an availability of 99.99%.

v. PORTABILITY:

- The system should be extremely via the usb drive.
- The system shall be easy to migrate or backed up via another use drive.

vi. MAINTAINABILITY:

- The system shall utilize interchangeable plugins.
- The system shall be easily updateable for fixes and patches.

(c). HARDWARE REQUIREMENTS:

1. Processor - Intel Pentium IV-2.0 GHZ.
2. Hardware - 40 GB.
3. RAM - 512mb
4. DVD RAM - 1 nos.
(d). SOFTWARE REQUIREMENTS:

1. OS - windows XP/vista.
2. Front end Tool - Rational Rose Enterprise suite.
3. Back end Tool - Oracle 10i.

5. MODULE DISCRIPTION:

In this project we have defined different modules to enable the Book Bank system in successful manner.

i. REGISTER:

The register module contains the application form or registration form which contains following details.

- Name, Address, Contact number, E-mail id, Password etc.

ii. LOGIN:

The Login module contains the form which contain membership name and member password.

It includes Username and Password.

iii. SEARCH BOOK:

The search book module contain list of books, from this list we search for the book which we need. This also contains another field called as categories where can select the category of the book.

iv. DISPLAY DETAILS:

Display the details about the students particulars, the payments, the books, rental and schedule times for books etc.

v. MAINTAIN BOOK DETAILS:

The administrator maintains the details of books.

vi. LOGOUT:

To sign off from the webpage or your account log off.

6. DOMAIN MODEL:

A domain model is a visual representation of conceptual classes or real situations object in a domain.

- In object oriented analysis, the domain model is the most important.
- It illustrates the concept in the domain.
- It act as a source of inspiration for designing some software objects.
7: PARTIAL LAYERED ARCHITECTURE:

Sequence diagram is an interaction overview diagram. It provides a big picture overview of how a set of interaction are related in terms of logic and process flow.

This Partial layer architecture shows the interface of the sequence diagram, here the administrator shows the interface by displaying actor symbol.

8: LOGICAL ARCHITECTURE:

The Logical architecture is large scale organization of software class into packages, sub system layer. It is called logical architecture because there is no direction about how these elements are display on different operating system.

9: RISK ACTIVITY:

- Personal short falls.
- Unrealistic schedules and budgets.
- Developing the wrong functions and programs.
- Developing the wrong user interface.
- Continuing stream of requirements changes.
- Short falls in externally furnished component.
- Real time performance short falls.
- Straining computer science capabilities.

RISK ASSESSMENT:

- Figure out what the risks are and what to focus on.
- Making a list of all of the potential dangers that will affect the project.
- Assessing the probability of occurrence and potential ross of each item listed.
Virtual Tutor

**RISK CONTROL:**

- Monitoring the effectiveness of the strategies and the changing levels of risk throughout the project.
- Techniques and strategies to migrate the highest ordered risks.

10: **GANTT CHART:**

It describes the time schedule for the planning to complete the corresponding and after completion of core product, what the time is taken for the project action of core product.

11: **POST- FUNCTION AND PRE-FUNCTION:**

**MODULE:** Registering

**PRE-FUNCTION:**

- Login to the website.
- Collection the required documents to be submitted for registration.

**POST-FUNCTION:**

- Verification of documents submitted.
- Conformation email sent accessing that authentication can be prevailed for the individual.

**MODULE:** Display book details

**PRE-FUNCTION:**

Analyze the course of semester of logger.

**POST-FUNCTION:**

Display the required book details.
12: UML PACKAGE DIAGRAM:

A package diagram provides a way to group elements. Here we have grouped the 6 main elements of a software project: register, login, search book. It goes to update and sales records. A package name may be based on a tab if the package shows the inner member of the main package.
13: TECHNICAL SERVICES LAYER:

Technical service layer describes the relationship between different actors, components of the software process for any admin seek the registration for new visitor. So that the new visitor can login the website and search for book and if need can buy it.

14: DOMAIN SERVICES LAYER:

After technical services layer from partial layered architecture, we are going to generate the coding in java or VB the project domain is now finalized in JAVA/VB.

15: USER INTERFACE LAYER:

i. ADMIN:
The administrator is used to register a new visitor for the website. Admin can keep those records update. Admin keep sales record i.e., now much book is downloaded and how much book is buyed newly with copy write etc.

ii. REGISTERED USER:
Registered user can search the book whatever needs and can read the book in online. If user is satisfied with this book, can download the book but after payment.

iii. VISITOR:
The visitor can register the application form and become a member of the website. The visitor can search books and read the book but he must register to buy or download the book.
UML USECASE DIAGRAM:

Uml provides use case diagram notation to illustrate the names of use case and author relationship between them. Use case diagram and case relationship are secondary in use case work use case text document.
**UML CLASS DIAGRAM:**

The Uml class diagram is to illustrate class interfaces as their actions. They are used for static object modeling, we have already introduced and used their uml diagram while domain modeling.
UML SEQUENCE DIAGRAM:

A sequence diagram illustrate a kind of format in which each object interact via message. It is generalize between two or more specialized diagram.
PARTIAL LAYER ARCHITECTURE DIAGRAM:

Sequence diagram is an interaction overview diagram. It provides a big picture overview of how a set of interaction are related in terms of logic and process flow.

This Partial layer architecture shows the interface of the sequence diagram, here the administrator shows the interface by displaying actor symbol.
UML COLLABRATION DIAGRAM:

COLLABRATION diagram illustrate that object interact on a graph or network format in which object can be placed where the diagram. In collaboration diagram the object can be placed in anywhere on the diagram. The collaboration comes from sequence diagram.
UML STATE CHART DIAGRAM:

A Uml state machine represents the interaction events and states of an object and behavior of an object in reaction to an event.. Transaction shown as allows labeled with their event. It is included with initial pseudo state and finds end state.
UML ACTIVITY DIAGRAM:

A Uml activity diagram shows sequential and parallel activates in a process, work flows, data flows and compiler algorithm.
UML COMPONENT DIAGRAM:

Components are slightly fuzzy concept in this Uml, because both class and components can be used to model the something.
UML DEPLOYMENT DIAGRAM:

Deployment diagram shows the assignment of concrete software artifact to computational nodes. It shows the deployment of software elements to the physical elements. Deployment diagram are useful to communicate or deployment architecture.

![Deployment Diagram](image-url)
IMPLEMENTATION:

After the completion of UML diagrams for BOOK BANK SYSTEM we should generate the code. Select the tools option from main menu then select the sub option program module in which we are going to generate the code in JAVA/VB/J2EE. Then go ahead and select the generate code option as a result of code generation.

TESTING:

To perform the testing for the generated code again select the tools option from menu bar and choose quality architecture option a sub window gets opened. We are going to perform Unit Test and Scenario testing for our project. So, select these testing option one by one and testing activity is carried out for all the coding and testing is carried out.

CONCLUSION:

Thus the project for BOOK BANK SYSTEM has been successfully executed and codes are generated.