Department of Computer Science & Engineering
Chitkara University Institute of Engineering & Technology
(Accredited by NAAC with Grade 'A+')

CHITKARA UNIVERSITY

Ref No: CUIET/CSE/ACAD/2025/177 Date: 03rd April 2025

NOTICE
ATTENTION: - BE CSE Batch 2023
BE CSE 4th Semester- PROJECT BASED EVALUATION

Dear Students,

Project-based assessments are an essential part of your university experience, aimed at filling the gap between theory and practice. They provide you with a special chance to illustrate your comprehension of central computer science concepts through the creation of practical solutions. This practical work not only reinforces your course material but also develops valuable skills such as problem-solving, critical thinking, and collaboration, skills that are well sought after in the industry.

Thus, we are declaring the imminent project-based assessment for the following topics:

S. No.	Course Name (Course Code)	Remarks	Total Marks
1	Data Structures (23CS004) and Programming in JAVA (23CS005)*	Students are required to submit one (1) project demonstrating their understanding and application of concepts from both Data Structures and Programming in JAVA. ***Refer to Annexure-A for more details***	40 marks *(Sum of marks of ST-3 of DS and JAVA as per CHO)*
2	Linux Administration (22CS009)*	Students are required to submit one (1) project showcasing their proficiency in Linux administration tasks and concepts. ***Refer to Annexure-B for more details***	20 marks *(ST-3 as per CHO)*
3	Object Oriented Software Engineering (22CS017)*	Students are required to submit one (1) project illustrating their ability to apply Object Oriented Software Engineering principles and methodologies. ***Refer to Annexure-C for more details***	20 marks *(ST-3 as per CHO)*
4	Backend-Engineering-I (23CS006)	There is NO CHANGE in the project evaluation for Backend Engineering. Students will continue with the project as previously assigned and communicated (notice no CUIET/CSE/ACAD/2025/98, dated 25.02.2025).	40 marks *(Final Project Based Evaluation as per CHO)*

***Project Teams:** Students will form **ONE COMMON TEAM of 3-4 members** to develop projects for these three/four subjects (*S.No. 1-3)*. This single team will work together on individual projects for each of these courses.

Important Dates/Deadlines:

S. No.	Task	Deadline
1	**Project team** communication to the mentor	7th April 2025
2	**Project title** finalization and approval from faculty/trainer	20th April 2025
3	**Project completion** deadline *(for all projects)*	11th May 2025
4	**Final evaluation** *(of all projects)* *(Detailed evaluation schedule will be shared in due course of time)*	12th May to 20th May 2025

-Sd-
Prof. (Dr.) Rishu Chhabra: Dean (CSE-Beta Cluster)
Prof. (Dr.) Vikas Khullar: Associate Dean (CSE-iBeta Cluster)

Annexure-A

Data Structures (23CS004) & Programming in JAVA (23CS005):

Please find below the rubrics and report format for the Programming in Java (23CS005) and Data Structures (23CS004) project as per the syllabus.

1.	Guidelines for the Project	-Annexure A1
2.	Rubrics for Project based evaluation	-Annexure A2
3.	Report Format	-Annexure D

Note:

- For Project Evaluation, students (in teams) are required to develop console based Java project that incorporates data structures and algorithms with CRUD operations. This evaluation will involve the data structure concepts.
- Subsequently, the same project will be considered for both subjects (Programming in Java and Data structures).
- Project team may consist 3- 4 members. In case student is absent in project evaluation, he/she will be given zero in the ST3 for both subjects (under all rubrics).
- Students have to submit a soft copy of the report (refer to Annexure D), one from each group.

Annexure A1- Guidelines for the Project

Sample
Title of the project Library Management System

1. **Introduction**- console-based Java project that incorporates data structures and algorithms with CRUD operations:

2. **Problem Definition and Requirements**- This project simulates a library management system where users can manage books, patrons, and borrowing transactions.

