

**Portada**

Estudiante

Alejandro Quintero Zeledon

Profesor

Luis Alonso Bogantes Rodríguez

Año

2020

Contenido

**Introducción**……………………………………………………………………………………………………………………2

**Objetivos**……………………………………………………………………………………………………………………..……3

[**1- Descripción del problema** 3](#_Toc77685504)

[**2. Análisis de requerimientos funcionales** 4](#_Toc77685505)

[**3. Reglas básicas del negocio** 5](#_Toc77685506)

[**4. Modelo conceptual de la base de datos** 6](#_Toc77685507)

[**5. Planteamiento a nivel conceptual de la solución** 7](#_Toc77685508)

[**6. Crear el modelo lógico de la base de datos** 8](#_Toc77685509)

[**7.Crear el diccionario de datos de la base de datos** 9](#_Toc77685510)

**11.Conclucion**………………………………………………………………………………………………………………11

**Introducción**

Venimos a crear la solución del proyecto empresarial mediante el diseño e implementación de la base de datos, para eso vamos a revisar cuales son los requisitos dados por el cliente, una vez dados y revisados vamos a buscar cuales son otras opciones que se pueden agregar a esa base de datos para que esta sea mas completa después de eso se crean las reglas de negocio lo cual da los limites de cada información, una vez elegido cuales son las mejores opciones para agregar vamos a hacer un modelo conceptual el cual creamos las tablas necesarias para ver si la información concuerda con las tablas, una vez evaluado eso, vamos al modelo conceptual, en el hacemos una normalización la cual hace que la tabla este mucho mas optimizada y de una ahí mismo unimos las llave foráneas y las primarias una vez todo eso juntos creamos el diccionario de datos el cual describe mucho mejor que hace que datos de las tablas.

**Objetivo general**

Resolver el problema de la clientela creando una base de datos funcional, que cumpla con todos los requisitos dados, y los necesarios para que el programa funcione correctamente, evaluando que tipo de información se introduce.

**Objetivos específicos**

Elaborar las tablas necesarias

Identificar cuales son los posibles datos que se introducen

Definir las propiedades necesarias de cada tabla

# **1- Descripción del problema**

La Universidad **“Fuente del Conocimiento”** va a brindar sus servicios de enseñanza a partir el año 2022, y entre sus prioridades está el diseño de la Base de Datos.

Se requiere que la Base de Datos administre la información de los estudiantes, la matrícula, el pago de los módulos, las calificaciones obtenidas en cada materia y la información de los docentes. Además, debe administrar la información de los programas que se imparten en la universidad y los cursos comprendidos en cada uno de los programas.

Usted ha sido contratado para el diseño de la base de datos en 3 etapas: (1) Diseño, (2) Implementación y (3) la creación de los scripts de consultas, procedimientos almacenados, funciones.

Es importante que contemple la siguiente información en el diseño, análisis e implementación del sistema de Bases de Datos que se necesita.

# 2. Análisis de requerimientos funcionales

El programa deberá tener información de cada estudiante, programa, materia, profesores, certificados de los profesores, laboratorios, cursos, becas.

El programa tendrá la posibilidad de crear un curso nuevo y tendrá su propio código independiente, los cursos nuevos tendrán toda la información correspondiente como horario del día cantidad de días a la semana, profesor certificado, laboratorio y fecha de inicio.

El sistema registrará los días feriados no lectivos para que el programa pueda reconocer perfectamente cuando finalizara el modulo

El programa debe tener los pagos completos y los pagos deben ser el pago completo para poder recibir la nota final

El programa debe guardar las horas de cada curso e igualmente de los laboratorios para que no pegue ningún horario, si sucede no puede llevar uno de esos cursos

# **3. Reglas básicas del negocio**

1. El horario de la universidad va a ser de 7:00am a 6:00pm de lunes a viernes, por lo tanto, las lecciones deben estar contempladas en ese horario.
2. Los cursos pueden tener un mínimo de dos horas y un máximo de 5 horas diarias, pero en un único bloque al día, es decir, un curso no se puede impartir en la mañana y luego en la tarde del mismo día. Además, se puede impartir: uno, dos, tres, cuatro o cinco días por semana.
3. Cuando se abre un curso se solicita al usuario la fecha de inicio. El sistema calculará la fecha final considerando los datos anteriores (días y horarios en que se impartirá), además de los días feriados, los cuales no son lectivos.
4. El sistema debe registrar los pagos realizados por los estudiantes.
5. Los profesores deben tener una certificación para impartir un determinado curso. Un profesor puede tener certificaciones para impartir diferentes módulos, mientras que otro sólo podría tener una sola certificación, por lo que sólo podrá impartir un solo módulo en la Universidad.
6. Cada curso tiene calificaciones finales, cuya nota mínima es de 70 para continuar en el programa. Debe verificar el cumplimiento de este requisito en cada nueva matrícula.
7. Un estudiante no va a recibir la nota final si no ha cancelado el curso, por ende, tampoco podrá avanzar al siguiente módulo.
8. Un estudiante no puede matricular un curso si registra choque de horarios con otros módulos.
9. Cada módulo para ser abierto debe tener su profesor asignado, y verificar que no haya un choque de horario con otros cursos previamente asignados al profesor.
10. También debe contemplar que para abrir un curso debe tener un laboratorio asignado. Verificar que no haya choques de horario en los laboratorios con otros módulos asignados al laboratorio
11. A los módulos se les puede asignar los horarios correspondientes y calcular su fecha final, pero se debe asignar un profesor y un laboratorio antes de colocar el módulo como “abierto”, es decir, solo se puede abrir un curso si tiene profesor y laboratorio previamente asignados.
12. El estudiante puede pagar una parte del curso, pero para poder recibir la nota final necesitará pagar el total
13. Cada profesor puede dar las materias que su horario permita mientras tenga el certificado que corresponda a esa materia
14. Solo se disponen de 10 laboratorios los cuales están abiertos de 7am a 6pm y cada curso debe repartirse en ese horario
15. El estudiante puede llevar diferentes cursos mientras los horarios no choquen entre ellos

