Prueba Técnica de Conocimientos

|  |  |
| --- | --- |
| Nombre Postulante: | Diego Soto |
| Rut: | 15.366.708-K |
| Fecha: | 12-10-2021 |

# PARTE 1. DECLARATORIA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Lenguaje/Tecnología | Nivel | Meses de experiencia codificando | Última vez usado (mes/año) |
| 1 | Visual Studio .Net - C# o VB | Bajo |  |  |
| 2 | SQL Server T-SQL | Medio |  |  |
| 3 | HTML5 - CSS3 | Medio |  |  |
| 4 | Javascript - JQuery | Medio |  |  |
| 5 | MVC | Bajo |  |  |

# PARTE 2. BASE DE DATOS

## 2.1 DIAGRAMA MODELO DATOS

### 2.1.1 EXPLICACIÓN MODELO

En el problema planteado, el ***curso*** se vislumbra como una entidad central, que surge como una relación entre varias entidades básicas o elementales. Se evidencian las siguientes entidades:

* La entidad *anyo* representa *cuándo* (en unidad de tiempo año) se dicta un *curso*. Un curso puede ser dictado en un año dado, y en un año pueden dictarse muchos cursos. (1-N)
* La entidad *bimestre* representa *cuándo*, en unidad de tiempo bimestre, se dicta un *curso*. Un curso puede ser dictado en un bimestre dado, y en un bimestre pueden dictarse muchos cursos. (1-N)
* La entidad *asignatura* representa qué (área de conocimiento) se dicta en un *curso*. En un curso se dicta una asignatura, y una asignatura puede ser dictada en muchos cursos. (1-N)
* La entidad *profesor* representa quién (persona) dicta un *curso*. Un curso puede ser dictado por un profesor y un profesor puede dictar muchos cursos a lo largo del tiempo. (1-N)
* La entidad *sala* representa dónde (lugar físico) se dicta el curso. Un curso puede ser dictado en una sala y en una sala se puede dictar muchos cursos a lo largo del tiempo. (1-N)

Adicionalmente, los cursos disponibles para ser dictados o impartidos por la institución educacional pueden ser tomados por un alumno. A esta relación que se origina entre el alumno y el curso se le ha llamado matrícula

* La entidad *alumno* representa quién (persona) toma el curso. Un curso puede ser tomado por un alumno y un alumno puede tomar muchos cursos a lo largo del tiempo (1-N)
* La entidad *matrícula* representa la toma o realización de un curso por un alumno. Es decir, relaciona las entidades alumno y curso. Un alumno puede realizar muchas matrículas y una matrícula es realizada por un alumno (1-N). Un curso es tomado en una matrícula y en una matrícula se toma un curso (1-1)

Finalmente existen relaciones de probidad entre los profesores y las asignaturas, asi como entre las salas y las asignaturas. Estas son representadas por las siguientes entidades de asociación

* La entidad *profesor\_asignatura* especifica qué asignatura puede dictar qué profesor
* La entidad sala\_asignatura especifica en qué sala se puede dictar qué asignatura

### 2.1.2 ESQUEMA DE TABLAS

Por facilidad y consistencia en la manipulación de los datos en el posterior desarrollo, se ha evitado el uso de llaves compuestas, definiendo las llaves primarias como una columna id de tipo entero y agregando restricciones de unicidad adicionales cuando sea necesario.