3. **Proposed Design / Methodology** –
DSA Concepts to Implement:

1. **Linked Lists**: For maintaining collections of books, patrons, and transactions
2. **Binary Search Trees**: For efficient book searching by ID or title
3. **Queues**: For handling book reservation requests
4. **Sorting Algorithms**: For displaying books by various criteria (title, author, publication date)
5. **Searching Algorithms**: To find books and patrons

CRUD Operations:

1. **Books Management**:

- Create: Add new books to library
- Read: View book details and availability
- Update: Modify book information or status
- Delete: Remove books from system

2. **Patron Management**:
    - Create: Register new library members
    - Read: View patron details and borrowing history
    - Update: Update patron information
    - Delete: Remove patrons from system
3. **Borrowing Transactions**:
    - Create: Check out books to patrons
    - Read: View current borrows and history
4. **Results**- Successfully execution of the project

Annexure A2- Rubrics for the Project based evaluation

Component	**Criteria**	**Points**
Logic Building (30%)	Implementation of library workflows	10
	Error handling and edge cases	10
	Input validation	10
Clean Code (10%)	Code organization and structure	3
	Naming conventions and readability	3
	Comments and documentation	4
DSA Implementation (30%)	Correct implementation of data structures	10
	Efficiency of algorithms	10
	Appropriate DSA selection for use cases	10
CRUD Operations (30%)	Completeness of operations	15
	Transaction consistency	5
	Data persistence implementation	10
Total		**100**

Annexure-B

Linux Administration (22CS009):

Please find below the rubrics and report format for the Linux Administration (22CS009) project as per syllabus.

1. Guidelines for the Project -Annexure B1
2. Rubrics for Project based evaluation -Annexure B2
3. Report Format -Annexure D

Note:

- For Project Evaluation, students (in teams) are required to develop a console-based/GUI Shell script-based project that incorporates various Linux commands to automate the system administration task. This evaluation will involve the Linux Administration concepts.
- The project team may consist of 3- 4 members.
- Students have to submit a soft copy of the report (refer to Annexure D), one from each group.

Annexure B1- Guidelines for the Project

Project Statement: Bash User Management Application: Automating Linux User Tasks.
The GUI interface in Linux shell script simplifies the user management tasks using a command-line interface with dialog boxes as GUI interface for improved usability. The script will allow system administrators to add, remove, modify, and list users efficiently while reinforcing basic shell scripting concepts such as loops, conditionals, functions, and input handling.

Objectives:
The main objectives of this project are:

- Automate Linux user management tasks.
- Provide an interactive and easy-to-use interface.
- Utilize different dialog box alternatives for user interaction.
- Implement security measures to prevent unauthorized access.
- Maintain logs for auditing user management actions.

Scope
This project focuses on user management in a Linux system, ensuring administrators can:

- **Add a New User** – Create a user with a username, password, and optional home directory.
- **Delete a User** – Remove a user account with a confirmation prompt.
- **Modify User Information** – Update details like username, password, or group membership.
- **List Existing Users** – Display a list of system users in a formatted manner.
- **Use Dialog Boxes** – Implement a user-friendly interface using dialog/whiptail/kdialog/zenityfor input and selection menus.

Features & Functionality

1. **Menu-Based Interface**
   - The script will present an interactive menu using dialog/whiptail/kdialog/zenity, allowing administrators to navigate options easily.
2. **User Input Handling**
   - Input validation ensures that usernames and passwords are correctly formatted and meet system requirements.
3. **File and Command Execution**
   - Use Linux commands such as useradd, passwd, usermod, and userdelto perform user management tasks.
4. **Logging and Error Handling**
   - Record user actions in a log file for auditing purposes.
   - Handle errors gracefully and provide feedback to the administrator.
5. **Security Considerations**
   - Ensure only privileged users (root) can execute the script.
   - Implement confirmation dialogs before executing destructive actions.

Technology & Tools

- **Operating System** Linux Operating System (Ubuntu 24, RHEL 9, Kali Linux 2024, Fedora)
- **Shell Scripting** (bash)
- **Linux User Management Commands** (useradd, passwd, usermod, userdel, cat /etc/passwd)
- **Dialog Box Tools** (dialog/whiptail/kdialog/zenity)
- **Text File Logging** (/home/username/user_management.log)

Expected Outcome

By the end of the project, the script will provide an intuitive and efficient way to manage users in a Linux environment while demonstrating practical shell scripting concepts. This will serve as a foundational tool for administrators and an educational exercise in Linux scripting. The below figure 1 shows the expected outcome of the project.

