Annexure 1

Project Report

On

**Student Record Management System (SQLite)**

Submitted

In Partial Fulfilment of

**BACHELOR OF COMPUTER APPLICATIONS (BCA)**

**Submitted by:**

Name: Ayush Das

Roll No: 24/SCA/BCA(AI&ML)/012

**Under the Supervision of:**

Ms shruti Gupta

Assistant Professor, SCA



**School of Computer Applications**

**Manav Rachna International Institute of Research and Studies**

**(DEEMED TO BE UNIVERSITY)**

Sector-43, Aravalli Hills Faridabad

– 121001

**June 2025**

Annexure 2

**Declaration**

I so certify that my project, "Student Record Management System (SQLite)," which I turned in to partially meet the requirements for the BACHELOR OF COMPUTER APPLICATIONS degree, is an original work of mine. The report, which has never before been presented for the award of a degree or certificate to any institution or university, contains the conclusions drawn from my research and observations.

SIGNATURE

Name: Ayush Das

Roll No:24/SCA/BCA(AI&ML)/012

Date: 21.7.25

Annexure 3

**Certificate from the Guide**

This certifies that Mr. Ayush Das (24/SCA/BCA(AI&ML)/012) completed the project report titled "Student Record Management System (SQLite)" under my supervision in order to partially fulfil the requirements for the degree of **BACHELOR OF COMPUTER APPLICATIONS** at Manav Rachna International Institute of Research and Studies, Faridabad.

Signature of the Guide

Name: Ms. Shruti Gupta

Date: 21.7.25

**Head of Department**

**Prof. Dr. Suhail Javed Quraishi**

**ACKNOWLEDGEMENT**

I sincerely appreciate your help, collaboration, direction, and clarity during the creation of the Student Record Management System (SQLite) . I am really appreciative of Dr. Angela Yu , who helped me with the assignment. This project would not have been able to be finished on schedule without her cooperative attitude, spirit of flexibility, Candor, prompt clarification. Special consideration should be given to her willingness to discuss any significant work-related issue.

Name: AYUSH DAS

Roll No: 24/SCA/BCA(AI&ML)/012

Date: 21/7/25

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **Page No** | **Sign** | **Remark** |
| Introduction | 1 |  |  |
| OBJECTIVES | 1 |  |  |
| TOOLS & TECHNOLOGIES USED | 2 |  |  |
| SYSTEM DESIGN OVERVIEW | 2 |  |  |
| FUNCTIONALITIES IMPLEMENTED | 3 |  |  |
| SAMPLE CODE SNIPPET | 3 |  |  |
| OUTPUT (EXAMPLE) | 4 |  |  |
| LEARNING OUTCOMES | 4 |  |  |
| FUTURE SCOPE | 5 |  |  |
| CONCLUSION | 5 |  |  |
| Bibliography | 6 |  |  |
|  |  |  |  |

1. **INTRODUCTION**

This project titled “Student Record Management System” is a simple yet practical application built using Python and SQLite. It is designed to manage basic student records in a structured and organized way. The system allows users to perform CRUD (Create, Read, Update, Delete) operations through a command-line interface.

Python was chosen due to its simplicity and powerful built-in libraries, and SQLite was selected as the database for lightweight and local data storage.

1. **OBJECTIVES**

* Develop a basic student management application using Python.
* Perform standard database operations like add, view, search, update, and delete.
* Implement a simple CLI interface for user interaction.
* Practice using the sqlite3 module in Python for database management.
* Build a project suitable for academic submission and real-world demonstration.

1. **TOOLS & TECHNOLOGIES USED**

* Programming Language: Python 3.x
* Database: SQLite
* Module: sqlite3
* IDE/Editor: Visual Studio Code / PyCharm / IDLE
* Platform: Cross-platform (Windows/Linux/Mac)

1. **SYSTEM DESIGN OVERVIEW**

The system is a console-based application. It displays a menu with the following options:

1. Add Student
2. View All Students
3. Search Student
4. Update Student
5. Delete Student
6. Exit

The student data includes:

* Roll Number (Primary Key)
* Name
* Branch

All records are stored in a SQLite database file named student.db.

1. **FUNCTIONALITIES IMPLEMENTED**

a. Add Student

* Takes user input and inserts the new record into the SQLite database.
* Checks for duplicate roll numbers (as it's the primary key).

b. View All Students

* Displays a list of all students currently stored in the database.

c. Search Student

* Allows searching for a student by roll number and displays the record if found.

d. Update Student

* Allows updating name and branch by entering an existing roll number.

e. Delete Student

* Removes a student record from the database based on roll number.

1. **SAMPLE CODE SNIPPET**

Using sqlite3 module to insert a new student:

import sqlite3  
conn = sqlite3.connect('student.db')  
c = conn.cursor()  
c.execute("INSERT INTO students (roll, name, branch) VALUES (?, ?, ?)", (roll, name, branch))  
conn.commit()

1. **OUTPUT (EXAMPLE)**

==== Student Record System ====

1. Add Student
2. View All Students
3. Search Student
4. Update Student
5. Delete Student
6. Exit

Enter choice (1-6): 1  
Enter Roll Number: 101  
Enter Name: Anjali  
Enter Branch: CSE  
Student added successfully!

1. **LEARNING OUTCOMES**

* Gained hands-on experience in Python programming.
* Learned how to use SQLite as a backend for storing data.
* Understood the basics of CRUD operations and database management.
* Applied problem-solving and modular coding practices.

1. **FUTURE SCOPE**

* Add GUI using Tkinter for a better user interface.
* Upgrade to a web-based version using Flask or Django.
* Add validation, error handling, and export features (CSV/PDF reports).
* Implement login authentication for admin access.

1. **CONCLUSION**

This mini project successfully demonstrates how Python can be used to develop simple database-driven applications. It is a practical implementation of CRUD operations using SQLite and showcases a real-world use case of managing records in a lightweight and efficient manner.

1. **BIBLIOGRAPHY**

The following resources were referenced and consulted during the development of this project:

1. GeeksforGeeks Python Tutorials  
   <https://www.geeksforgeeks.org/python-programming-language>
2. Python Official Documentation – sqlite3  
   <https://docs.python.org/3/library/sqlite3.html>
3. W3Schools – Python SQLite Tutorial  
   <https://www.w3schools.com/sql/sql_python.asp>
4. Stack Overflow – Community Discussions and Solutions  
   <https://stackoverflow.com/>
5. TutorialsPoint – Python Database Access  
   <https://www.tutorialspoint.com/sqlite/sqlite_python.htm>
6. Real Python – Working with SQLite in Python  
   <https://realpython.com/python-sql-libraries/>
7. SQLite Official Documentation  
   <https://www.sqlite.org/docs.html>

These sources provided guidance on Python syntax, database operations, and programming best practices.