	PROJECT REPORT Code: .--. .-. --- -.-- . -.-. - --- / ..--- ----- .---- ..--- -...- .---- ..--- -...- .---- ..--- Version: 0001 Author: ********** Adviser: ********** Entity: **** Country: ***** City: **** date: ********

Project Protection and Usage License

This report and the project described within are intellectual property of the company GR. Its use, distribution, and modification are allowed exclusively under the terms of this license, which is focused on ensuring that any improvements or edits to the project are made solely for the purpose of optimizing its functionality and educational value.

Terms of Use

1. **Reproduction and Distribution:** Reproduction and distribution of this project are permitted as long as the integrity of the original content is maintained and proper credit is given to the company GR as the creator.
2. **Allowed Modifications:** Modifications and adaptations to the project are allowed only when intended to improve its design, functionality, or implementation. Any changes must be documented and shared with GR for evaluation and potential integration into future versions.
3. **Commercial Use:** This project may not be used for commercial purposes without prior written authorization from GR.

User Responsibilities

Users accessing the content of this report and modifying the project must: • Ensure that any improvements respect the original development principles. • Document the changes made clearly and in detail. • Notify GR of any proposed improvements, in order to maintain a centralized record of contributions to the project.

Authorization for Commercial Use

Although the project is freely accessible for educational and improvement purposes, any commercial use must be authorized in advance by GR. Authorization requests will be evaluated to ensure compliance with the company's ethical and legal principles.

Charges and Penalties

Improper use of this project, including unauthorized commercial exploitation or modifications that negatively affect its original purpose, may result in legal actions. These actions may include: • Fines proportional to the harm caused. • Legal proceedings for breach of license terms.

Intellectual Property Statement

The company GR retains ownership of this project and its associated rights. This licensing framework aims to foster collaboration and continuous improvement, always under an ethical and responsible scheme.

Contact

For notifications, improvement suggestions, or authorization requests, users can contact GR through the following channels:

• Email: **********

• Phone: **********

• Address: **********

Table of Contents

Summary

The university management system developed as a personal project represents an effort to design a comprehensive solution that optimizes academic and administrative processes in educational institutions. This project, still in the development stage and without an official launch, focuses on providing a tool that centralizes the management of data related to students, teachers, and courses. With specific functionalities for each user role, the system not only improves operational efficiency but also ensures the security and accessibility of information. Based on technologies such as NetBeans and MySQL, the system uses intuitive forms that connect directly to a robust database, allowing for the automation of key tasks such as enrollment, information retrieval, and course assignment. Although currently in a testing phase, this personal project has the potential to become a valuable solution for educational institutions seeking to modernize and digitize their processes.

Introduction

In an increasingly digital world, the automation and centralization of administrative and academic processes are priority challenges for educational institutions. This personal project, focused on the development of a university management system, arises as a proposal to address these needs, offering a solution that integrates all aspects of academic administration into a single, efficient platform.

The system is designed to serve two main user profiles: administrators and students. Administrators can manage key data such as enrolling new students, updating teacher information, and assigning courses, while students can easily and securely view and complete their enrollments. The implementation of an authentication system based on national identification numbers (DNI) and passwords ensures the protection of user data.

Although this is a personal project still in the development stage, it reflects a significant effort to understand and address the challenges that educational institutions face today. Once completed, it is expected that this system can be adapted for implementation in real-world scenarios, marking an advancement in the way academic processes are managed.

Justification

This personal project arises from the need to explore innovative solutions for academic management in educational institutions. Although it has not yet been officially launched or implemented in a real educational setting, the developed system's main objective is to demonstrate the feasibility of a platform that centralizes and automates essential processes, such as managing students, teachers, courses, and enrollments. In contexts where information is handled manually or through outdated systems, this project aims to offer a modern and efficient alternative that can adapt to the needs of institutions.

The primary justification for this work lies in its ability to simplify administrative tasks, reduce human errors, and improve the user experience. As a personal project, it also reflects the learning and mastery of technological tools such as NetBeans and MySQL, with a focus on data security and accessibility. Although it is not in official use, its modular and adaptable design suggests a future where this system could be implemented in educational institutions, transforming the way they manage their internal processes and contributing to technological advancement in the education sector.

