



Instituto Tecnológico de Costa Rica

Escuela de Computación

IC-4302 Bases de Datos II

Grupo #20

Proyecto

II Semestre

Profesora:

María Auxiliadora Mora Cross

Estudiantes:

Pablo Sandí Sánchez 2021120523

Axel Solano Bejarano 2021453483

Fecha de entrega:

23 de setiembre del 2022

Introducción:

Descripción del problema: Se solicita el desarrollo de una aplicación web que logre conectarse a bases de datos para almacenar información para imitar de esta forma los requerimientos que se pueden llegar a solicitar en un sistema de información de la vida real.

Propuesta de solución: Para la creación de la aplicación web el equipo de desarrollo utilizara como ejemplo el video juego llamado “Fortnite” en el cual podemos encontrar distintas entidades que podremos almacenar en las bases de datos.

Descripción del sistema: El sistema permitirá agregar, editar, eliminar datos representativos del video juego “Fortnite”, además permitirá acceder a ellos con su nombre, imágenes y descripción todos mediante una interfaz web.

Objetivos y alcance del sistema: Se buscará incluir las características más representativas del video juego en su última versión basándonos en la popularidad de las compras realizadas por los usuarios mientras dejamos por aparte otros artículos que se han descontinuado del juego conforme se han realizado actualizaciones en el mismo.

Casos de uso:

Caso de Uso: Agregar datos	
Actor Primario:	Usuario.
Actor Secundario:	Sistema.
Pre-Condiciones:	El usuario cuenta con permisos para agregar datos.
Post-Condiciones:	El sistema agrega datos a la base de datos.
Flujo Principal:	1.El usuario ingresa a la aplicación. 2.El usuario selecciona la opción de “agregar datos”. 3.El usuario especifica el tipo de dato que desea agregar (aspecto, arma, ubicación, etc.). 4.El sistema verifica que no exista duplicación de datos. 5.El sistema muestra el mensaje “dato agregado con éxito”.
Flujos Alternos:	4.(b) El sistema indica que el dato ya existe en la base de datos.

Caso de Uso: Modificar datos	
Actor Primario:	Usuario.
Actor Secundario:	Sistema.
Pre-Condiciones:	El usuario cuenta con permisos para modificar datos.
Post-Condiciones:	El sistema modifica datos en la base de datos.
Flujo Principal:	1.El usuario ingresa a la aplicación. 2.El usuario selecciona la opción de “modificar datos”. 3.El usuario especifica el tipo de dato que desea modificar (aspecto, arma, ubicación, etc.). 4.El sistema verifica que el dato exista en la base de datos. 5.El sistema muestra el mensaje “dato modificado con éxito”.
Flujos Alternos:	4.(b) El sistema indica que el dato no existe en la base de datos.

Caso de Uso: Eliminar datos	
Actor Primario:	Usuario.
Actor Secundario:	Sistema.
Pre-Condiciones:	El usuario cuenta con permisos para eliminar datos.
Post-Condiciones:	El sistema elimina datos en la base de datos.
Flujo Principal:	<ol style="list-style-type: none"> 1.El usuario ingresa a la aplicación. 2.El usuario selecciona la opción de “eliminar datos”. 3.El usuario especifica el tipo de dato que desea eliminar (aspecto, arma, ubicación, etc.). 4.El sistema verifica que el dato exista en la base de datos. 5.El sistema muestra el mensaje “dato eliminado con éxito”.
Flujos Alternos:	4.(b) El sistema indica que el dato no existe en la base de datos.

Caso de Uso: Acceder a datos	
Actor Primario:	Usuario.
Actor Secundario:	Sistema.
Pre-Condiciones:	El usuario cuenta con permisos para acceder a datos.
Post-Condiciones:	El usuario recibe los datos solicitados.
Flujo Principal:	<ol style="list-style-type: none"> 1.El usuario ingresa a la aplicación. 2.El usuario selecciona la opción de “acceder a datos”. 3.El usuario especifica el tipo de dato al que desea acceder (aspecto, arma, ubicación, etc.). 4.El sistema verifica que el dato exista en la base de datos. 5.El sistema muestra los datos solicitados.
Flujos Alternos:	4.(b) El sistema indica que el dato no existe en la base de datos.

Diseño del sistema:

Arquitectura del sistema:

La aplicación contará con una página de inicio donde podrá observar tablas con los diferentes datos almacenados y los diferentes botones de acción. Por medio de los botones se podrá insertar, editar y eliminar datos.

