

Universidad Autónoma de Nuevo León
Facultad de Ciencias Físico Matemáticas

Avance

Modelos de

Administración de datos

Grupo: 055

Maestro: Juan Alejandro Villarreal Mojica

Integrantes del equipo:

Andrea Berenice Reyna Gutiérrez

1873789

José Armando Hernández Santander

2034261

Tabla de contenido

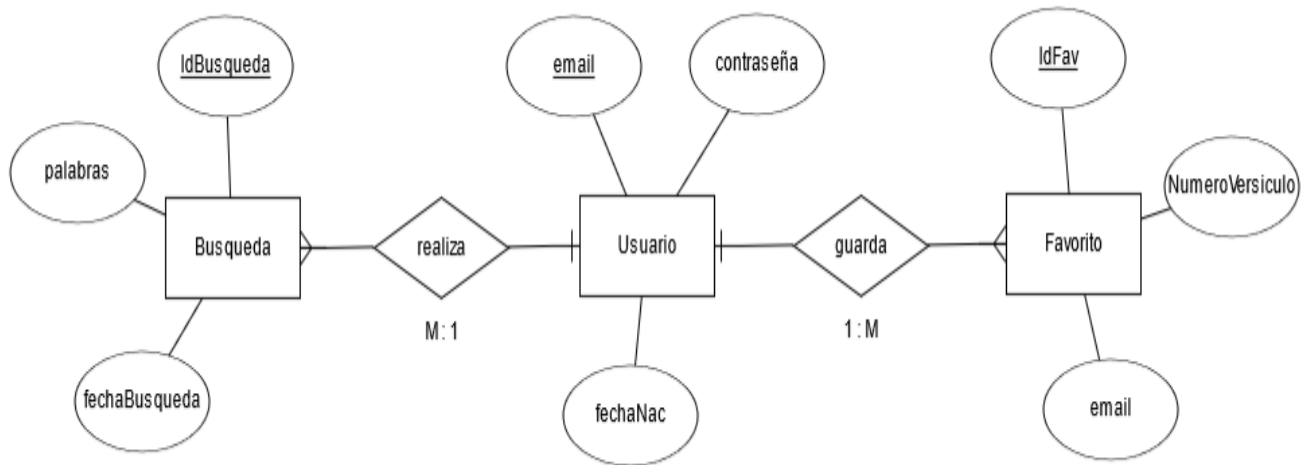
Modelo Entidad-Relación.....	3
Diagrama Entidad-Relación.....	4
Mapeo de datos:.....	5
Normalización:.....	6
Modelo Relacional:.....	9
Creación de tablas en SQL:.....	10
Diagrama:.....	13
Diccionario de Datos:	14
Operaciones de álgebra relacional:	18
Pantallas del proyecto	19

Modelo Entidad-Relación

Entidad:	Atributos:
Usuario	<ul style="list-style-type: none"> • email • contraseña1 • contraseña2 • nombreCompleto(nombre, apellido1, apellido2) • fechaNac • genero • estatus • fechahoraAlta • fechahoraBaja
Favorito	<ul style="list-style-type: none"> • IdFav • IdIdioma • IdTestamento • IdVersion • IdLibro • NumeroCapitulo • NumeroVersiculo • email
Búsqueda	<ul style="list-style-type: none"> • IdBusqueda • Palabras • Idioma • Version • Testamento • Libro • fechaBusqueda • HuboResultados • email

Entidad:	Relación:	Entidad:	Cardinalidad:
Usuario	guarda	Favorito	1-M
Usuario	realiza	Búsqueda	1-M

Diagrama Entidad-Relación



Restricciones:

- Todas las contraseñas deben de tener al menos 8 caracteres y debe de incluir una mayúscula, una minúscula y un carácter especial.
Carácter especial = cualquier símbolo generado por el teclado que no sea letra ni número, por ejemplo: (¡"#\$%&/='?¡¿::,.-_+*{}|~).
- El estatus solo puede ser Activo, Inhabilitado o Eliminado.
- Los favoritos y las búsquedas solo pueden agregarse o eliminarse.
- Se puede eliminar una o todas las búsquedas.

Mapeo de datos:

1.- Las entidades pasan a ser tablas.

2.- Los atributos pasan a ser columnas.

3.- Para las relaciones de 1 a Muchos se agrega una FK en la tabla de Muchos, que corresponda a la PK de la tabla del 1.