|  |  |  |  |
| --- | --- | --- | --- |
| **Entidad** | **Columna** | **Tipo** | **Descripción** |
| anyo | id (pk) | int | Identificador primario |
| anyo | int | El año (Ej: 2011, 2012, 2013, etc.) |
| bimestre | id (pk) | int | Identificador primario |
| bimestre | int | El bimestre 1, 2, 3 o 4 |
| nombre | varchar(50) | El nombre del bimestre |
| asignatura | id (pk) | int | Identificador primario |
| código | varchar(50) | Código alfanumérico identificador de alto nivel |
| nombre | varchar(50) | El nombre de la asignatura |
| profesor | id | int | Identificador primario |
| rut | varchar(10) | El rut del profesor |
| nombre | varchar(50) | El nombre del profesor |
| sala | id | int | Identificador primario |
| código | varchar(50) | Código alfanumérico identificador de alto nivel |
| capacidad | int | Cantidad de alumnos que puede contener |
| curso | id | int | Identificador primario |
| id\_anyo | int | Llave foránea referencia a entidad anyo |
| id\_bimestre | int | Llave foránea referencia a entidad bimestre |
| id\_asignatura | int | Llave foránea referencia a entidad asignatura |
| id\_sala | int | Llave foránea referencia a entidad sala |
| id\_profesor | int | Llave foránea referencia a entidad profesor |
| código | varchar(50) | Código alfanumérico identificador de alto nivel |
| estado | varchar(50) | Estado del curso (Abierto o Cerrado) |
| alumno | id | int | Identificador primario |
| rut | varchar(10) | El rut del alumno |
| nombre | varchar(50) | El nombre del alumno |
| nacimiento | datetime | La fecha de nacimiento del alumno |
| género | varchar(50) | Género del alumno (Ej: Masculino o Femenino) |
| matricula | id | int | Identificador primario |
| id\_alumno | int | Llave foránea referencia a entidad alumno |
| id\_curso | int | Llave foránea referencia a entidad curso |
| nota | decimal | La nota que obtuvo el alumno al final del curso |
| código | varchar(50) | Código alfanumérico identificador de alto nivel |
| profesor\_asignatura | id | int | Identificador primario |
| id\_profesor | int | Llave foránea referencia a entidad profesor |
| id\_asignatura | int | Llave foránea referencia a entidad asignatura |
| sala\_asignatura | id | int | Identificador primario |
| id\_sala | int | Llave foránea referencia a entidad sala |
| id\_asignatura | int | Llave foránea referencia a entidad asignatura |

### 2.1.3 RESTRICCIONES DE IDENTIDAD REFERENCIALES, DE DOMINIO Y CLAVES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidad** | **Nombre** | **Tipo** | **Columnas** | **Descripción** |
| anyo | UNIQ\_anyo | unique | anyo | Restricción de unicidad para el año |
| bimestre | UNIQ\_bimestre | unique | bimestre | Restricción de unicidad para el bimestre |
| CHECK\_bimestre | check | bimestre | El valor del bimestre no puede ser mayor a 4 |
| asignatura | UNIQ\_codigo | unique | código | Restricción de unicidad para el código |
| UNIQ\_nombre | unique | nombre | Restricción de unicidad para el nombre |
| profesor | UNIQ\_rut | unique | rut | Restricción de unicidad para el rut |
| sala | UNIQ\_codigo | unique | código | Restricción de unicidad para el código |
| curso | FK\_curso\_anyo | fk | id\_curso | Rstricción referencial hacia anyo |
| FK\_curso\_bimestre | fk | id\_bimestre | Rstricción referencial hacia bimestre |
| FK\_curso\_asignatura | fk | id\_asignatura | Rstricción referencial hacia asignatura |
| FK\_curso\_sala | fk | id\_sala | Rstricción referencial hacia sala |
| FK\_curso\_profesor | fk | id\_profesor | Rstricción referencial hacia profesor |
| UNIQ\_curso\_bimestre\_prof | unique | id\_curso, id\_anyo id\_bimestre, id\_profesor | Restricción de unicidad para garantizar que un profesor dicte un solo curso en un bimestre y año dado |
| UNIQ\_curso\_bimestre\_sala | unique | id\_curso,  id\_anyo, id\_bimestre,  id\_sala | Restricción de unicidad para garantizar que un curso sea dictado en una única sala en un bimestre y año dado |
| alumno | UNIQ\_rut | unique | rut | Restricción de unicidad para el rut |
| matricula | FK\_matricula\_alumno | fk | id\_alumno | Rstricción referencial hacia alumno |
| FK\_matricula\_curso | fk | id\_curso | Restricción referencial hacia curso |
| UNIQ\_alumno\_curso | unique | id\_alumno, id\_curso | Restricción de unicidad para garantizar que un alumno no pueda tomar el mismo curso más de una vez |
| profesor\_asignatura | FK\_profesor\_asignatura\_asignatura | fk | id\_asignatura | Restricción referencial hacia asignatura |
| FK\_profesor\_asignatura\_profesor | fk | id\_profesor | Restricción referencial hacia profesor |
| UNIQ\_profesor\_asignatura | unique | id\_profesor, id\_asignatura | Restricción de unicidad para que el par profesor asignatura sea único |
| sala\_asignatura | Id | fk |  |  |
| Id\_pro | fk |  |  |
| UNIQ\_sala\_asignatura | unique |  | Restricción de unicidad para que el par sala asignatura sea único |

