**Gestión interna universidades**

**Presentado por:**

**Andres Felipe Medina Lozano**

**Nicolas Andres Rodríguez Lasso**

**Universidad Santo Tomás Seccional Tunja**

**Facultad de Ingeniería de Sistemas**

**Proyecto de Responsabilidad Social**

**Tunja, Boyacá**

**2020**

**CONTENIDO**

Tabla de contenido

[1. GENERALIDADES 2](#_Toc54675351)

[1.1 Antecedentes 3](#_Toc54675352)

[1.2 Planteamiento del problema 3](#_Toc54675353)

[1.3 justificación 3](#_Toc54675354)

[1.4 Marco de referencia 4](#_Toc54675355)

[1.4.1 Marco teórico 4](#_Toc54675356)

[1.4.2 Marco conceptual 5](#_Toc54675357)

[1.5 Objetivos 5](#_Toc54675358)

[1.5.1 Objetivo general 5](#_Toc54675359)

[1.5.2 Objetivos específicos 5](#_Toc54675360)

[1.6 Metodología 5](#_Toc54675361)

[2. ESPECIFICACION DE REQUERIMIENTOS 6](#_Toc54675362)

[2.1 Descripción general 6](#_Toc54675363)

[2.2 Características de los usuarios 6](#_Toc54675364)

[2.3 Requerimientos de interfaz externa 7](#_Toc54675365)

[2.4 Requerimientos funcionales 7](#_Toc54675366)

[2.5 Requerimientos no funcionales 8](#_Toc54675367)

[3 DISEÑO DE LA BASE DE DATOS 9](#_Toc54675368)

[3.1 Modelo conceptual 9](#_Toc54675369)

[3.2 Modelo relacional 9](#_Toc54675370)

[REFERENCIAS 9](#_Toc54675371)

# GENERALIDADES

## Antecedentes

La entidad para la que va a a ser aplicado el proyecto es la Universidad santo tomas secciona Tunja, siendo esta un sector educativo.

## Planteamiento del problema

La universidad quiere crear un sistema que le permita tener un control exhausto y riguroso de sus estudiantes a nivel generar, que lleva a cabo información personal de este, que estudia, a que pensum pertenece, en que semestre académico se encuentra, a que división académica esta asignada su facultad. Esto con el fin de controlar mejor los procesos de aprendizaje, mirar con detalles las problemáticas de las dificultades académicas o por que se retira o ¿Por qué un estudiante repite una materia hasta 3 veces? , ya que la deserción escolar es uno de los índices que más afecta y la razón por la cuela esta va a ser hecha ya que miden los índices de calidad de las Universidades por los índices de deserción., se desarrollara el plan piloto en la sede Tunja, pero es necesario que esta estrategia poda ser implementada en todas sus sedes.

## justificación

Al desarrollar este proyecto estamos ayudando a crear un sistema de información y gestión de las universidades a nivel general, orientados a la administración de datos e información, organizados y listos para su posterior uso, esto le permitirá generar un sistema ampliamente especializado sacando reporte de Cuantos estudiantes de cierta facultad, se encuentra atrasados en el desarrollo de su pensum Por ello, es que se hace necesario Universidad Santo Tomas crear un sistema de información para el pleno desarrollo de los procesos académicos en la misma, donde se trate y se administre bien la información de la comunidad tomasina.

Modulo de Estudiante: Le permite ingresar la información del estudiante, numero de su documento, lugar de expedición, nombres apellidos entre otra información básica.

Modulo de facultad: Le permite crear edita y eliminar facultades, teniendo en cuenta que cada una de estas cuentas con un decano.

Modulo Pensum

## 1.4 Marco de referencia

### 1.4.1 Marco teórico

Los sistemas de información corresponden a un conjunto de elementos que interactúan entre sí para soportar el proceso de toma de decisiones en una organización, elementos que se estructuran sobre una base de información de vital importancia para controlar los procesos que controlan las actividades desarrolladas por cualquier empresa.

Básicamente están compuestos por hardware, software, talento humano, comunicaciones, y datos, los cuales se integran para obtener datos de entrada y generar información útil como salida fundamental para la toma de decisiones.

Los sistemas de información se pueden clasificar de la siguiente manera:

Sistemas de información geográfica (SIG), los cuales vinculan una ubicación con información de interés relacionada, lo que le permite a un usuario generar información que se asocia a un espacio físico a través de mapas georreferenciados mediante bases de datos con la ventaja de representar los resultados en entornos web o dispositivos móviles de modo ágil e intuitivo[[1]](#footnote-1).