Objectives

General Objective

Develop a university management system as a personal project, with the purpose of centralizing and automating administrative and academic processes, including the management of students, teachers, courses, and enrollments, using modern and secure technologies such as NetBeans and MySQL.

Specific Objectives

1. Design a platform with differentiated roles (administrator and student) that facilitates efficient access and use of the system according to the needs of each user type.
2. Implement features that allow administrators to register, consult, and update key information about students, teachers, and courses, improving the organization and accuracy of the data.
3. Develop an authentication module based on DNI and passwords, ensuring secure and personalized access to the system.
4. Provide students with intuitive tools to consult their enrollment and register for courses, optimizing their interaction with the platform.
5. Validate the functionality and robustness of the system through controlled testing, ensuring it meets basic quality standards and is adaptable for future implementation in educational institutions.
6. Demonstrate the potential of the project as a modern solution for academic management needs, highlighting its benefits for institutions seeking to digitalize and improve their processes.

Thematic Development

General Details of the System

The system for the Sideral Carrion University is focused on five main areas: students, teachers, courses, enrollments, and users. Each module is supported by a specific table in the database. The main functionalities include:

- **Student Management** (add, view).
- **Teacher Management** (add, view).
- **Course Management** (add, view).
- **Assignment of Teachers to Courses**.
- **Student Enrollment in Courses**.
- **Secure Login with Roles** (administrator/student).

Explanation of Each Database Table

Table: Students (alumnos)

This table is the core of student management within the system. It represents the students registered at the university and stores all the necessary information to identify them and associate them with enrollments. It allows for operations related to students, such as adding new records and consulting existing data. Additionally, it ensures that students' personal and academic data are organized and available for the enrollment process.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id_alumno	int(11)			No	*Ninguna*		AUTO_INCREMENT
2	DNI	varchar(20)	*utf8mb4_general_ci*		No	*Ninguna*		
3	Nombres	varchar(100)	*utf8mb4_general_ci*		No	*Ninguna*		
4	Apellido_Paterno	varchar(100)	*utf8mb4_general_ci*		No	*Ninguna*		
5	Apellido_Materno	varchar(100)	*utf8mb4_general_ci*		No	*Ninguna*		
6	Departamento	varchar(100)	*utf8mb4_general_ci*		Sí	*NULL*		
7	Provincia	varchar(100)	*utf8mb4_general_ci*		Sí	*NULL*		
8	Direccion	varchar(255)	*utf8mb4_general_ci*		Sí	*NULL*		
9	Correo	varchar(100)	*utf8mb4_general_ci*		Sí	*NULL*		
10	Sexo	enum('M', 'F')	*utf8mb4_general_ci*		No	*Ninguna*		

Table: Teachers (docentes)

This table is the foundation of faculty management. It registers the professors working at the university, allowing courses to be assigned to them. This table facilitates the tracking of teachers and their academic responsibilities, ensuring that the information is available for consultation and modification at any time.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id_docente	int(11)			No	*Ninguna*		AUTO_INCREMENT
2	DNI	varchar(20)	*utf8mb4_general_ci*		No	*Ninguna*		
3	Nombres	varchar(100)	*utf8mb4_general_ci*		No	*Ninguna*		
4	Apellidos	varchar(200)	*utf8mb4_general_ci*		No	*Ninguna*		
5	Profesion	varchar(100)	*utf8mb4_general_ci*		No	*Ninguna*		
6	Departamento	varchar(100)	*utf8mb4_general_ci*		Sí	*NULL*		
7	Provincia	varchar(100)	*utf8mb4_general_ci*		Sí	*NULL*		
8	Correo	varchar(100)	*utf8mb4_general_ci*		Sí	*NULL*		
9	Sexo	enum('M', 'F')	*utf8mb4_general_ci*		No	*Ninguna*		

Table: Courses (cursos)

This table is crucial for academic planning, as it contains the courses offered by the university. It is used by both administrators, to manage the academic offerings, and by students, who can view the list of available courses for enrollment. Through this table, the details of each course are organized, such as its name and duration.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id_curso	int(11)			No	*Ninguna*		AUTO_INCREMENT
2	Codigo	varchar(20)	*utf8mb4_general_ci*		No	*Ninguna*		
3	nombre_curso	varchar(255)	*utf8mb4_general_ci*		No	*Ninguna*		
4	Duracion	int(11)			No	*Ninguna*		