# **4. Modelo conceptual de la base de datos**

Una captura de pantalla de una computadora

Descripción generada automáticamente

# **5. Planteamiento a nivel conceptual de la solución**

-Primero se colocan la información de los programas

-luego se colocan las materias o cursos de cada programa

-los cursos tienen cursos abiertos que ellos muestran cuales están activos y cuales no

- igualmente esos cursos abiertos muestran el horario del profesor el cual el horario se una con el profesor, que igualmente muestra cuales profesores dan ese curso y su horario

-antes de eso los cursos abiertos evalúan si el profesor tiene la certificación para poder dar esa materia dado que si, se coloca un horario activo de el y se conectan los 3 puntos (horarioProfesor,cursosAbiertos,Profesores)

-Para los estudiantes deben meter su información básica igualmente como los profesores

-Los estudiantes tienen su nota de cada curso hecho, pero antes de eso debe tener una matricula la cual la matricula hace una unión con cursos abiertos, ya que primero se evalúa que cursos están abiertos para poder matricularse, después de eso se escoge el horario dependiendo del curso y sucede lo mismo anterior sobre el horarioProfesor cursos y profesores, así para saber que profesor da a que hora del día, e igualmente en la matricula debe ser un paco “completo” para poder recibir el curso

# **6. Crear el modelo lógico de la base de datos**

Una captura de pantalla de una computadora

Descripción generada automáticamente

# **7.Crear el diccionario de datos de la base de datos**

Programas

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Programas | Varchar(20) | X | Llave primaria |
| NombreProgramas | Varchar(80) | X |  |

Cursos

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Curso | Varchar(20) | X | Llave primaria |
| ID\_CursoAbierto | Varchar(20) | X | Llave foránea |
| ID\_Programas | Varchar(20) | X | Llave foránea |
| NombreCurso | Varchar(60) | X |  |
| Duración | Int | X | --Se almacenará en horas |
| Precio | Float | X |  |

CursosAbiertos

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_CursoAbierto | Varchar(20) | X | Llave primaria |
| ID\_Certificado | Varchar(20) | x | Llave foranea |
| ID\_HorarioProfesor | Varchar(20) | X | Llave foranea |
| Fecha\_inicio | Date | X |  |
| Fecha\_final | Date | X |  |

Horario

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Horario | Varchar(20) | X | Primary key |
| ID\_CursoAbierto | Varchar(20) | x | Llave foranea |
| Hora\_inicio | Time | X |  |
| Hora\_final | Time | X |  |
| Dia | Varchar(20) | X | --se colocan días como “L”,”k”,”M”,”J”,etc |

DetalleMatricula

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Matricula | Varchar(20) | x | Llave foránea |
| ID\_CursoAbierto | Varchar(20) | X | Llave foranea |

Matricula

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Matricula | Varchar(20) | X | Primary key |
| ID\_Estudiantes | Varchar(20) | X | Llave foranea |
| FechaMatricula | Date | X |  |
| TotalMatricula | Int | X |  |
| EstadoPago | Varchar(20) | X | --puede ser “completo” o pago “incompleto” |

Estudiantes

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Estudiantes | Varchar(20) | x | Llave primaria |
| Nombre | Varchar(20) | X |  |
| Apellido1 | Varchar(20) | X |  |
| Apellido2 | Varchar(20) | X |  |
| Correo | Varchar(50) | X |  |
| telefono | int | X |  |

NotasMateria

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Nota | Varchar(20) | X | Llave primaria |
| ID\_Curso | Varchar(20) | X | Llave foránea |
| ID\_Estudiantes | Varchar(20) | X | Llave foranea |
| Periodo | Int | X |  |
| Nota | Float | X |  |

Laboratorios

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Laboratorio | Varchar(20) | x | Llave primaria |
| ID\_Curso | Varchar(20) | X | Llave foránea |
| NombreLaboratorio | Varchar(50) | X |  |

Certificaciones

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Certificado | Varchar(20) | x | Llave primaria |
| ID\_Profesor | Varchar(20) | X | Llave foranea |
| Materia | Varchar(20) | x |  |

Profesores

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_Profesor | Varchar(20) | x | Llave primaria |
| Nombre | Varchar(20) | X |  |
| Apellido1 | Varchar(20) | X |  |
| Apellido2 | Varchar(20) | X |  |
| Correo | Varchar(50) | X |  |
| telefono | Int | X |  |

HorarioProfesores

|  |  |  |  |
| --- | --- | --- | --- |
| Campo | Tipo | Not null | Observaciones/Restricciones |
| ID\_HorarioProfesor | Varchar(20) | X | Llave primaria |
| ID\_Profesor | Varchar(20) | X | Llave foranea |
| Hora\_inicio | Datetime | X |  |
| Hora\_final | Datetime | X |  |

**Conclusión**

A la hora de crear una base de datos con la información que el cliente nos da, debemos analizar muy muy bien cada dato, también si se necesitaran mas, es un poco complejo el análisis pero una de las partes mas importantes una vez que tenga un buena análisis del problema se puede ir a la base de datos, el análisis para mi fue una de las partes mas complicadas, pero una vez un poco mas entendido se puede continuar de buena forma