Modelo de clases:

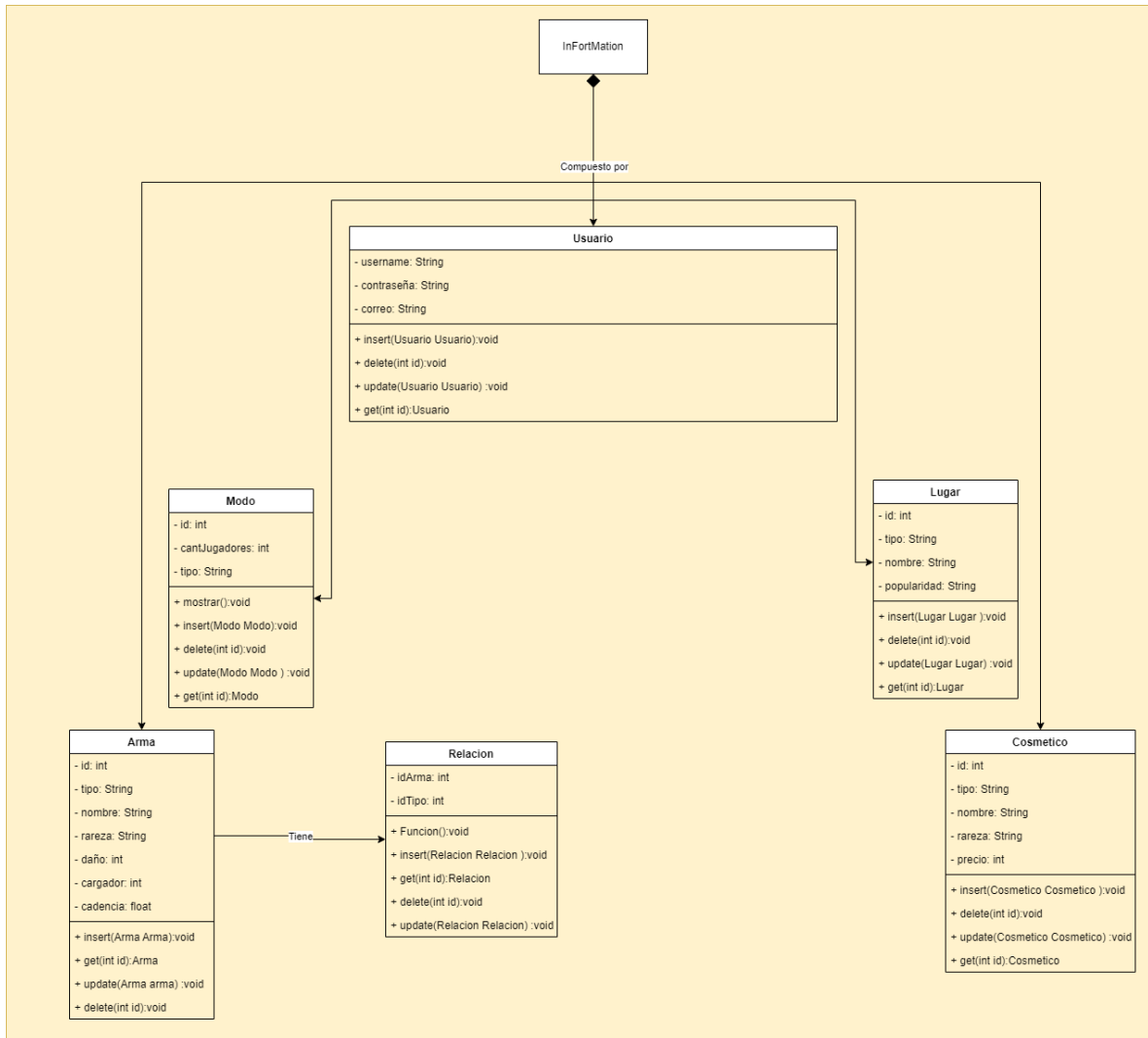
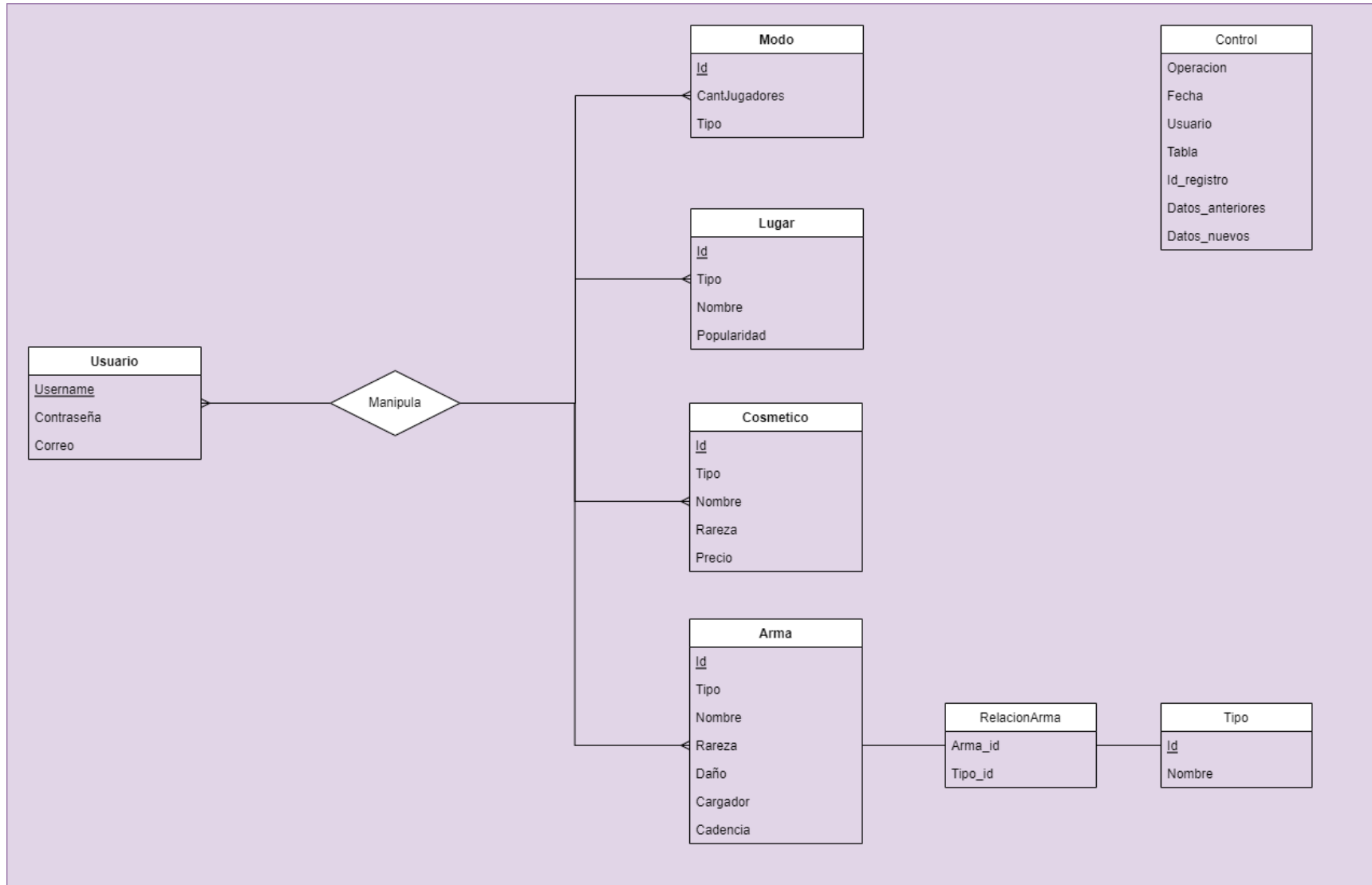


