

ABSTRACT:

This project endeavours to revolutionize the management of student information within educational institutions by developing a sophisticated web-based application utilizing the CodeIgniter framework. The primary objective is to create a highly efficient system capable of handling student details seamlessly through a user-friendly interface. Emphasizing the implementation of CRUD operations (Create, Read, Update, Delete), this project aims to establish a secure, scalable, and intuitive platform for storing and managing comprehensive student records.

This project delves into the domain of educational information systems, aspiring to redefine the paradigms of managing student data through the development of a sophisticated web-based application. The project centres around harnessing the capabilities of the CodeIgniter framework to create an innovative solution tailored to the dynamic needs of educational institutions. The primary ambition is to construct an all-encompassing platform adept at efficiently handling student details while ensuring data integrity, security, and user-friendliness.

In today's educational landscape, the efficient management of student information plays a pivotal role in shaping administrative efficiency and academic progress. The prevailing conventional methods reliant on manual record-keeping, disparate spreadsheets, and fragmented databases pose significant challenges. These include data redundancy, integrity issues, limited accessibility, and a lack of agility to adapt to evolving educational requirements. Hence, this project sets out to engineer a revolutionary solution that streamlines the management of student details.

INDEX

Contents

INTRODUCTION:	3
For Edit	3
For Filling :	4
Objectives:	4
Methodology:	4
Technology Stack:	4
Implementation Steps:	5
1.Database Design:	5
2.CodeIgniter Setup:	5
3.Model-View-Controller (MVC) Structure:	5
4.CRUD Operations:	5
5.User Interface Design:	5
My Database	6
Results and Discussion:	8
Conclusion:	8
REFERENCES:	8
GITHUB REPOSITORY LINK:	8

INTRODUCTION:

The efficient management and organization of student information serve as foundational pillars in facilitating seamless administrative operations and fostering academic excellence. However, the prevalent reliance on archaic, paper-based record-keeping methods and disjointed digital spreadsheets presents formidable challenges. These challenges encompass data redundancy, inconsistent data integrity, limited accessibility, and an inability to swiftly adapt to the dynamic and evolving needs of educational institutions. Recognizing these limitations, this project embarks on a transformative journey towards architecting an innovative solution poised to revolutionize the management of student details within educational ecosystems.

[For Edit](#)

Studnet Registration Form

Course Code

CIVIL401

Branch Code

CE

Semester

5

Scheme Code

SC004

Leet/Non-Leet Status

Non-Leet

Six month training

No

Subject Code

CE401

For Filling :

Form fields for filling student information:

- Six month training
- Subject Code
- M-Code
- Theory/Practical
- Elective status
- Internal max marks
- External max marks
- Credit
- Submit button

Objectives:

- Create a database schema to store student information.
- Implement CRUD functionality to manage student records.
- Design a user-friendly interface for administrators to interact with student data.
- Ensure data security and validation for user inputs.

Methodology:

Technology Stack:

- CodeIgniter Framework: Chosen for its MVC architecture, simplicity, and scalability.
- PHP: Server-side scripting language for backend functionality.
- MySQL: Database management system for storing student information.
- HTML, CSS: Frontend development for a user-friendly interface.

Implementation Steps:

1.Database Design:

- Created a MySQL database with tables to store student details (e.g., ID, name, contact info, courses enrolled, etc.).
- Ensured normalization and defined relationships among tables for efficient data retrieval.

2.CodeIgniter Setup:

- Installed and configured CodeIgniter on the server.
- Set up the necessary configuration files, including database configuration.

3.Model-View-Controller (MVC) Structure:

- Implemented the MVC architecture for organized development.
- Models handle interactions with the database.
- Views represent the user interface.
- Controllers manage the application flow.

4.CRUD Operations:

- Created functions/methods within controllers and models for CRUD operations.
- Developed forms and interfaces for administrators to add, view, edit, and delete student records.

5.User Interface Design:

- Designed a user-friendly interface using HTML, CSS.
- Implemented validation for user inputs to ensure data integrity.

My Database

Showing rows 0 - 7 (8 total, Query took 0.0007 seconds)

SELECT * FROM `students`

Filter rows: Search this table: Sort by key: Name

Extra options

	student_name	course_id	branch_code	semester	scheme_code	test	training	subject_code	mode	theory	elective	int_marks	ext_marks	credit	created_at
<input type="checkbox"/>	1	101	BC001	Spring	SC001	Yes	No	SB001	NC001	Theory A	Elective A	75	85	4	2023-12-18 11:18:33
<input type="checkbox"/>	2	102	BC002	Fall	SC002	No	Yes	SB002	NC002	Theory B	Elective B	80	90	4	2023-12-18 11:18:33
<input type="checkbox"/>	3	103	BC003	Summer	SC003	Yes	Yes	SB003	NC003	Theory C	Elective C	85	95	5	2023-12-18 11:18:33
<input type="checkbox"/>	4	104	BC001	Winter	SC004	No	No	SB004	NC004	Theory D	Elective D	70	80	3	2023-12-18 11:18:33
<input type="checkbox"/>	5	105	BC002	Spring	SC005	Yes	No	SB005	NC005	Theory E	Elective E	78	88	4	2023-12-18 11:18:33
<input type="checkbox"/>	6	106	BC003	Fall	SC006	No	Yes	SB006	NC006	Theory F	Elective F	82	92	5	2023-12-18 11:18:33
<input type="checkbox"/>	7	107	BC001	Summer	SC007	Yes	Yes	SB007	NC007	Theory G	Elective G	88	98	4	2023-12-18 11:18:33