Usuarios

Email PK	Contraseña1	Contraseña2	(Nombre Completo)			fechaNac	genero	estatus
			nombre	Apellido1	Apellido2			

fechahoraAlta	fechahoraBaja

Favoritos

Idfav PK	IdIdioma	IdTestamento	IdVersion	IdLibro	NumeroCap	NumeroVers	EmailUsuario FK

Búsqueda

IdBusqueda PK	palabras	Idioma	Version	Testamento	Libro	fechaBusqueda

HuboResultados	EmailUsuario FK

Normalización:

1FN – Datos atómicos, eliminar datos repetitivos

La columna compuesta “nombre” de la tabla Usuarios se divide en datos atómicos

Usuarios

Email PK	nombre	Apellido1	Apellido2	fechaNac	genero	estatus

Las columnas “contraseña1” y “contraseña2” contienen datos repetitivos por lo que se crea una tabla nueva de las contraseñas de los usuarios

ContraseñasdeUsuarios

IdContra PK	Email FK	tipo	contraseña

Tanto la columna “fechahoraAlta” como “fechahoraBaja” contienen datos repetitivos por lo que se crea una tabla nueva de las fechas de gestión de los usuarios

FechasGestiónUsuarios

IdFecha PK	Email FK	tipo	fecha

Favoritos

Idfav PK	IdIdioma	IdTestamento	IdVersion	IdLibro	NumeroCap	NumeroVers	EmailUsuario FK

Búsqueda

IdBusqueda PK	palabras	Idioma	Version	Testamento	Libro	fechaBusqueda

HuboResultados	EmailUsuario FK

2FN-Eliminar datos redundantes

Al llenar datos de la tabla Usuarios, varias filas tendrían los mismos valores en genero y estatus por lo cual se crean tablas independientes

Usuarios

Email PK	nombre	Apellido1	Apellido2	fechaNac	IdGenero FK	IdEstatus FK

Géneros

IdGenero PK	Genero

Estatus

IdEstatus PK	estatus

ContraseñasdeUsuarios

IdContra PK	Email FK	tipo	contraseña

FechasGestiónUsuarios

IdFecha PK	Email FK	tipo	fecha

Favoritos

Idfav PK	IdIdioma	IdTestamento	IdVersion	IdLibro	NumeroCap	NumeroVers	EmailUsuario FK

Búsqueda

IdBusqueda PK	palabras	Idioma	Version	Testamento	Libro	fechaBusqueda

HuboResultados	EmailUsuario FK

3FN- Eliminar datos no dependientes(incoherentes)

Las tablas ya no necesitan más cambios por lo que pasan directamente a la tercera forma normal

Usuarios

Email PK	nombre	Apellido1	Apellido2	fechaNac	IdGenero FK	IdEstatus FK

Géneros

IdGenero PK	Genero

Estatus

IdEstatus PK	estatus

ContraseñasdeUsuarios

IdContra PK	Email FK	tipo	contraseña

FechasGestiónUsuarios

IdFecha PK	Email FK	tipo	fecha

Favoritos

Idfav PK	IdIdioma	IdTestamento	IdVersion	IdLibro	NumeroCap	NumeroVers	EmailUsuario FK

Búsqueda

IdBusqueda PK	palabras	Idioma	Version	Testamento	Libro	fechaBusqueda

HuboResultados	EmailUsuario FK

Modelo Relacional:

Usuarios

Email PK	nombre	Apellido1	Apellido2	fechaNac	IdGenero FK	IdEstatus FK

Géneros

IdGenero PK	Genero

Estatus

IdEstatus PK	estatus

ContraseñasdeUsuarios

IdContra PK	Email FK	tipo	contraseña

FechasGestiónUsuarios

IdFecha PK	Email FK	tipo	fecha

Favoritos

Idfav PK	IdIdioma	IdTestamento	IdVersion	IdLibro	NumeroCap	NumeroVers	EmailUsuario FK

