

Title: Fundamentals of Database Systems (FDS) Final Project Outline

Objective: The objective of this project is to apply the concepts learned in the subject Fundamentals of Database Systems (FDS) by developing a web-based application that incorporates proper database design, SQL queries, and CRUD (Create, Read, Update, Delete) functionalities.

Yr/Section: BSIT 1A

Deadline: May 19, 2025

Google Sheet Link : <https://docs.google.com/spreadsheets/d/1NrW9EhS3-UiXHz8SOESVAEtD77s77zu735DEJRjdqNE/edit?usp=sharing>

General Requirements:

- 1. Choose a Topic. (Deadline: April 11, 2025)**
 - Students must select a topic or system based on their interest. Some examples include:
 - Inventory Management System
 - Student Records System
 - Appointment Booking System
 - Library Management System
 - Event Tracking System
- 2. Create a Database Design. (Deadline: April 22, 2025)**
 - Create an Entity Relationship Diagram (ERD) or relational schema.
 - Include proper relationships (1:1, 1:M, M:N where applicable).
- 3. Develop a System with CRUD Operations**
 - Students may use any backend language they are comfortable with (e.g., PHP, Python, Node.js).
 - The system must have at least three interconnected tables.
- 4. Implement SQL Queries**
 - Use SQL queries as covered in class:
 - SELECT, INSERT, UPDATE, DELETE
 - JOIN, GROUP BY, ORDER BY
 - Subqueries and aggregate functions (e.g., COUNT, SUM)
- 5. Documentation & Report**
 - System overview
 - Database schema with explanation
 - Screenshots of each functionality
 - Sample SQL queries used
 - Reflection on learnings and experience

Deliverables:

- Complete system/website source code
- Database file (.sql or database export)
- Documentation (printed)
- Final project presentation/demo (recorded)

Rubric for Project Evaluation

Criteria	Weight	Excellent (4)	Good (3)	Satisfactory (2)	Needs Improvement (1)
Database Design (ERD, Normalization, Schema Design)	30%	Clearly defined ERD, well-normalized, with efficient and scalable schema design.	Mostly clear ERD, mostly normalized, minor issues in schema design.	ERD present but lacks clarity, partial normalization, some design flaws.	Poor or missing ERD, no normalization, inefficient schema.
SQL Query Implementation (JOIN, GROUP BY, correctness)	30%	All queries are correct, efficient, and use advanced SQL (JOIN, GROUP BY, etc.) properly.	Queries are mostly correct and functional with appropriate SQL usage.	Queries work but have minor errors or limited SQL features.	Queries are incorrect, inefficient, or missing key SQL elements.
CRUD Functionality	20%	Full CRUD implemented accurately and efficiently with validations.	CRUD is mostly implemented with minor bugs or limitations.	Partial CRUD with basic functionality, some errors.	CRUD features incomplete or non-functional.
User Interface and Usability	5%	Intuitive, user-friendly, and visually appealing interface.	Usable interface with some areas for improvement.	Basic interface, somewhat usable but not intuitive.	Poor or confusing interface, difficult to use.
Documentation	5%	Complete, well-written, and includes all necessary diagrams and instructions.	Good documentation with most required content.	Some documentation present but lacks detail or clarity.	Incomplete or missing documentation.
Presentation and Demo	10%	Clear, confident presentation; all features demonstrated effectively.	Good presentation with most features covered.	Basic presentation, some features not shown clearly.	Weak presentation, unclear demo or missing parts.