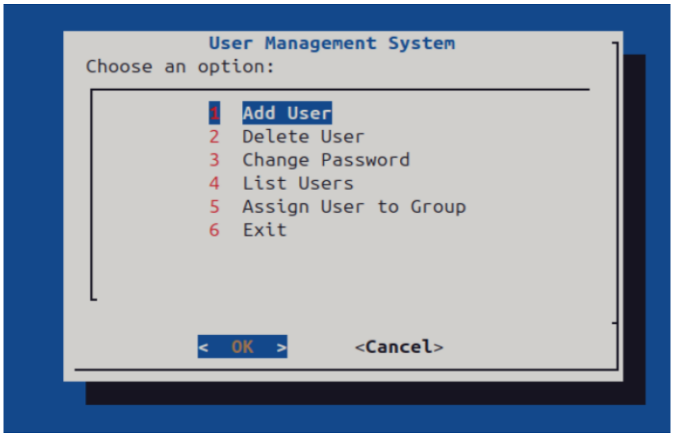


Fig. 1. User Management console

Annexure B2- Rubrics for Project based evaluation

Component	**Criteria**	**Points**
Logic Building (30%)	Implementation of Linux Commands	10
	Error handling	10
	Input validation	10
Clean Code (10%)	Code organization and structure	3
	Naming conventions and readability	3
	Comments and documentation	4
Dialog Implementation (30%)	Appropriate selection of Dialog widgets	10
	Logical correctness of Dialog usage	10
	Correct implementation of dialog widgets	10
Script Operations (30%)	Completeness of operations	15
	Data consistency	5
	Directory/File/User/process implementation	10
Total		**100**

Annexure-C

Object Oriented Software Engineering (22CS017):

Please find below the rubrics and report format for the Object Oriented Software Engineering (22CS017) and Data Structures (23CS004) project as per syllabus.

1. Guidelines for the Project -Annexure C1
2. Rubrics for Project based evaluation -Annexure C2
3. Report Format -Annexure D

Note:

- For Project Evaluation, students (in teams) are required to analyze the project details using the concepts of software engineering and create a document (the outline is given below in Annexure C1) for the project they are working on Data Structure and Programming in JAVA.
- Students have to submit a soft copy of the report (refer to Annexure D), one from each group.

Annexure C1- Guidelines for the Project

The documents should contain the following points (for example, project title is flight management system):

1. Introduction

A Flight Management System (FMS) project aims to develop a comprehensive software solution for managing all aspects of flight operations, from scheduling and reservations to crew management and flight tracking, enhancing efficiency and safety.

1.1 Purpose

The purpose of this document is to build an online system to manage flights and passengers to ease flight management. *<<Include the purpose as applicable to your project >>*

1.2 Document Conventions

This document uses the following conventions. *<<Include the conventions as per your application >>*

DB	Database
DDB	Distributed Database
ER	Entity Relationship

1.3 Intended Audience and Reading Suggestions

This project is a prototype for the flight management system, and it is restricted to the college premises. This has been implemented under the guidance of college professors. This project is useful for the flight management team and as well as to the passengers.

1.4 Project Scope

The purpose of the online flight management system is to ease flight management and to create a convenient and easy-to-use application for passengers, trying to buy airline tickets. The system is based on a relational database with its flight management and reservation functions. We will have a database server supporting hundreds of major cities around the world as well as thousands of flights by various airline companies. Above all, we hope to provide comfortable user experience along with the best pricing available.

1.5 References

- Fundamentals of database systems by ramez elmarsi and shamkant b.navathe

2. Overall Description

2.1 Product Perspective

A distributed airline database system stores the following information.

- **Flight details:** It includes the originating flight terminal and destination terminal, along with the stops in between, the number of seats booked/available seats between two destinations etc.
- **Customer description:** It includes customer code, name, address, and phone number. This information may be used for keeping the records of the customer for any emergency or for any other kind of information.
- **Reservation description:** It includes customer details, code number, flight number, date of booking, date of travel.