Table: Teaching Load (carga_lectiva)

This table acts as a bridge between teachers and courses. Its main purpose is to assign a specific course to a teacher, reflecting the teachers' workload. In this way, it is possible to clearly identify which teacher is in charge of each course, ensuring an efficient allocation of human resources.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id_carga	int(11)			No	*Ninguna*		AUTO_INCREMENT
2	id_Docente	int(11)			No	*Ninguna*		
3	id_Curso	int(11)			No	*Ninguna*		

Table: Enrollments (matriculas)

This table is the heart of the student registration process. It records the course assignments made by students during the enrollment process. Additionally, it allows for queries to verify which courses a student has taken, as well as the data associated with their enrollment, which is essential for academic tracking.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id_matricula	int(11)			No	*Ninguna*		AUTO_INCREMENT
2	id_alumno	int(11)			No	*Ninguna*		
3	id_curso	int(11)			No	*Ninguna*		
4	fecha_matricula	timestamp			No	current_timestamp()		

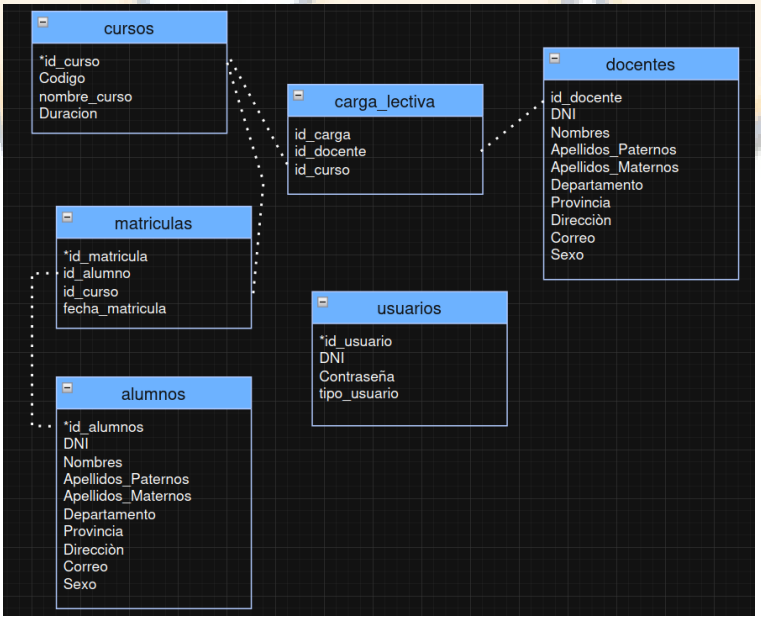
Table: Users (usuarios)

This table manages system access through credentials, ensuring that only registered users can log in. Additionally, it establishes specific roles to differentiate between administrators and students, assigning permissions based on their needs. It is crucial for the security and customization of the system's functionalities.

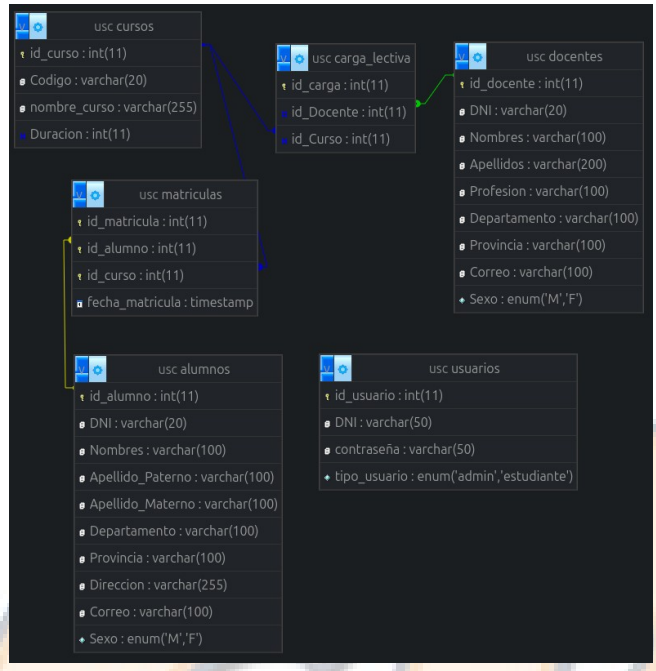
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id_usuario	int(11)			No	*Ninguna*		AUTO_INCREMENT
2	DNI	varchar(50)	*utf8mb4_general_ci*		No	*Ninguna*		
3	contraseña	varchar(50)	*utf8mb4_general_ci*		No	*Ninguna*		
4	tipo_usuario	enum('admin', 'estudiante')	*utf8mb4_general_ci*		No	*Ninguna*		

