Project Proposal

Project Name: LibMaster: Automated Library Management System
Group Name: Nothing
Student Names: Md. Arif Shekh(23103022), Laxmi sarker(23103189),
Taohid Hassan(23103103)
Date: 22/11/2025

1. Project Overview

This project aims to develop a desktop-based **Library Management System** to automate the manual processes of a library. The system will streamline the tracking of book inventory, member management, and the issuing/returning of books, reducing human error and increasing efficiency for librarians.

2. Key Features

The application will be divided into two modules: **Administrator** and **General User (Student)**.

- **Book Management (CRUD):** The admin can add new books, update book details and delete obsolete records.
- **Member Registration:** Ability to register new students/members and assign them unique Library IDs.
- **Issue & Return System:**
  - **Issue:** Check out books to a member, recording the issue date and due date.
  - **Return:** Process book returns and update inventory status automatically.
- **Search Functionality:** A robust search bar allowing users to find books by Title, Author, or Category.
- **Dashboard/Reporting:** A visual display showing total books, books currently issued, and pending returns.

3. Tools & Technologies

To achieve a robust and user-friendly application, the following technology stack will be used:

- **Programming Language:** C# (C-Sharp)
- **IDE:** Microsoft Visual Studio 2022
- **Framework:** .NET Framework (Windows Forms Application for GUI)
- **Database:** Microsoft SQL Server (LocalDB or Express)
- **Database Connectivity:** ADO.NET or Entity Framework
- **Design Pattern:** Object-Oriented Programming (OOP) principles.

4. How to Implement

The development process will follow the **Waterfall Model** to ensure structured progress:

1. **Database Design:**
   - Create a relational database schema.
   - Design tables for `Books`, `Members`, `Staff`, and `IssueRecords`.
   - Establish primary and foreign keys to link members to issued books.
     Opens in a new windowGetty ImagesExplore
     Database development
2. **User Interface (UI) Design:**
   - Use **Windows Forms (WinForms)** to drag and drop controls (Buttons, TextBoxes, DataGridViews).
   - Design a clean Login Form, Dashboard, and Data Entry forms.
3. **Backend Logic (C# Coding):**
   - Create C# classes (e.g., `Book.cs`, `Member.cs`) to represent data objects.
   - Implement logic for date calculations (for due dates and fines).
   - Write SQL queries (INSERT, SELECT, UPDATE, DELETE) and execute them using C# `SqlCommand` objects.
4. **Testing:**
   - Perform unit testing on the "Issue Book" function to ensure inventory decreases by 1 when a book is issued and increases by 1 when returned.