Diagrama entidad-relación:



Diccionario de datos:

Nombre del elemento	Descripción	Tipo de dato	Null?	PK
Username	Hace referencia al nombre de usuario del usuario.	Varchar(100)	No	Sí
Contraseña	La contraseña que usa el usuario para iniciar sesión.	Varchar(100)	No	No
Correo	El correo electrónico asociado a una cuenta de usuario.	Varchar(100)	No	No
ID	Un valor auto incremental único representativo de cada registro.	Int	No	Sí
Tipo	Representa el tipo de objeto. Cambia dependiendo de la entidad.	Varchar(100)	No	No
Nombre	Nombre representativo de cada registro.	Varchar(100)	No	No
Rareza	Representa el valor de los registros.	Varchar(100)	No	No
Daño	Representa el daño que causan las armas.	Int	No	No
Cargador	Representa el tamaño del cargador de las armas.	Int	No	No
Cadencia	Representa la velocidad de disparo de las armas.	Float	No	No
Precio	Representa el valor dentro del juego de cada registro utilizando una moneda virtual.	Int	No	No

Popularidad	Representa la cantidad de jugadores que visitan el lugar específico.	Varchar(100)	No	No
Cantidad de jugadores	Representa la cantidad máxima de jugadores que pueden jugar juntos un modo de juego específico.	Int	No	No
Operacion	Representa el tipo de operación realizada (INSERT, UPDATE, DELETE)	Varchar(100)	No	No
Fecha	Representa la fecha y hora de la operación.	Date	No	No
Usuario	Representa el usuario el cual realiza la operación.	Varchar(100)	No	No
Tabla	Representa el nombre de la tabla en la cual se realiza la operación.	Varchar(100)	No	No
Id_registro	Representa el id del dato a modificar, insertar o eliminar.	Int	No	No
Datos_anteriores	Representa los datos guardados antes de la operación.	Varchar(100)	No	No
Datos_nuevos	Representa los datos nuevos ingresados en la operación.	Varchar(100)	No	No

Entidades Oracle:

En la base de datos de Oracle vamos a almacenar las entidades mencionadas en el diagrama entidad-relación las cuales son: usuario, cosméticos, armas y lugares.

Procedimiento almacenado:

El procedimiento almacenado tendrá la función de unir los datos de las armas y relacionarlos con la tabla tipo la cual definirá el tipo del arma.