Database Diagram

Below is the logical diagram of the USC System, which was designed using the 'draw.io' program:

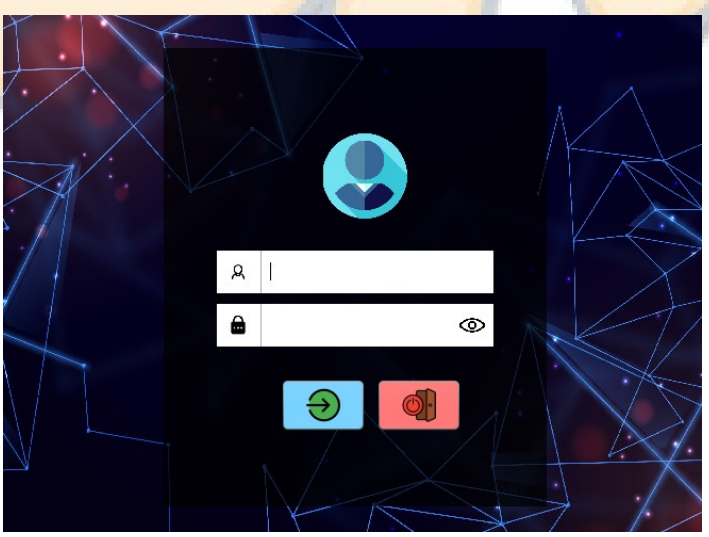


Now, the physical diagram of the database is presented, which was obtained from the MySQL database management system:



System Interfaces

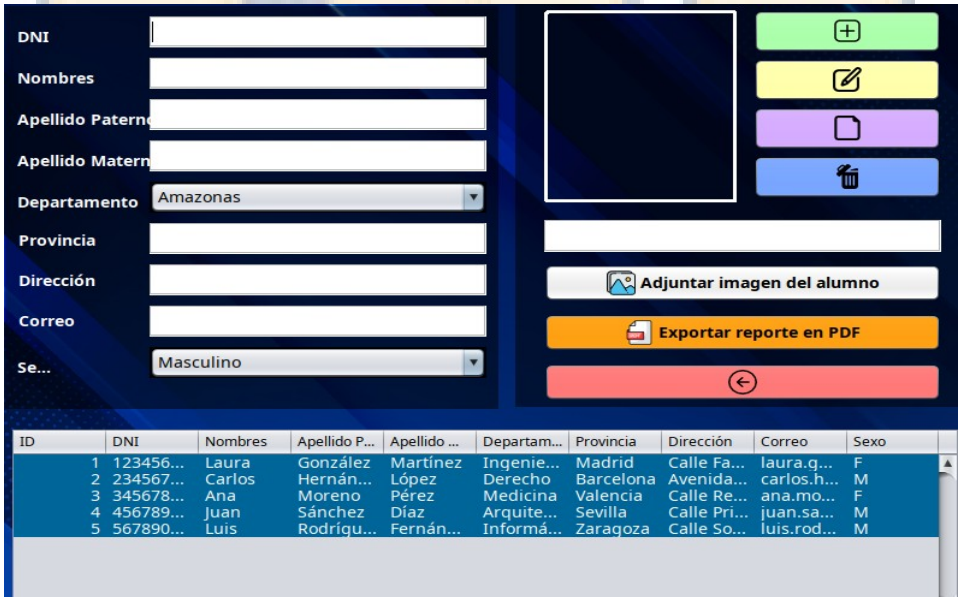
Firstly, upon starting the system, a login section will be displayed, where the user will need to enter their DNI and password. Depending on the type of user, the system will redirect them to their respective menu. These menus include the **Student Menu** (where they can view and enter their enrollment and personal data) or the **Administrator Menu** (where they can add new students, teachers, and courses, as well as consult the previously mentioned information). There is also a button available to log out of the system permanently if the user wishes to do so.