2.2 Product Features

The major features of the airline database system as shown in below **entity–relationship model** (**ER model**)

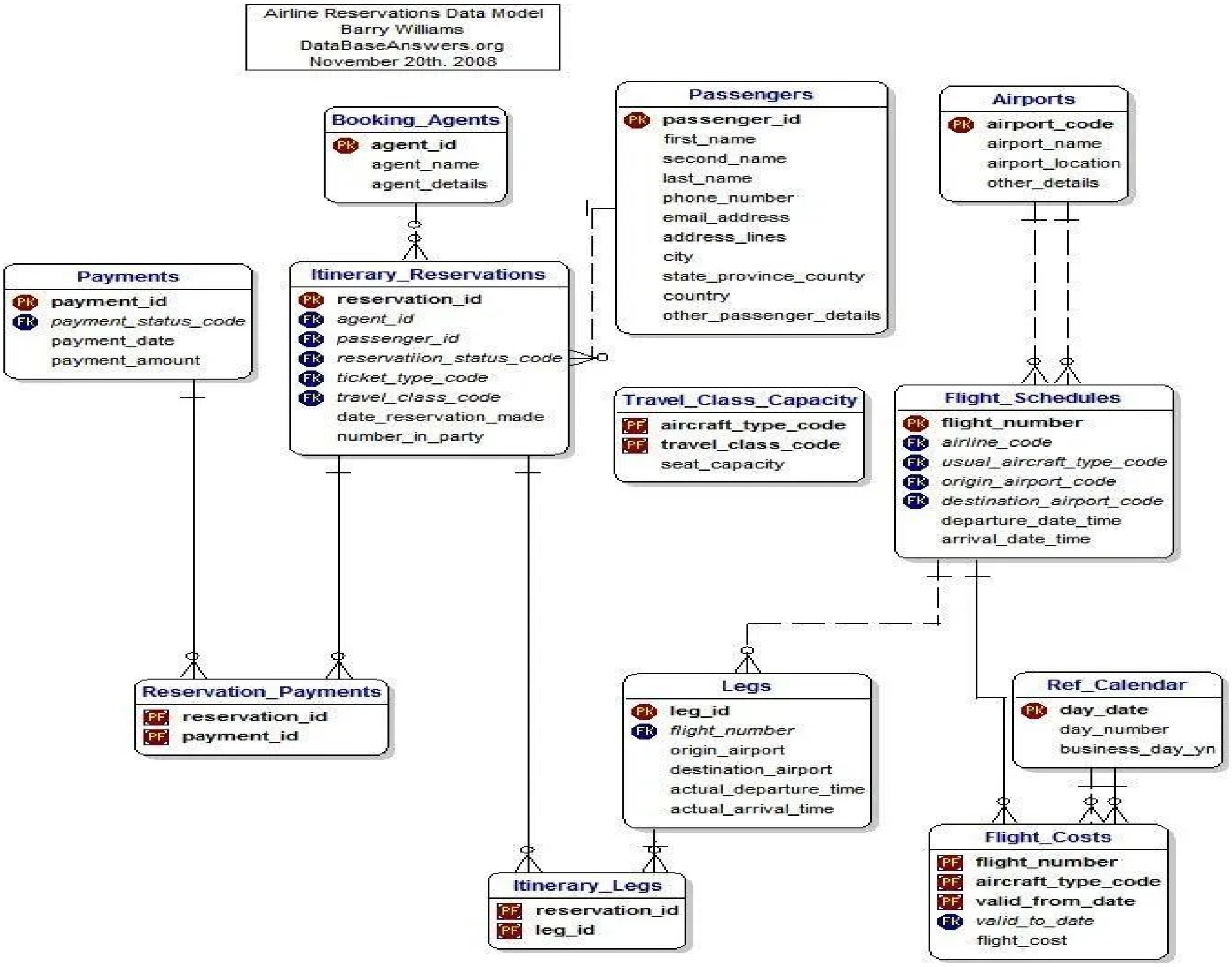


Figure 1: The diagram shows the layout of airline database system – entity–relationship model

2.3 User Class and Characteristics

Users of the system should be able to retrieve flight information between two given cities with the given date/time of travel from the database. A route from city A to city B is a sequence of connecting flights from A to B such as: a) there are at most two connecting stops, excluding the starting city and destination city of the trip, b) the connecting time is between one to two hours. The system will support two types of user privileges, Customer, and Employee. Customers will have access to customer functions, and the employees will have access to both customer and flight management functions. The customer should be able to do the following functions:

- Make a new reservation
  - One-way
  - Round-Trip
  - Multi-city
  - Flexible Date/time
  - Confirmation
- Cancel an existing reservation
- View his itinerary

The Employee should have the following management functionalities:

- CUSTOMER FUNCTIONS.
  - Get all customers who have seats reserved on a given flight.