Sistemas de procesamiento de transacciones (SPT), los cuales se encargan de efectuar transacciones cotidianas que facilitan el funcionamiento de una organización.

Sistemas de control de procesos de negocio, (BPM) los cuales permiten gestionar los procesos a través de indicadores.

Sistemas de planeación empresarial, Enterprise Resource planning (ERP) los cuales apoyan los diferentes procesos permitiendo gestionar el flujo de información controlando todas las tareas de oficina.

### 1.4.2 Marco conceptual

### pgAdmin 4

Es una herramienta de código abierto para la administración de bases de datos PostgreSQL y derivados (EnterpriseDB Postgres Plus Advanced Server y Greenplum Database). Incluye:

* Interfaz administrativa gráfica
* Herramienta de consulta SQL (con un EXPLAIN gráfico)
* Editor de código procedural
* Agente de planificación SQL/shell/batch
* Administración de Slony-I
* pgAdmin se diseña para responder a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas.

La interface gráfica soporta todas las características de PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, FreeBSD, Mac OSX y Solaris.

PostgreSQL

Es un sistema de código abierto de administración de bases de datos del tipo relacional, aunque también es posible ejecutar consultas que sean no relaciones. En este sistema, las consultas relacionales se basan en SQL, mientras que las no relacionales hacen uso de JSON.

Como decíamos, se trata de un sistema de código abierto y además gratuito, y su desarrollo es llevado adelante por una gran comunidad de colaboradores de todo el mundo que día a día ponen su granito de arena para hacer de este sistema una de las opciones más sólidas a nivel de bases de datos.

Dos detalles a destacar de PostgreSQL es que posee data types (tipos de datos) avanzados y permite ejecutar optimizaciones de rendimiento avanzadas, que son características que por lo general solo se ven en sistemas de bases de datos comerciales, como por ejemplo SQL Server de Microsoft u Oracle de la compañía homónima.

## 1.5 Objetivos

### 1.5.1 Objetivo general

Diseñar e implementar apropiadamente modelos conceptuales y procedimientos almacenados en bases de datos relacionales, manejar los conceptos básicos de fragmentación de datos y comprender los conceptos relacionados con la estructura e implementación de un motor de base de datos.

### 1.5.2 Objetivos específicos

* Entender la estructura interna de un motor de bases de datos, entre ellos los archivos, índices y procesamiento de consultas.
* Realizar un modelo entidad relación el cual permita generalizar más la problemática descrita.
* Generar una base de datos que solucioné las necesidades para la implementación futuro del sistema de información

## 1.6 Metodología

El instrumentó que primordialmente se utilizara es el levantamiento de requerimientos y el modelado de la Base de datos por medio de Star UML un lenguaje unificado para el modelado y diseño de software.

Otro instrumento que se utilizaría para el trabajo de campo y desarrollo de este proyecto es el lenguaje posgresSQL con el motor Pgadmin en su versión 4 .

Tabla 1. *Fases de investigación.*

*Fases de investigación y desarrollo*

|  |  |  |  |
| --- | --- | --- | --- |
| **Fase** | **Objetivo** | **Instrumentos** | **Análisis** |
| Primera fase, documentación | Análisis documental de los referentes teóricos y requisitos. | Diseño estructura metodológica, y planteamiento de un bosquejo. | Descripción densa del marco contextual. |
| Segunda fase, consolidación | Describir las relaciones que se dan entre lo teórico y el objeto de desarrollo | Recopilación de información documental. | Descripción analítica de las funcionalidades de la base de datos |
| Tercera Fase:  Desarrollo y diseño. | Crear y hacer un diseño la base de datos | Matriz documental | Análisis y categorización de prioridades del MER. |
| Cuarta Fases: Elaboración del documento final, con el producto final | Compilación y pruebas del BD. | Documento de apoyo generado a partir del resultado final. | Propuesta finalizada para la Base de datos |

Fuente: Autores.