A excepción de las FK de la entidad matrícula, todas las restricciones referenciales son definidas como restricted implicando con ello que no pueden existir registros huérfanos

Adicionalmente, para la restricción de máximo de cursos por bimestre y máximo de alumnos por sala, se han creado los siguientes triggers

|  |  |  |  |
| --- | --- | --- | --- |
| **Entidad** | **Nombre** | **Tipo** | **Descripción** |
| curso | trg\_cursos | Instead of insert | Si los cursos del año y bimestre del curso a insertar superan 20, lanzar excepción, de lo contrario insertar el registro |
| matricula | trg\_matriculas | Instead of insert | Si la cantidad de matrículas asociadas al curso al cual se está matriculando, supera a la capacidad de la sala asociada al curso, lanzar excepción, de lo contrario insertar el registro |

### 2.1.4 CONSULTAS SQL

• Profesores que dictaron UML el tercer bimestre de 2011.

|  |  |
| --- | --- |
|  |  |

• Lista de cursos que ha aprobado el alumno Perez (bimestre, año, profesor y nota).

|  |  |
| --- | --- |
|  |  |

• Profesores que pueden dictar .NET.

|  |  |
| --- | --- |
|  |  |

• Salas de clases disponibles en las que se pueda dictar EJB para el primer bimestre de 2014.

|  |  |
| --- | --- |
|  |  |

• Total de alumnos que han cursado UML en 2011 y 2012.

|  |  |
| --- | --- |
|  |  |

• Total de cursos dictados y alumnos atendidos por el profesor Perez en 2011 y 2012.

|  |  |
| --- | --- |
|  |  |

## 2.2 DESCRIBA POR QUÉ EL ESQUEMA ES 3FN (FORMA NORMAL)

# PARTE 3. PROGRAMACIÓN

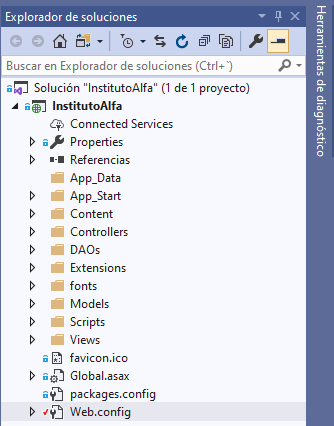
## 3.1 PROGRAME UN PEQUEÑO SISTEMA WEB CON LAS HERRAMIENTAS/TECNOLOGÍAS INDICADAS EN 3.2

3.1.1 y 3.1.2 proyecto en repositorio Google Drive

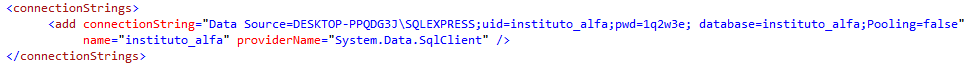
### 3.1.3 INDICACIONES PARA LA INSTALACIÓN

Descargar el archivo InstitutoAlfa.zip desde Google Drive. El archivo corresponde a un comprimido del proyecto .net mvc, por lo que una vez descargado y descomprimido, debería ser posible abrirlo desde visual studio[[1]](#footnote-1).

Una vez abierto en visual studio, editar archivo Web.config



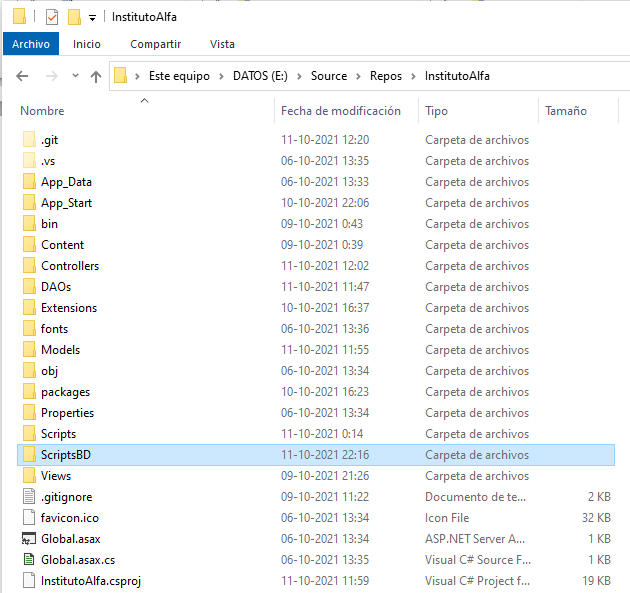
En la sección connectionStrings, añadir la siguiente cadena de conexión