  - • Get all flights for a given airport.
  - • View flight schedule.
  - • Get all flights whose arrival and departure times are on time/delayed.
  - • Calculate total sales for a given flight.
- ADMINISTRATIVE
  - • Add/Delete a flight
  - • Add a new airport
  - • Update fare for flights.
  - • Add a new flight leg instance.
  - • Update departure/arrival times for flight leg instances.

Each flight has a limited number of available seats. There are a number of flights that depart from or arrive at different cities on different dates and times.

2.4 Operating Environment

The operating environment for the airline management system is listed below. *<<Include the details as per your application >>*

- distributed database
- client/server system
- Operating system: Windows.
- database: sql+ database
- platform: vb.net/Java/PHP

2.5 Design and Implementation Constraints

- Use Case Diagram
- Class Diagram
- Sequence Diagram
- Activity Diagram
- Component Diagram
- State Diagram
- Deployment Diagram

3. <u>System Features</u>

- **Description and Priority**

The airline reservation system maintains information on flights, classes of seats, personal preferences, prices, and bookings. Of course, this project has a high priority because it is very difficult to travel across countries without prior reservations.

- **Stimulus/Response Sequences**
  - Search for Airline Flights for two Travel cities
  - Displays a detailed list of available flights and makes a “Reservation” or Books a ticket on a particular flight.
  - Cancel an existing Reservation.
- **Functional Requirements**

4. <u>External Interface Requirements</u>

4.1 User Interfaces

- Front-end software: Vb.net version
- Back-end software: SQL+

4.2 Hardware Interfaces

- Windows.
- A browser that supports CGI, HTML & Javascript.

4.3 Software Interfaces

The following are the software used for the flight management online application. *<<Include the software details as per your project >>*

Software used	**Description**
Operating system	We have chosen the Windows operating system for its best support and user-friendliness.
Database	To save the flight records, passenger records we have chosen SQL+ database.
VB.Net	To implement the project, we have chosen Vb.Net language for its more interactive support.

4.4 Communication Interfaces

This project supports all types of web browsers. We are using simple electronic forms for reservation forms, ticket booking etc.