# ESPECIFICACION DE REQUERIMIENTOS

## Descripción general

Plantilla de requerimientos de análisis y diseño de software para el levantamiento de requerimientos de la DB

## 2.2 Requerimientos de interfaz externa

*En esta sección se describen a través de maquetas (mockup), los requerimientos en cuanto a interfaz gráfica expuestos para mostrar el manejo de la información de algunas tablas (no todas, sólo las que va a manipular en los reportes) específicas utilizadas en los informes y reportes y salidas de código plgsql.*

*Igualmente, los mockups para la salida de los informes y reportes.*

*En general, la interfaz de entrada, de manipulación de algunas tablas, y los reportes y salidas.*

## Requerimientos funcionales

|  |  |
| --- | --- |
| ID: | RF-1 |
| NOMBRE: | -- |
| ACTOR: | Administrador, Decanos, secretarios |
| DESCRIPCIÓN: | Guardar la información básica de un estudian Nombres apellidos documento teléfono fecha de expedición, etc |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | Ingreso del usuario  Registrar la persona |
| POS CONDICIÓN: | Al final de haber realizado el registro debe arrojar un mensaje de éxito o error |

|  |  |
| --- | --- |
| ID: | RF-2 |
| NOMBRE: | -- |
| ACTOR: | Administrador, Decanos, secretarios |
| DESCRIPCIÓN: | Le permite crear una facultad, modificar datos de esta o eliminarla solo si es administrador |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | No se podrá borrar una facultad si a esta esta asociada un estudiante y un pensum.  Registrar las nuevas facultades con su información necesaria, nombre, división a la que pertenece, decano y teléfono |
| POS CONDICIÓN: | Al final de haber realizado el registro o la edición debe arrojar un mensaje de éxito o error |

|  |  |
| --- | --- |
| ID: | RF-3 |
| NOMBRE: | -- |
| ACTOR: | Administrador, Decanos, secretarios |
| DESCRIPCIÓN: | Le permite crear una Asignatura, modificar datos de esta o eliminarla solo si es administrador |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | Permite registrar una nueva materia o cambiar los datos de alguna existente esta debe contener, nombre, número de créditos, horas presenciales entre otros |
| POS CONDICIÓN: | Al final de haber realizado el registro o la edición debe arrojar un mensaje de éxito o error |

|  |  |
| --- | --- |
| ID: | RF-4 |
| NOMBRE: | -- |
| ACTOR: | Administrador, Decanos, secretarios |
| DESCRIPCIÓN: | Le permite crear una Estudiante programa donde asocien un pensum, un programa, y una persona a este para relacionar al estudiante |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | Permite registrar un nuevo registro de estudiante programa o cambiar los datos si se cambia de carrera y continúan en la universidad o si realizan un cambio de pensum a nivel general |
| POS CONDICIÓN: | Al final de haber realizado el registro o la edición debe arrojar un mensaje de éxito o error |

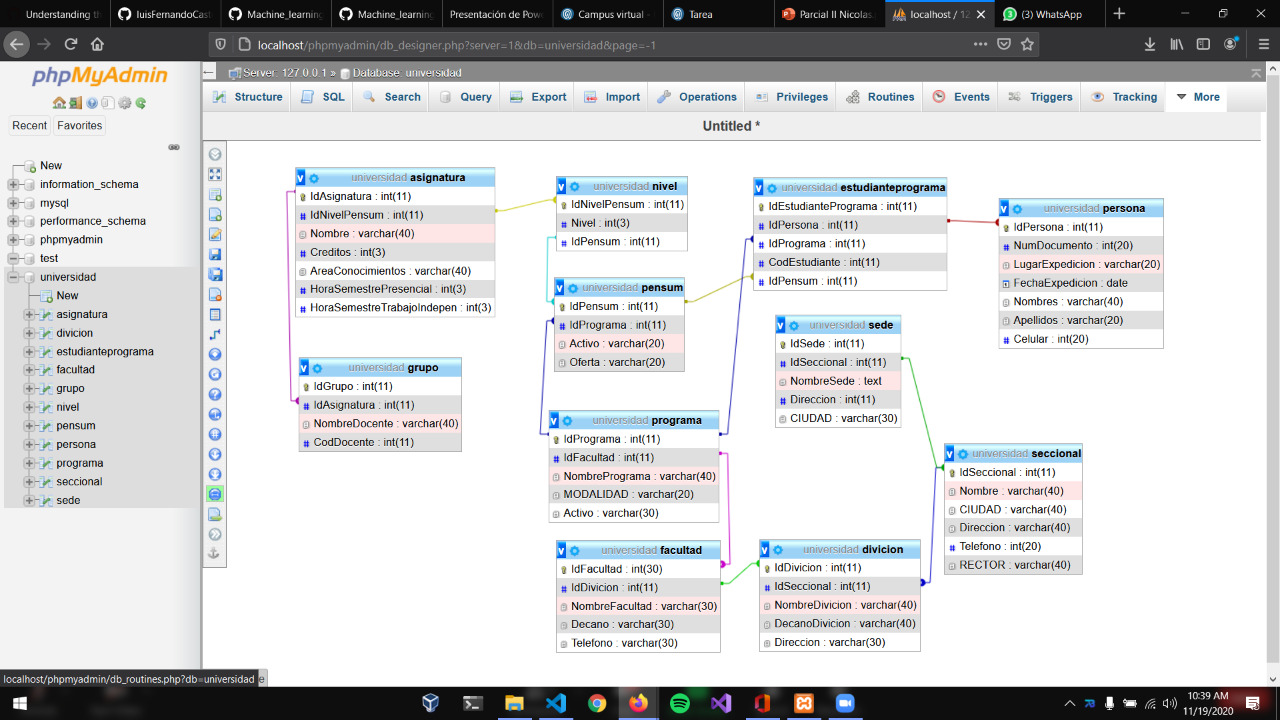
|  |  |
| --- | --- |
| ID: | RF-5 |
| NOMBRE: | -- |
| ACTOR: | Administrador, Decanos, secretarios |
| DESCRIPCIÓN: | Le permite crear un programa donde asocien una facultad para poder crearlo en el sistema |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | Permite registrar un nuevo registro de programa o cambiar los datos este debe tener nombre del programa, facultad y el estado del programa si es activo o inactivo. |
| POS CONDICIÓN: | Al final de haber realizado el registro o la edición debe arrojar un mensaje de éxito o error |