1) Administrator Menu:

In this section, there are the modules **Students**, **Teachers**, **Courses**, and **Assign**. Each module leads to a different section where various actions can be performed. The menu also includes a logout button, allowing the administrator to return to the initial section if desired.

This module is divided into two options: **Add** or **Consult**. The **Add** option will take the administrator to a CRUD (Create, Read, Update, Delete) interface, where they can add, edit, or delete any student, along with their respective data. Additionally, there will be a table listing all the students who have been entered into the system.



This option will redirect the administrator to a form where various text fields will be displayed. One of the fields will be for the **DNI**, where the administrator must enter the DNI of the student they wish to search for. The student must already be registered in the database. Once the DNI is entered, the form will display the respective details of the student being searched.

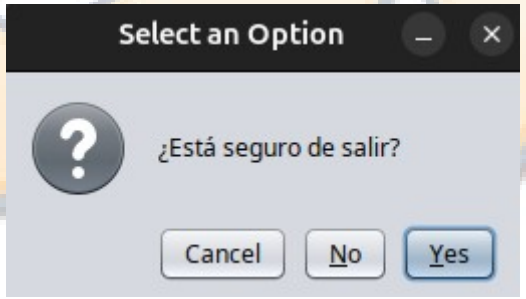
This module is divided into two options: **Add** and **Consult**. The **Add** option will redirect the administrator to a CRUD interface, where they can add, edit, or delete teachers from the system. Additionally, this form will include a table that lists all the teachers along with all their data.

This option will redirect the administrator to a form where various text fields will be displayed. One of the fields will be for the **DNI**, where the administrator must enter the DNI of the teacher they wish to search for. The teacher must already be registered in the database. Once the DNI is entered, the form will display the respective details of the teacher being searched.

This module has a single option: **Add**. Upon selecting this option, the administrator will be redirected to a CRUD interface where they can add, edit, or delete all courses in the system, along with their corresponding details. Additionally, the form will include a table that lists all the courses that are already registered in the database.

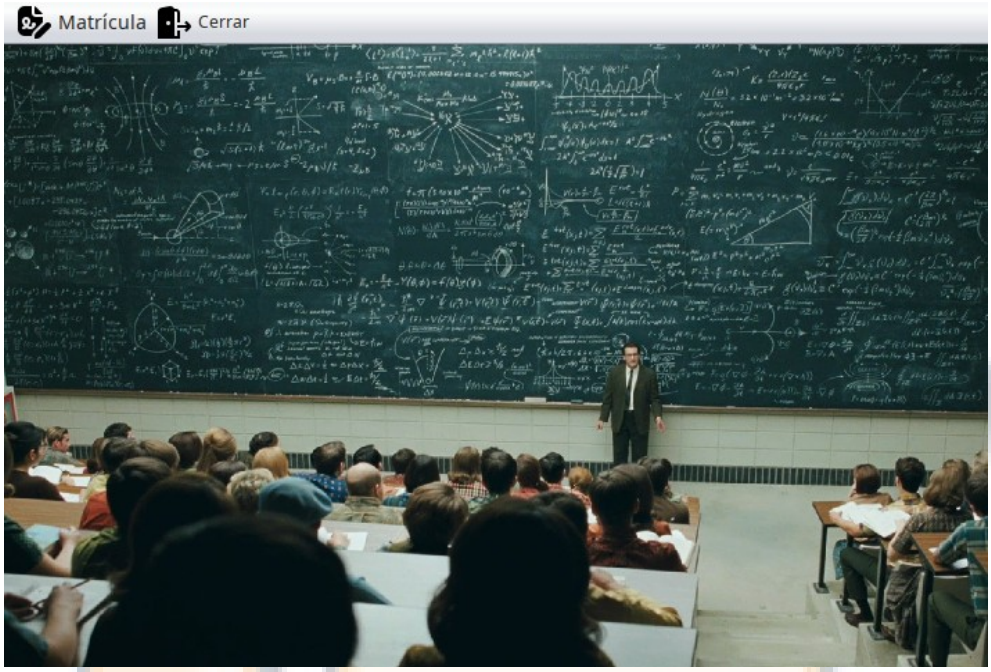
This module will have a single option: **Assign Course**. Upon selecting this option, the administrator will be redirected to a form where they can assign a course to the requested teacher. Additionally, the form will include a table that displays all the IDs of the assigned requests, along with their corresponding details.

