



UNIVERSIDAD CATOLICA DE ESMERALDAS

# INFORME

# Proyecto: “Sistema de Visualización de Desempeño Académico y Restricciones de Materias para Matriculación en Estudiantes de la PUCESE”

# Nombre:

CAICEDO ORTIZ IBER JAVIER

# ASIGNATURA:

Programación

# FECHA:

12/11/2024

# CURSO:

2do de TIC’s

1. **Introducción**

El presente proyecto tiene como objetivo desarrollar una interfaz interactiva para que los estudiantes puedan ingresar su ID universitario o nombre y consultar el estado de sus materias aprobadas y reprobadas. Además, esta interfaz permitirá visualizar las restricciones de inscripción en semestres posteriores, en función de las materias reprobadas. Por ejemplo, si un estudiante reprueba Matemáticas en el primer semestre, se le notificará que no podrá matricularse en Cálculo en el segundo semestre ni en Estadísticas en el tercero. El sistema se implementará utilizando Java, JavaScript, PHP y una base de datos para gestionar la información académica de los estudiantes.

1. **Objetivos del Proyecto**

Objetivo General: Desarrollar un sistema que permita a los estudiantes consultar su estado académico y las restricciones en la inscripción de materias en semestres posteriores, basándose en un historial de aprobaciones y reprobaciones.

* 1. **Objetivos Específicos:**
* Implementar una base de datos que almacene las materias cursadas, los resultados académicos y las correlatividades entre materias.
* Diseñar una interfaz amigable en Java y JavaScript que permita el ingreso de datos y la visualización de restricciones académicas.
* Desarrollar una lógica en PHP para consultar la base de datos y calcular las restricciones de materias según las materias reprobadas.
* Generar un informe para el estudiante sobre su rendimiento y las restricciones académicas.

1. **Estado Actual del Proyecto**

Hasta el momento, el proyecto se encuentra en una fase inicial, en donde me he centrado principalmente en la recopilación de información como la malla curricular de la carrera de TIC’s, la definición de la estructura del sistema y la creación de un prototipo inicial.

* 1. **Investigación y Recopilación de Información**

Se ha realizado una investigación sobre la estructura curricular y las correlatividades entre materias, identificando qué materias dependen de otras para poder ser cursadas. Esto permitirá configurar las restricciones de inscripción de manera precisa.

* 1. **Diseño de la Base de Datos**

Se ha diseñado un esquema preliminar de la base de datos, que incluye las siguientes tablas:

* **Estudiantes**: ID, nombre, apellido.
* **Materias**: NRC, nombre, semestre.
* **Historial** **Académico**: ID estudiante, NRC materia, estado (aprobada/reprobada).
* **Correlatividades**: NRC materia origen, NRC materia destino (restricción).

Este diseño preliminar permitirá almacenar y gestionar la información necesaria para el sistema de restricciones.

1. **Próximos Pasos**

Finalizar el diseño de la base de datos con todas las correlatividades y dependencias entre materias.

Completar la interfaz gráfica para una navegación más intuitiva y para que los estudiantes puedan visualizar fácilmente el estado de sus materias y restricciones de inscripción.

Implementar la lógica de cálculo de restricciones en PHP para que el sistema verifique automáticamente qué materias no podrán cursarse debido a reprobaciones en semestres anteriores.

Realizar pruebas iniciales del sistema para verificar la precisión de los datos mostrados y la funcionalidad de las restricciones.

1. **Conclusión**

El proyecto avanza de acuerdo a lo previsto, habiendo completado la fase de diseño preliminar y habiéndose realizado avances en la interfaz de usuario y la estructura de la base de datos. Los siguientes pasos incluyen la finalización de la interfaz y la implementación de la lógica de restricciones, con miras a un sistema funcional que cumpla con las necesidades académicas de los estudiantes.

**CAPITULO 2**

1. **Introducción**

En este avance se han incorporado nuevas funcionalidades al sistema, las cuales incluyen el diseño de una interfaz web inicial y la creación de una base de datos funcional utilizando *DB Browser for SQLite*. Estas mejoras sientan las bases para el desarrollo completo del sistema, permitiendo gestionar la información académica de los estudiantes de manera eficiente.

1. **Descripción de los Avances**

**2.1. Interfaz Web**

Se desarrolló una interfaz web que permite a los estudiantes ingresar sus datos iniciales:

* **Campos del formulario**:
  + ID universitario (único para cada estudiante).
  + Nombre completo del estudiante.
* **Funcionalidad actual**:
  + Captura de los datos proporcionados por el usuario para su validación y posterior almacenamiento en la base de datos.

La interfaz está diseñada con un enfoque responsivo y amigable para el usuario, utilizando tecnologías web como HTML, CSS y JavaScript. Se prevé integrar funciones adicionales en futuras versiones, como la visualización de resultados y recomendaciones personalizadas.

**2.2. Base de Datos**

Se creó la estructura de la base de datos en *DB Browser for SQLite*, con las siguientes tablas principales:

1. **Tabla estudiante**
   * **Campos**:
     + id\_estudiante (TEXT, PK).
     + nombre (TEXT).
     + apellido
2. **Tabla materias**
   * **Campos**:
     + NRC (INTEGER, PK).
     + nombre\_materia (TEXT).
     + semestre (INTEGER).
3. **Tabla notas**
   * **Campos**:
     + id\_universitario (TEXT, FK).
     + NRC (INTEGER, FK).
     + estado (BLOB).
   * **Relación**: Relaciona estudiantes y materias para almacenar el desempeño académico.
4. **Tabla correlaciones**
   * **Campos**:
     + NRC\_origen (INTEGER, PK).
     + NRC\_destino (INTEGER, FK).
   * **Función**: Define las restricciones entre materias, indicando qué materias son prerrequisitos para otras.

**Relaciones entre tablas**

* estudiante ↔ notas: Un estudiante puede tener múltiples notas asociadas a diferentes materias.
* materias ↔ correlaciones: Establece dependencias entre materias para la validación de prerequisitos.

**4. Próximos Pasos**

1. Implementar la funcionalidad de visualización de datos académicos en la interfaz web.
2. Crear algoritmos para validar las restricciones de materias en base a las correlaciones.
3. Integrar el sistema con PHP o Node.js para la comunicación entre la base de datos y la interfaz web.