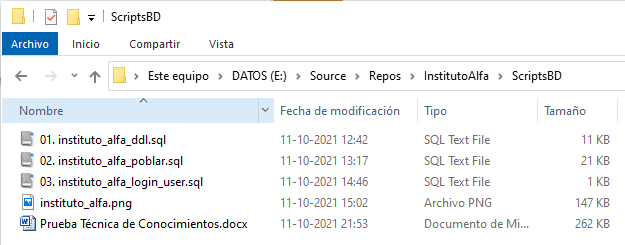
Reemplazar el valor del atributo Data Source por el servidor/instancia del ambiente propio



A continuación abrir la ubicación del proyecto mediante el sistema de archivos, y ubicar el directorio ScriptsBD



Abrir el directorio ScriptsBD



Ejecutar los archivos con extensión. sql en el orden que aparecen. Para ello utilizas SSMS u otra herramienta

## 3.2 EXPLIQUE CÓMO UTILIZÓ O APLICÓ LAS SIGUIENTES HERRAMIENTAS/TECNOLOGÍAS PARA DESARROLLAR EL PROBLEMA

|  |
| --- |
| 3.2.1 SQL SERVER |
| Se utilizó SQL Server en su versión Express 2012 + SSMS 19 para la ejecución de los scripts ddl previamente generados desde una herramienta CASE en donde se realizó el modelo de datos. Además se elaboraron con estas herramientas los scripts para poblar las distintas tablas de la BD y objetos adicionales como triggers. Adicionalmente SQL Server fué el componente responsable del despliegue de la BD durante la etapa de desarrollo |
| 3.2.2 VISUAL STUDIO |
| Se utilizó visual Studio en su versión Community 2019 para la creación y desarrollo del aplicativo, Se creó, mediante el asistente de Visual Studio, un proyecto del tipo .NET MVC, generandose todo lo necesario, estructura del proyecto, librerías y dependencias para empezar con el proyecto |
| 3.2.3 ASP.NET CON C# |
| Se utilizó C# como lenguaje de backend, tanto para construir las clases de negocio (modelo) como para la capa de persistencia (DAO), asi como la capa controlador Controllers. No se utilizaron herramientas ORM, por el poco conocimiento que se tiene con estas dentro del ecosistema de .NET, y en su lugar, se optó por contruir manualmente la capa de persistencia mediante el patrón DAO y consultas SQL utilizando SQLClient |
| 3.2.4 MVC |
| MVC se utilizó para separar las responsabilidades en las distintas operaciones con las entidades del negocio y el flujo entre las distintas páginas y/o acciones. Por ejemplo: Para los alumnos  Controladores:  AlumnoController:  Index, View, New  Vista  Alumno  Index  View  etc  Modelo  Alumno  AlumnoDAO |
| 3.2.5 WEBFORMS |
| No se utilizó, ya que es una tecnología o patrón alternativo a MVC |
| 3.2.6 BOOTSTRAP 4 |
| Los proyectos .NET MVC utilizan Bootstrap para los estilos y sistema de grillas, se utilizaron estas características para el diseño de formularios y estilo en los botones |
| 3.2.7 CSS3 |
| No se utilizó CSS ya que no se customizaron los estilos de las vistas |
| 3.2.8 HTML5 |
| Se utilizó para marcar las etiquetas en formulario de ingreso de alumno, en específico para declarar los campos como obligatorios y el tipo de campo date para el dato nacimiento del alumno |
| 3.2.9 JQUERY |
| Se utilizó el plugin jquery Datatables para agregar funcionalidad y usabilidad a las tablas, como paginación, filtro, ordenamiento etc |
| 3.2.10 AJAX |
| No se utilizó, ya que el envío de datos se hizo mediante acciones del formulario y la lectura de datos se hizo mediante Razor |
| 3.2.11 OTRO |
| Se utilizó la extensión FlashMessaggeExtension para el envío de mensajes en las vistas como resultado de las operaciones del mantenedor |

1. El proyecto fue creado y desarrollado con visual studio community 2019, por lo que se recomienda usar el uso de un ide similar o superior [↑](#footnote-ref-1)