When selecting this option, a message will appear asking if the administrator wants to log out of the menu. Upon confirmation, the system will redirect the user to the login interface, as it was initially.

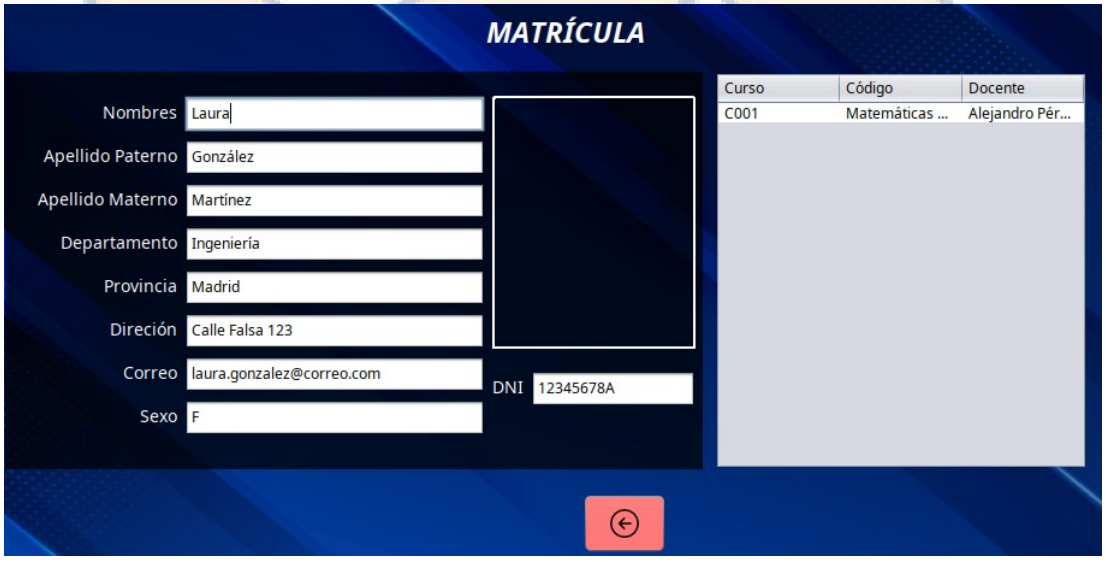


1) Student Menu:

In this section, there is only the **Enrollment** module, which is divided into two options: **View** and **Add**. Additionally, there is a logout button that allows the student to return to the initial section if desired.



The user will be shown all their corresponding data attached in the database related to their DNI, along with a table listing all the courses in which the student has previously enrolled.



This will redirect the student to a form where their personal details will be displayed, along with options to add an enrollment. The student will be able to choose the course they wish to enroll in, with necessary details shown, such as the course code, teacher, and duration in hours.

Having explained all the menus along with their respective modules and options, the USC system is now ready to be understood and tested. It is free to use under the jurisdiction and licensing of the established regulations. Thank you very much.

CONCLUSIONS

This personal project of a university management system reflects a significant effort to create a modern and efficient solution for academic and administrative management. Although it has not yet been implemented in a real environment, the development of the system demonstrates the feasibility of centralizing and automating essential tasks, such as managing students, teachers, courses, and enrollments, using contemporary technologies like NetBeans and MySQL. The integration of differentiated roles ensures that each user can interact with the platform in a personalized and secure way, improving both the user experience and the overall organization of the data.

The project not only showcases technical expertise and development skills but also the ability to identify current issues in academic management and propose practical solutions. Through testing and optimization, this system is prepared to adapt to different educational scenarios, making it a potentially valuable tool for any institution seeking to modernize its processes. In summary, this work emphasizes the importance of integrating technology into the educational field, highlighting its positive impact on the efficiency, security, and accessibility of institutional data.

REFERENCES

1. Oracle. (n.d.). MySQL Documentation. Retrieved from: https://dev.mysql.com/doc/
2. Apache NetBeans. (n.d.). NetBeans IDE Documentation. Retrieved from: https://netbeans.apache.org/kb/
3. Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach. McGraw-Hill Education.
4. IEEE Xplore. (n.d.). Database Management and Information Retrieval. Retrieved from: https://ieeexplore.ieee.org/
5. GitHub. (n.d.). Database Design Best Practices. Retrieved from: https://github.com/