Búsqueda

IdBusqueda PK	palabras	Idioma	Version	Testamento	Libro	fechaBusqueda

HuboResultados	EmailUsuario FK

Creación de tablas en SQL:

```
1 CREATE DATABASE DB_Pray;
2
3 use DB_Pray
4
5 if OBJECT_ID('Generos') is not null
6     drop table DB_Pray.dbo.Generos;
7
8 CREATE TABLE DB_Pray.dbo.Generos(
9     idGenero        SMALLINT NOT NULL IDENTITY(1, 1),
10    genero           VARCHAR(15) NOT NULL,
11    CONSTRAINT PK_Gen
12        PRIMARY KEY(idGenero)
13);
14
15 --insert generos
16 insert into DB_Pray.dbo.Generos(genero)
17     values('Masculino'),
18         ('Femenino');
19
20     select *from Generos;
21     -----fin
22
23 if OBJECT_ID('Estatus') is not null
24     drop table Estatus;
```

```

25
26 CREATE TABLE DB_Prayer.dbo.Estatus(
27     idEstatus      SMALLINT NOT NULL IDENTITY(1, 1),
28     estatus        VARCHAR(15) NOT NULL,
29     CONSTRAINT PK_Est
30         PRIMARY KEY(idEstatus)
31 );
32
33 insert into DB_Prayer.dbo.Estatus(estatus)
34     values('Activo'),
35           ('Inhabilitado'),
36           ('Eliminado')
37
38     select *from Estatus;
39
40 if OBJECT_ID('Usuarios') is not null
41     drop table Usuarios;
42
43 CREATE TABLE DB_Prayer.dbo.Usuarios(
44     email          VARCHAR(40) NOT NULL,
45     nombre         VARCHAR(20),
46     apellido1      VARCHAR(20),
47     apellido2      VARCHAR(20),
48     fechaNac       DATE,
49     idGenero       SMALLINT,
50     IdEstatus      SMALLINT,
51     CONSTRAINT PK_User
52         PRIMARY KEY(email),
53     CONSTRAINT FK_User_Gen
54         FOREIGN KEY(idGenero)
55             REFERENCES DB_Prayer.dbo.Generos (idGenero),
56     CONSTRAINT FK_User_Est
57         FOREIGN KEY(IdEstatus)
58             REFERENCES DB_Prayer.dbo.Estatus(idEstatus)
59 );
60
61
62 if OBJECT_ID('ContrasenadesUsuarios') is not null
63     drop table ContrasenadesUsuarios;
64
65 CREATE TABLE DB_Prayer.dbo.ContrasenadesUsuarios(
66     idContra       SMALLINT NOT NULL IDENTITY(1, 1),
67     email          VARCHAR(40),
68     tipo           VARCHAR(15),
69     contrasena     VARCHAR(40),