|  |  |
| --- | --- |
| ID: | RF-6 |
| NOMBRE: | -- |
| ACTOR: | Administrador |
| DESCRIPCIÓN: | Le permite crear una sede en la base de datos |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | Permite registrar un nuevo registro de una sede o cambiar los datos este debe tener nombre de la sede, dirección y ciudad donde está ubicada. |
| POS CONDICIÓN: | Al final de haber realizado el registro o la edición debe arrojar un mensaje de éxito o error |

|  |  |
| --- | --- |
| ID: | RF-7 |
| NOMBRE: | -- |
| ACTOR: | Administrador |
| DESCRIPCIÓN: | Le permite crear un reporte de los estudiantes que han pedido mas de 2 veces la materia. |
| PRECONDICIÓN: | Ingresar al Aplicativo |
| FLUJO DE EVENTOS: | Genera una lista de los estudiantes que han perdido dos veces una misma asignatura |
| POS CONDICIÓN: | Al final de haber realizado el registro o la edición debe arrojar un mensaje de éxito o error |

# DISEÑO DE LA BASE DE DATOS

## Modelo conceptual



Repositorio: https://github.com/afml9954/proyectomotores.git

1. Si son dos estudiantes:

* Mínimos 12 tablas, con máximo 3 tablas base (son las que no tienen foráneas). Con nomenclatura definida con colores y convenciones utilizadas.
* Mínimo 1 tabla con PK compuesta.
* Mínimo 2 campos con restricción de check.
* Mínimo 6 tablas con relaciones de 2 fk o más.

1. Si son tres estudiantes:

* Mínimos 14 tablas, con máximo 5 tablas base. Con nomenclatura definida con colores y convenciones utilizadas.
* Mínimo 2 tablas con PK compuesta.
* Mínimo 3 campos con restricción de check.
* Mínimo 8 tablas con relaciones de 2 fk o más.

## 3.2 Modelo relacional

*Se peg-a y se deja en archivo exterior, el Modelo Relacional, no sólo como imagen, sino también con la extensión con el software que fue creado. Aquí se define el nombre del aplicativo en el que fue hecho.*

\* Se entrega el backup de la base de datos.

\* Se entrega el script de la creación de la base de datos.

\* Se entrega el script Insert.

## 3.3 Consultas

# *Se piden mínimo 5 consultas haciendo uso de Join con mínimo el enlace de 3 tablas utilizando cláusulas de consulta y ordenamiento.*

# *Aquí se escribe el enunciado, el código de las consultas y las salidas de las consultas.*

## 3.3 Código PLSQL

# *Lo que se pide se define la próxima clase según los temas vistos.*

*Aquí se escribe el enunciado, el código del código PLSQL y las salidas de las consultas*

# REFERENCIAS

*Preferiblemente consulta del CRAI, y de libros impresos. O artículos científicos. Con normas APA, o la que deseen.*

1. SIG.CEA.ES, 2010. *SIG* [en línea]. 2010. S.l.: s.n. Disponible en: [http://sig.cea.es/SIG.](http://sig.cea.es/SIG) [↑](#footnote-ref-1)