Query results operations: Print, Copy to clipboard, Export, Display chart, Create view

Showing rows 0 - 9 (10 total, Query took 0.0005 seconds)

SELECT * FROM `students`

Filter rows: Search this table: Sort by key: Name

Extra options

	student_name	course_id	branch_code	semester	scheme_code	test	training	subject_code	mode	theory	elective	int_marks	ext_marks	credit	created_at	updated_at	deleted_at
<input type="checkbox"/>	1	101	BC001	Spring	SC001	Yes	No	SB001	NC001	Theory A	Elective A	75	85	4	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	2	102	BC002	Fall	SC002	No	Yes	SB002	NC002	Theory B	Elective B	80	90	4	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	3	103	BC003	Summer	SC003	Yes	Yes	SB003	NC003	Theory C	Elective C	85	95	5	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	4	104	BC001	Winter	SC004	No	No	SB004	NC004	Theory D	Elective D	70	80	3	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	5	105	BC002	Spring	SC005	Yes	No	SB005	NC005	Theory E	Elective E	78	88	4	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	6	106	BC003	Fall	SC006	No	Yes	SB006	NC006	Theory F	Elective F	82	92	5	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	7	107	BC001	Summer	SC007	Yes	Yes	SB007	NC007	Theory G	Elective G	88	98	4	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	8	108	BC002	Winter	SC008	No	No	SB008	NC008	Theory H	Elective H	76	86	3	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	9	109	BC003	Spring	SC009	Yes	No	SB009	NC009	Theory I	Elective I	79	89	4	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A
<input type="checkbox"/>	10	110	BC001	Fall	SC010	No	Yes	SB010	NC010	Theory J	Elective J	83	93	5	2023-12-18 11:18:33	2023-12-18 11:18:33	N/A

Query results operations: Print, Copy to clipboard, Export, Display chart, Create view

OUR VIEW

Student URN	Course ID	Branch Code	Semester	Scheme Code	Leet	Training	Subject Code	M Code	Theory	Elective	Int Max Marks	Ext Max Marks	Credit	Created at	Action	
12345	CSE101	CS	3	SC001	Leet	Yes	CS301	M123	Theory	Yes	80	90	4	2023-01-01 12:00:00	Delete	Edit
23456	ECE201	EC	2	SC002	Non-Leet	No	EC201	M456	Practical	No	75	85	3	2023-01-02 14:30:00	Delete	Edit
34567	MECH301	ME	4	SC003	Leet	Yes	ME301	M789	Theory	Yes	85	95	4	2023-01-03 10:45:00	Delete	Edit
45678	CIVIL401	CE	5	SC004	Non-Leet	No	CE401	M101	Practical	No	78	88	3	2023-01-04 16:15:00	Delete	Edit
56789	ITS01	IT	6	SC005	Leet	Yes	ITS01	M202	Theory	Yes	88	98	4	2023-01-05 08:00:00	Delete	Edit

Delete Operation Done

Student URN	Course ID	Branch Code	Semester	Scheme Code	Leet	Training	Subject Code	M Code	Theory	Elective	Int Max Marks	Ext Max Marks	Credit	Created at	Action	
12345	CSE101	CS	3	SC001	Leet	Yes	CS301	M123	Theory	Yes	80	90	4	2023-01-01 12:00:00	Delete	Edit
23456	ECE201	EC	2	SC002	Non-Leet	No	EC201	M456	Practical	No	75	85	3	2023-01-02 14:30:00	Delete	Edit
34567	MECH301	ME	4	SC003	Leet	Yes	ME301	M789	Theory	Yes	85	95	4	2023-01-03 10:45:00	Delete	Edit
45678	CIVIL401	CE	5	SC004	Non-Leet	No	CE401	M101	Practical	No	78	88	3	2023-01-04 16:15:00	Delete	Edit

After Editing We have changed Subject code

Student URN	Course ID	Branch Code	Semester	Scheme Code	Leet	Training	Subject Code	M Code	Theory	Elective	Int Max Marks	Ext Max Marks	Credit	Created at	Action	
12345	CSE101	CS	3	SC001	Leet	Yes	CS301	M123	Theory	Yes	80	90	4	2023-01-01 12:00:00	Delete	Edit
23456	ECE201	EC	2	SC002	Non-Leet	No	EC201	M456	Practical	No	75	85	3	2023-01-02 14:30:00	Delete	Edit
34567	MECH301	ME	4	SC003	Leet	Yes	ME301	M789	Theory	Yes	85	95	4	2023-01-03 10:45:00	Delete	Edit
45678	CIVIL401	CE	5	SC004	Non-Leet	No	CES00	M101	Practical	No	78	88	3	2023-01-04 16:15:00	Delete	Edit

Results and Discussion:

The developed application successfully fulfils the objectives outlined in the project. Key outcomes include:

- **Functional CRUD Operations:** Administrators can easily create, read, update, and delete student records through an intuitive web interface.
- **Database Management:** The database schema efficiently stores and manages student details, ensuring data integrity and scalability.
- **User-Friendly Interface:** The application provides a clean and responsive user interface for seamless interaction.

Conclusion:

The project demonstrates the successful implementation of a student management system using CodeIgniter, enabling efficient handling of student details. The application provides an intuitive interface for administrators to manage student records securely. Further enhancements could involve additional features such as user authentication, advanced search functionalities, and report generation capabilities.

REFERENCES:

- **CodeIgniter Documentation:** https://codeigniter.com/user_guide/
- **Bootstrap Documentation:** <https://getbootstrap.com/>

GITHUB REPOSITORY LINK:

Repository: https://github.com/rajat-28/AWT_LAB