```

```

70         CONSTRAINT PK_Contra
71             PRIMARY KEY(idContra),
72         CONSTRAINT FK_Contra_User
73             FOREIGN KEY(email)
74                 REFERENCES DB_Pray.dbo.Usuarios(email)
75 );
76
77 if OBJECT_ID('FechasGestionUsuarios') is not null
78     drop table FechasGestionUsuarios;
79
80 CREATE TABLE DB_Pray.dbo.FechasGestionUsuarios(
81     idFecha          SMALLINT NOT NULL IDENTITY(1, 1),
82     email             VARCHAR(40),
83     tipo              VARCHAR(15),
84     fecha             DATETIME,
85     CONSTRAINT PK_FechasGes
86         PRIMARY KEY(idFecha),
87     CONSTRAINT FK_FechasGes_User
88         FOREIGN KEY(email)
89             REFERENCES DB_Pray.dbo.Usuarios(email)
90 );
91
92 if OBJECT_ID('Favoritos') is not null
93     drop table Favoritos;
94
95 CREATE TABLE DB_Pray.dbo.Favoritos(
96     idFav            SMALLINT NOT NULL IDENTITY(1, 1),
97     Id_Idioma         SMALLINT NOT NULL,
98     Id_Testamento    SMALLINT NOT NULL,
99     Id_Version         SMALLINT NOT NULL,
100    Id_Libro           SMALLINT NOT NULL,
101    NumeroCap          TINYINT NOT NULL,
102    NumeroVers         TINYINT NOT NULL,
103    emailUsuario       VARCHAR(40),
104    CONSTRAINT PK_Favoritos
105        PRIMARY KEY(idFav),
106    CONSTRAINT FK_Favoritos_User
107        FOREIGN KEY(emailUsuario)
108            REFERENCES DB_Pray.dbo.Usuarios(email)
109 );
110
111
112 if OBJECT_ID('Busquedas') is not null
113     drop table Busquedas;
114

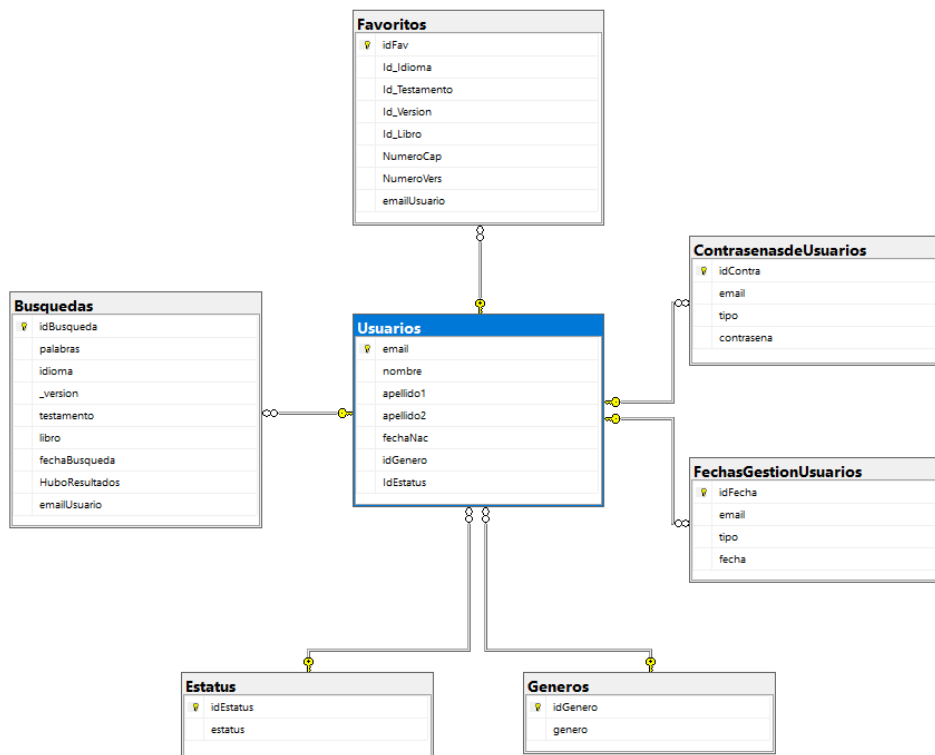
```

```

115 CREATE TABLE DB_Pray.dbo.Busquedas (
116     idBusqueda          SMALLINT NOT NULL IDENTITY(1, 1),
117     palabras             VARCHAR(30) NOT NULL,
118     idioma              SMALLINT NOT NULL,
119     _version             SMALLINT NOT NULL,
120     testamento          SMALLINT,
121     libro                SMALLINT,
122     fechaBusqueda        DATETIME NOT NULL,
123     HuboResultados      BIT NOT NULL,
124     emailUsuario         VARCHAR(40),
125     CONSTRAINT PK_Busquedas
126         PRIMARY KEY(idBusqueda),
127     CONSTRAINT FK_Busquedas_User
128         FOREIGN KEY(emailUsuario)
129             REFERENCES DB_Pray.dbo.Usuarios(email)
130 );

```

Diagrama:



Diccionario de Datos:

Generos			
Nombre del campo	Descripción	Tipo de dato	Longitud
idGenero	índice para identificar el género	SMALLINT	
genero	Nombre del género: "Masculino" o "Femenino"	VARCHAR	15

Estatus			
Nombre del campo	Descripción	Tipo de dato	Longitud
idEstatus	Índice para identificar el estatus	SMALLINT	
estatus	Nombre del estatus: "Activo", "Inhabilitado" o "Eliminado"	VARCHAR	15

Usuarios			
Nombre del campo	Descripción	Tipo de dato	Longitud
email	Correo electrónico del usuario	VARCHAR	40
nombre	Nombre del usuario	VARCHAR	20
apellido1	Apellido paterno del usuario	VARCHAR	20
apellido2	Apellido materno del usuario	VARCHAR	20
fechaNac	Fecha de nacimiento del usuario	DATE	
idGenero	Índice para identificar el género del usuario en la tabla "Generos"	SMALLINT	
idEstatus	Índice para identificar el	SMALLINT	

	estatus del usuario en la tabla "Estatus"		
--	---	--	--

ContrasenadeUsuarios			
Nombre del campo	Descripción	Tipo	Longitud
idContra	Índice para identificar la contraseña	SMALLINT	
email	Correo electrónico del usuario dueño de la contraseña	VARCHAR	40
tipo	Tipo de contraseña: "Actual" o "Anterior"	VARCHAR	15
contrasena	Contraseña del usuario	VARCHAR	40

FechasGestionUsuarios			
Nombre del campo	Descripción	Tipo	Longitud
idFecha	Índice para identificar la fecha de gestión	SMALLINT	
email	Correo electrónico del usuario de quien es la fecha de gestión	VARCHAR	40
tipo	Tipo de gestión realizada: "Registro" o "Baja"	VARCHAR	15
fecha	Fecha en que se realizó la gestión del usuario	DATETIME	