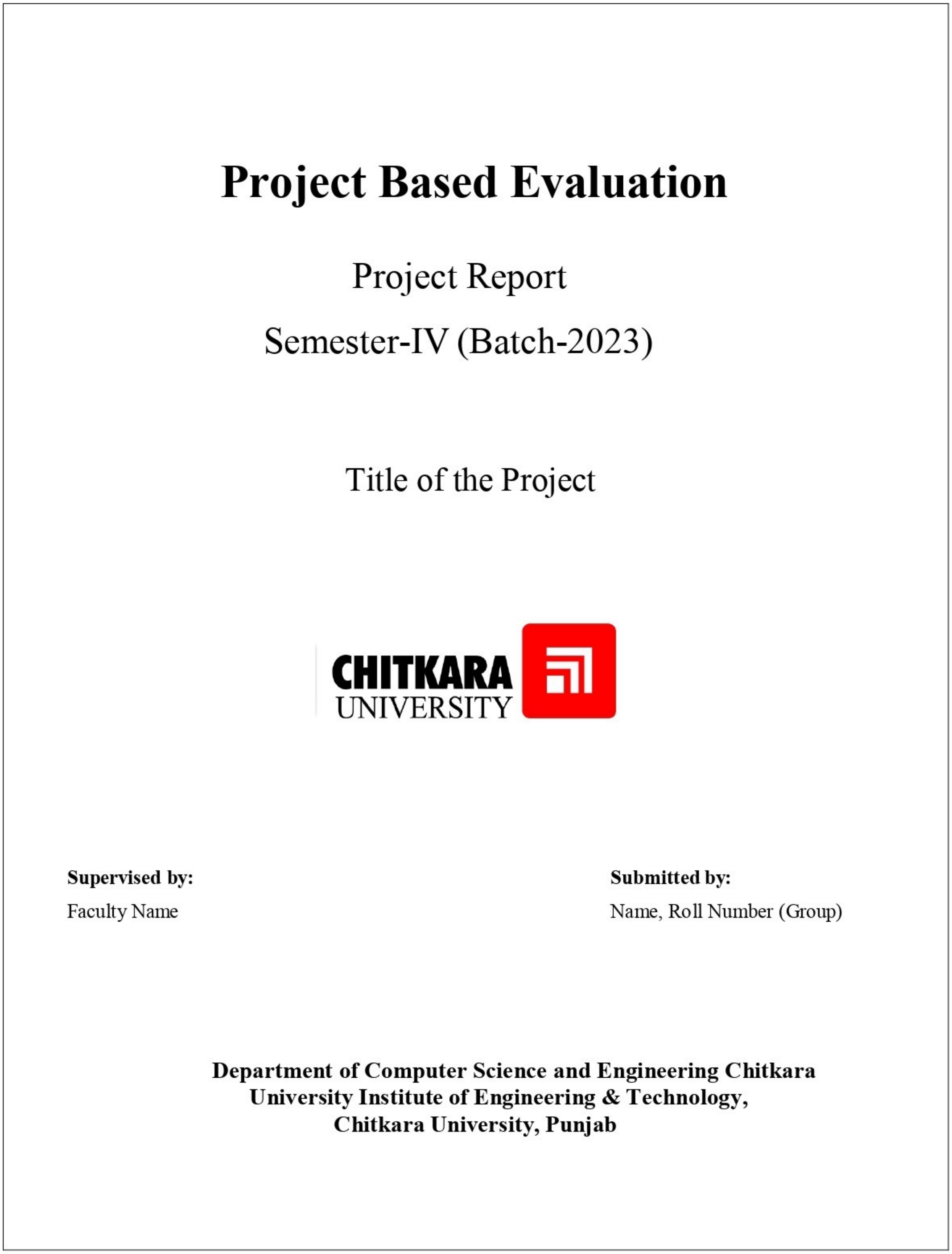
5. <u>Nonfunctional Requirements</u>

- **Performance Requirements:** How the flight is performing in different conditions.
- **Availability:** The flight should be available on the specified date and specified time as many customers are making reservations.
- **Correctness:** The flight should reach the start from the correct start terminal and should reach the correct destination.
- **Maintainability:** The web application must support regular updates to ensure compatibility with the latest web technologies, Bug fixes, and security patches.
- **Usability:** The flight schedules should satisfy the maximum number of customers’ needs. The application should be accessible on multiple devices, including desktop and mobile.

Annexure C2- Rubric for the Project based evaluation

Category	Criteria	Points	Weight (%)
Design and Architecture	- Requirements Analysis	10	50%
	- Scalability	10	
	- Use of Design Patterns	10	
	- Creativity	10	
	- System Integration	10	
Documentation	- Comprehensiveness	6	15%
	- Structure	6	
	- Usability	3	
Team Collaboration	- Contribution	6	10%
	- Conflict Resolution	4	
Testing and Validation	- Testing Coverage	2	10%
	- Test Documentation	2	
	- Automated Testing	2	
	- Validation	2	
	- Bug Reporting	2	
Presentation and Demo	- Clarity	2	10%
	- Engagement	2	
	- Key Features Highlighted	2	
	- Professionalism	2	
	- Q&A Session	2	
Innovation and Creativity	- Originality	1	5%
	- Impact	1	
	- Novel Features	1	
	- User Experience	1	
	- Future Potential	1	
Total		100	100%

Annexure D- Report Format



Project Based Evaluation

Project Report

Semester-IV (Batch-2023)

Title of the Project

CHITKARA
UNIVERSITY

Supervised by:
Faculty Name

Submitted by:
Name, Roll Number (Group)

Department of Computer Science and Engineering Chitkara University Institute of Engineering & Technology, Chitkara University, Punjab

Guidelines for the Project Report

Project Report Format

The order in which the content of the project report should be organized is as follows:

Title Page (Format attached)

Abstract

Table of Contents

1. **Introduction**- should include Background, objectives and significance, etc. (3-5 pages)
2. **Problem Definition and Requirements**- Problem statement and software requirements/ hardware requirements/data sets (1-2 pages)
3. **Proposed Design / Methodology** – student may include schematic diagram/ file structure/ (3-5 pages)
4. **Results**- screenshots etc. (10-15 pages)

References (if any, style-API)

Note:

Font: Times New Roman

Size:

- Heading-16 (Bold)-Align left
- Sub heading-14 (Bold)-Align-left
- Paragraph -12 (normal)-Justify

Line Spacing- 1.5

Table/Figures- Number and label each table and figure in order.

Caption for Table/Figure- Times New Roman (12); normal; align-center.

Layout for Sections (Example)-

1. **Introduction**

   1.1 Background

   1.2 Objectives

2. **Problem Definition and Requirements**