Favoritos			
Nombre del campo	Descripción	Tipo	Longitud
idFav	Índice para identificar el favorito	SMALLINT	
Id_Idioma	Id del idioma del pasaje favorito	SMALLINT	
Id_Testamento	Id del testamento donde se encuentra el pasaje favorito	SMALLINT	
Id_Version	Id de la versión de la biblia del pasaje favorito	SMALLINT	
Id_Libro	Id del libro donde se encuentra el pasaje favorito	SMALLINT	
NumeroCap	Número de capitulo donde está el favorito	TINYINT	
NumeroVers	Número del versículo favorito	TINYINT	
emailUsuario	Correo electrónico del usuario dueño del favorito	VARCHAR	40

Busquedas			
Nombre del campo	Descripción	Tipo	Longitud
idBusqueda	Índice para identificar la busqueda	SMALLINT	
palabras	La o las palabras ingresadas en la búsqueda del usuario	VARCHAR	30
Idioma	Id del idioma donde se buscó	SMALLINT	
_version	Id de la versión donde se buscó	SMALLINT	
testamento	Id del testamento donde se buscó (este valor solo se	SMALLINT	

	guarda si se especificó)		
libro	Id del libro donde se buscó (este valor solo se guarda si se especificó)	SMALLINT	
fechaBusqueda	La fecha y hora del momento que se realizó la búsqueda	DATETIME	
HuboResultados	Si la búsqueda arrojó resultados.	BIT	
emailUsuario	Correo electrónico del usuario que realizó la búsqueda	VARCHAR	40

Operaciones de álgebra relacional:

Obtener los testamentos con sus libros:

π Libros.libro, Testamentos.nombre

$(\sigma \text{ Testamentos.IDtestamento} = \text{Libros.testamentID} (\text{Libros} \times \text{Testamentos}))$

Obtener los usuarios con sus contraseñas actuales:

π Usuarios.nombre, ContrasenadeUsuarios.contrasena

$(\sigma \text{ ContrasenadeUsuarios.email} = \text{Usuarios.email}$

$\wedge \text{ContrasenadeUsuarios.tipo} = \text{actual} (\text{ContrasenadeUsuarios} \times \text{Usuarios}))$

Obtener la fecha en que se dio de alta un usuario:

π FechasGestionUsuarios.fecha

$(\sigma \text{ FechasGestionUsuarios.email} = \text{Usuarios.email} (\text{FechasGestionUsuarios} \times \text{Usuarios}))$

Obtener Usuarios con sus libros

π Libros.libro, Usuarios.nombre, Apellido1, Apellido2, fechaNac

$(\sigma \text{ Usuarios.email} = \text{Libros.libro} = (\text{Libros} \times \text{Usuarios}))$


Obtener el capítulo con su usuario

π Usuarios.nombre, Capitulo.nombre

$\sigma \text{ Capitulo.IDcapituloFK} = \text{Usuarios.email} (\text{Usuarios} \times \text{Capitulos})$

Pantallas del proyecto

Inicio Sesión ✕



Correo electrónico:

Contraseña:

¿No tiene cuenta?


Preferencias — □ ✕

Cerrar Sesión

Establecer preferencias:

Idioma:

Versión:

 Agregar usuario Minimizar Maximizar Cerrar

Correo electrónico:

Contraseña:

Nombre:

Apellido Paterno:

Apellido Materno:

Fecha de nacimiento: martes , 21 de noviembre de 20. ▾

Género:

Aceptar **Cancelar**

 Editar y baja de Usuario Minimizar Maximizar Cerrar

[Cerrar Sesión](#) [Busqueda](#) [Consultar Biblia](#) [Favoritos](#) [Historial](#) [Preferencias](#)

Correo electrónico:

Contraseña: **Cambiar contraseña**

Nombre:


Apellido Paterno:

Apellido Materno:

Fecha de nacimiento: martes , 21 de noviembre de 20. ▾

Género:


Guardar edición **Darse de baja** **Cancelar**

 CambioContraseña

Contraseña Anterior

Contraseña Actual

Nueva Contraseña

 Consultar Biblia

[Cerrar Sesión](#) [Favoritos](#) [Historial](#) [Busqueda](#) [Gestion Usuario](#) [Preferencias](#)

Elija el lugar donde quiere buscar:

Testamento: Número de Versículo (opcional):

Libro:

Número de Capítulo:

