

Lenguajes de Base de Datos

SC-504

Aplicación tecnológica de venta de videojuegos
TuJuego.

Integrantes:

Hernandez Vargas Héctor Jesús

Rivero Reyes Alejhandro

Sanchez Piedra José Andrés

Zamora Conejo Josué Alexander

II CUATRIMESTRE

26 agosto 2024

Prof.

RANDALL ALONSO LEITON JIMENEZ

1. Indicé

1	PLANTEAMIENTO DE LA INVESTIGACIÓN	2
1.1	Introducción.....	2
1.2	Problema de investigación	3
1.2.1	<i>Justificación del Problema</i>	3
1.3	Objetivos	4
1.3.1	Objetivo general.....	4
1.3.2	Objetivos específicos	4
2	DESARROLLO	5
2.1	Requerimientos funcionales	5
2.2	Justificación de la interfaz	6
2.3	Evidencias de creación de esquema	7
2.4	Esquema entidad relación	13
2.5	Propuesta de la interfaz	14
2.6	Conexión JDBC con Base de Datos	19
2.7	Diccionario de datos	19
3	RESULTADOS	33
4	CONCLUSIONES Y RECOMENDACIONES	48
5	BIBLIOGRAFÍA	49

1 PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1 *Introducción*

Las tiendas de videojuegos en línea han emergido como plataformas esenciales que conectan a los jugadores con una vasta gama de títulos, desde los clásicos hasta los últimos lanzamientos. Sin embargo, para que estas tiendas funcionen de manera eficiente y satisfagan las expectativas de los usuarios, es crucial contar con una gestión de datos sólida y una experiencia de usuario intuitiva. Este proyecto se propone abordar estos desafíos mediante el desarrollo de un software integral para una tienda de videojuegos en línea, utilizando bases de datos en lenguaje Oracle para manejar datos de manera eficiente y segura.

El corazón de este proyecto radica en la creación de una solución tecnológica que no solo sea robusta, sino también amigable para el usuario. Esto implica el diseño de una interfaz gráfica que sea atractiva y fácil de navegar, asegurando que los usuarios puedan interactuar con la tienda de manera intuitiva. A través de la implementación de tecnologías web modernas y el uso del Spring Framework, se busca crear una arquitectura que facilite la conexión entre la interfaz de usuario y la lógica del servidor, permitiendo una experiencia fluida y sin interrupciones.

Además de la interfaz, un componente crítico del proyecto es el desarrollo de un modelo de datos que responda a las demandas operativas de una tienda de videojuegos en línea. Utilizando bases de datos, se estructuran los datos de manera que se pueda manejar un alto volumen de usuarios y transacciones sin comprometer el rendimiento del sistema. La seguridad de los datos es otro pilar fundamental, por lo que se establecerán conexiones seguras entre la aplicación y la base de datos, utilizando Java para garantizar la integridad y protección de la información.

Este proyecto no solo se enfoca en la creación de un entorno funcional y eficiente para los usuarios finales, sino que también busca ofrecer una gestión de datos que sea robusta y capaz de

soportar las exigencias actuales del mercado de videojuegos. Al implementar funcionalidades clave como la gestión de usuarios y juegos, y el cálculo eficiente de facturas, se espera mejorar significativamente la experiencia del usuario y optimizar las operaciones de la tienda.

1.2 Problema de investigación

Gracias a la apertura del comercio electrónico en los últimos años y como este se vuelve el más utilizado, hace importante el desarrollo de nuevos proyectos orientados a la venta electrónica, dicho cambio crea la oportunidad de investigar la mejor manera de realizar los prototipos, para que estos cuenten con interfaces amigables y además un diseño robusto para el manejo de los datos.

1.2.1 Justificación del Problema

El comercio electrónico se ha convertido en una parte esencial de la economía moderna, además de un sector competitivo donde la calidad de experiencia y el manejo de datos son importantes para el éxito. Las interfaces intuitivas y amigables pueden reducir la dificultad en la navegación y la compra, aumentando la probabilidad de transacciones exitosas para el negocio. Por otro lado, un diseño robusto para el manejo de datos asegura que las plataformas puedan soportar grandes volúmenes de tráfico y transacciones.

El problema planteado es de gran importancia ya que aborda la intersección entre el diseño de interfaces amigables y la robustez en el manejo de datos, pilares cruciales en el diseño de prototipos exitosos en el comercio electrónico. Además, el estudio no solo contribuirá al conocimiento académico, sino que también ofrecerá soluciones prácticas que pueden ser aplicadas en la industria electrónica.

1.3 *Objetivos*

1.3.1 *Objetivo general*

Desarrollar una solución tecnológica intuitiva para usuarios de videojuegos, que incluye la codificación de una interfaz gráfica amigable y la implementación de una estructura de datos robusta diseñada para manejar eficientemente las consultas y el gran volumen de datos.

1.3.2 *Objetivos específicos*

Objetivo 1

Crear un diseño de interfaz de usuario que sea visualmente atractivo y fácil de navegar, utilizando HTML y CSS.

Objetivo 2

Implementar una arquitectura web que conecte la interfaz gráfica con la lógica del servidor a través de Spring Framework.

Objetivo 3

Crear un modelo de datos robusto que refleje las necesidades de la aplicación y soporte la carga esperada de usuarios y transacciones.

Objetivo 4

Configurar y establecer una conexión segura entre la aplicación y la base de datos utilizando java y Oracle.

2 DESARROLLO

2.1 Requerimientos funcionales

En esta sección se encuentran los requerimientos funcionales correspondientes a la interfaz gráfica y estructura de base de datos.

1. El sistema deberá realizar el inicio de sesión para permitir el uso de la aplicación.
2. El sistema permite crear, leer, modificar y borrar juegos.
3. El sistema permite crear, leer, modificar y borrar usuarios.
4. El sistema permite crear, leer, modificar y borrar categorías.
5. El sistema deberá realizar el cálculo total de la factura.
6. El sistema permite enviar comentarios sobre la plataforma
7. El sistema permite hacer búsquedas por medio de filtro de categoría.
8. El sistema permite agregar productos al carrito y gestionar los productos agregados.
9. El sistema mostrará los productos que se encuentren disponibles o activos en el menú.
10. El sistema debe reflejar la manipulación o cambios realizados en los datos en el gestor de datos seleccionado.

2.2 Justificación de la interfaz

Spring framework:

Java spring framework es un modelo integral de programación y configuración para aplicaciones basadas en java, esta herramienta facilita el desarrollo de una aplicación con spring framework ya que cuenta con las siguientes funcionalidades, (Spring Framework, 2005):

-Mayor modularidad y flexibilidad: Implementa el IoC que se encarga de injectar las dependencias entre objetos, esto permite escribir un código más limpio y fácil.

-Programación orientada a aspectos (AOP): deriva en un código más limpio, reutilizable y modular.

-Abstracción de datos: simplifica las conexiones, consultas y transacciones a diferentes fuentes de datos.

-Integración con otras tecnologías: cuenta con una amplia gama de tecnologías para ampliar sus funcionalidades, como el desarrollo web.

HTML y CSS:

El HTML es un lenguaje estándar de programación que define el contenido de la página web y el css es utilizado junto al HTML para definir el diseño y la apariencia de la página, como los colores, fuentes, layout, posiciones, entre otros.

Oracle Database:

Es un conjunto organizado de datos o información estructurada, que generalmente se almacena electrónicamente en un sistema informático. Un DBMS generalmente administra una base de datos (Oracle España, n.d), algunas de sus características son:

-Tiene la capacidad de manejar grandes cantidades de volúmenes de datos de manera eficiente.

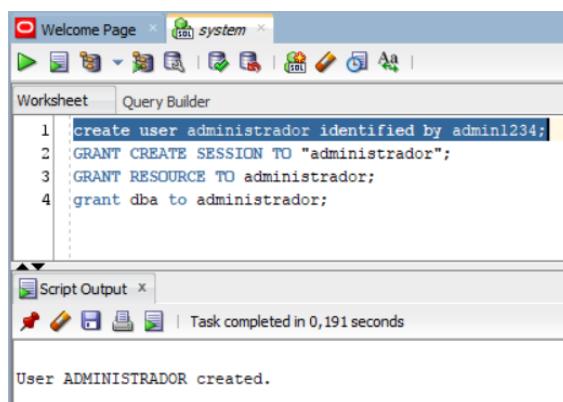
-Proporciona una amplia gama de seguridad para proteger los datos sensibles.

-Cuenta con una amplia compatibilidad con diferentes plataformas.

-Ofrece tiempos de respuesta rápidos y eficientes.

2.3 Evidencias de creación de esquema

En esta sección se presenta la evidencia de la creación del usuario, base de datos y tablas correspondientes para el prototipo.

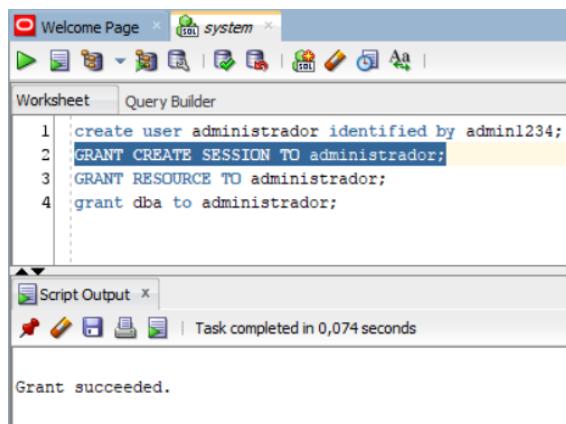


The screenshot shows the Oracle SQL Developer interface. In the 'Worksheet' tab, the following SQL script is run:

```
1 create user administrador identified by admin1234;
2 GRANT CREATE SESSION TO "administrador";
3 GRANT RESOURCE TO administrador;
4 grant dba to administrador;
```

In the 'Script Output' tab, the message 'User ADMINISTRADOR created.' is displayed, indicating the successful creation of the user.

Imagen 1. Creación de usuario y contraseña



The screenshot shows the Oracle SQL Developer interface. In the 'Worksheet' tab, the following SQL script is run:

```
1 create user administrador identified by admin1234;
2 GRANT CREATE SESSION TO administrador;
3 GRANT RESOURCE TO administrador;
4 grant dba to administrador;
```

In the 'Script Output' tab, the message 'Grant succeeded.' is displayed, indicating the successful granting of the 'CREATE SESSION' privilege.

Imagen 2. Permiso para inicio de sesión

The screenshot shows the Oracle SQL Developer interface. In the Worksheet tab, the following SQL code is executed:

```

1 create user administrador identified by admin1234;
2 GRANT CREATE SESSION TO administrador;
3 GRANT RESOURCE TO administrador;
4 grant dba to administrador;

```

In the Script Output tab, the message "Grant succeeded." is displayed.

Imagen 3. Permiso para creación y administración de objetos

The screenshot shows the Oracle SQL Developer interface. In the Worksheet tab, the following SQL code is executed:

```

1 create user administrador identified by admin1234;
2 GRANT CREATE SESSION TO administrador;
3 GRANT RESOURCE TO administrador;
4 grant dba to administrador;

```

In the Script Output tab, the message "Grant succeeded." is displayed.

Imagen 4. Permiso de usuario DBA para administrador

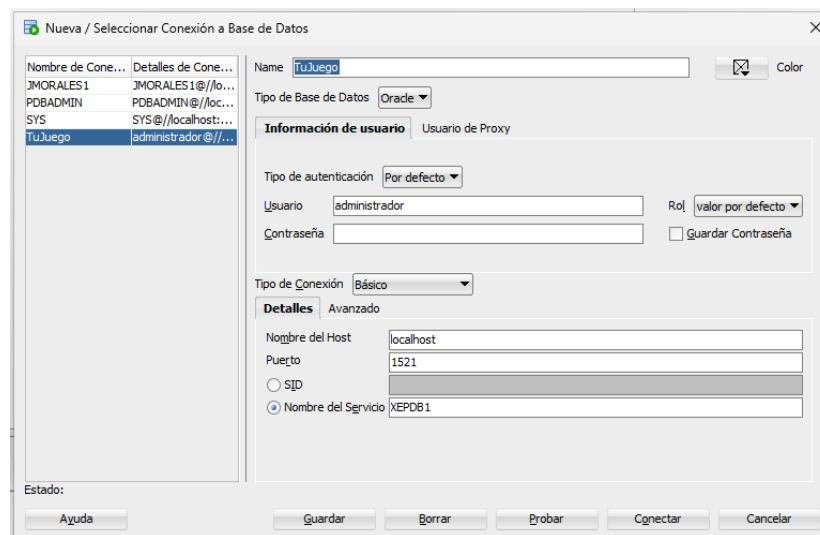


Imagen 5. Conexión a TuJuego

```

1 CREATE SEQUENCE categoria_seq
2   START WITH 1
3   INCREMENT BY 1
4   NOCACHE
5   NOCYCLE;
6
7 CREATE TABLE categoria (
8   id_categoria NUMBER PRIMARY KEY,
9   descripcion VARCHAR2(30),
10  ruta_imagen VARCHAR2(1024),
11  activo NUMBER(1)
12 );
13
14 CREATE TRIGGER categoria_before_insert
15  BEFORE INSERT ON categoria
16  FOR EACH ROW
17  BEGIN
18    :NEW.id_categoria := categoria_seq.NEXTVAL;
19  END;

```

Script Output x | Task completed in 0,282 seconds

Sequence CATEGORIA_SEQ created.

Table CATEGORIA created.

Trigger CATEGORIA_BEFORE_INSERT compiled

Imagen 6. Tabla Categoría

```

1 CREATE SEQUENCE contacto_seq
2   START WITH 1
3   INCREMENT BY 1
4   NOCACHE
5   NOCYCLE;
6
7 CREATE TABLE contacto (
8   id_contacto NUMBER PRIMARY KEY,
9   nombre VARCHAR2(100) NOT NULL,
10  apellido VARCHAR2(100) NOT NULL,
11  numero NUMBER NOT NULL,
12  mensaje VARCHAR2(1200) NOT NULL
13 );
14
15 CREATE TRIGGER contacto_before_insert
16  BEFORE INSERT ON contacto
17  FOR EACH ROW
18  BEGIN
19    :NEW.id_contacto := contacto_seq.NEXTVAL;
20  END;

```

Script Output x | Task completed in 0,877 seconds

Sequence CONTACTO_SEQ created.

Table CONTACTO created.

Trigger CONTACTO_BEFORE_INSERT compiled

Imagen 7. Tabla Contacto

```

1 CREATE SEQUENCE usuario_seq
2   START WITH 1
3   INCREMENT BY 1
4   NOCACHE
5   NOCYCLE;
6 CREATE TABLE usuario (
7   id_usuario NUMBER PRIMARY KEY,
8   username VARCHAR2(20) DEFAULT NULL,
9   contrasena VARCHAR2(512) DEFAULT NULL,
10  nombre VARCHAR2(20) DEFAULT NULL,
11  apellidos VARCHAR2(30) DEFAULT NULL,
12  correo VARCHAR2(25) DEFAULT NULL,
13  telefono VARCHAR2(15) DEFAULT NULL,
14  ruta_imagen VARCHAR2(1024) DEFAULT NULL,
15  activo NUMBER(1) DEFAULT NULL
16 );
17 CREATE TRIGGER usuario_before_insert
18 BEFORE INSERT ON usuario
19 FOR EACH ROW
20 BEGIN
21   :NEW.id_usuario := usuario_seq.NEXTVAL;
22 END;

```

Script Output x | Task completed in 0,079 seconds

Sequence USUARIO_SEQ created.

Table USUARIO created.

Trigger USUARIO_BEFORE_INSERT compiled

Imagen 8. Tabla Usuario

```

1 CREATE SEQUENCE factura_seq
2   START WITH 1
3   INCREMENT BY 1
4   NOCACHE
5   NOCYCLE;
6 CREATE TABLE factura (
7   id_factura NUMBER PRIMARY KEY,
8   id_usuario NUMBER DEFAULT NULL,
9   fecha DATE DEFAULT NULL,
10  total FLOAT DEFAULT NULL,
11  estado NUMBER DEFAULT NULL);
12 CREATE TRIGGER factura_before_insert
13 BEFORE INSERT ON factura
14 FOR EACH ROW
15 BEGIN
16   :NEW.id_factura := factura_seq.NEXTVAL;
17 END;
18 ALTER TABLE factura
19 ADD CONSTRAINT factura_ibfk_1 FOREIGN KEY (id_usuario)
20 REFERENCES usuario (id_usuario);

```

Script Output x | Task completed in 0,054 seconds

Sequence FACTURA_SEQ created.

Table FACTURA created.

Trigger FACTURA_BEFORE_INSERT compiled

Table FACTURA altered.

Imagen 9. Tabla Factura

```

1 | CREATE SEQUENCE juego_seq
2 |   START WITH 1
3 |   INCREMENT BY 1
4 |   NOCACHE
5 |   NOCYCLE;
6 | CREATE TABLE juego (
7 |   id_juego NUMBER PRIMARY KEY,
8 |   imagen VARCHAR2(1024) NOT NULL,
9 |   nombre VARCHAR2(100) NOT NULL,
10 |  empresa VARCHAR2(45) NOT NULL,
11 |  precio FLOAT NOT NULL,
12 |  existencias NUMBER NOT NULL,
13 |  activo NUMBER(1) NOT NULL,
14 |  id_categoria NUMBER DEFAULT NULL);
15 | CREATE TRIGGER juego_before_insert
16 | BEFORE INSERT ON juego
17 | FOR EACH ROW
18 | BEGIN
19 |   :NEW.id_juego := juego_seq.NEXTVAL;
20 | END;
21 | ALTER TABLE juego
22 | ADD CONSTRAINT fk_juego_categoria FOREIGN KEY (id_categoria)
23 | REFERENCES categoria (id_categoria);

```

Script Output x | Task completed in 0,074 seconds

Sequence JUEGO_SEQ created.

Table JUEGO created.

Trigger JUEGO_BEFORE_INSERT compiled

Table JUEGO altered.

Imagen 10. Tabla Juego

```

1 | CREATE SEQUENCE rol_seq
2 |   START WITH 1
3 |   INCREMENT BY 1
4 |   NOCACHE
5 |   NOCYCLE;
6 | CREATE TABLE rol (
7 |   id_rol NUMBER PRIMARY KEY,
8 |   nombre VARCHAR2(20) DEFAULT NULL,
9 |   id_usuario NUMBER DEFAULT NULL);
10 | CREATE TRIGGER rol_before_insert
11 | BEFORE INSERT ON rol
12 | FOR EACH ROW
13 | BEGIN
14 |   :NEW.id_rol := rol_seq.NEXTVAL;
15 | END;
16 | ALTER TABLE rol
17 | ADD CONSTRAINT rol_ibfk_1 FOREIGN KEY (id_usuario)
18 | REFERENCES usuario (id_usuario);

```

Script Output x | Task completed in 0,069 seconds

Sequence ROL_SEQ created.

Table ROL created.

Trigger ROL_BEFORE_INSERT compiled

Table ROL altered.

Imagen 11. Tabla Rol

```

1 CREATE SEQUENCE venta_seq
2   START WITH 1
3   INCREMENT BY 1
4   NOCACHE
5   NOCYCLE;
6 CREATE TABLE venta (
7   id_venta NUMBER PRIMARY KEY,
8   id_factura NUMBER DEFAULT NULL,
9   id_juego NUMBER DEFAULT NULL,
10  precio FLOAT DEFAULT NULL,
11  cantidad NUMBER DEFAULT NULL);
12 CREATE TRIGGER venta_before_insert
13  BEFORE INSERT ON venta
14  FOR EACH ROW
15  BEGIN
16    :NEW.id_venta := venta_seq.NEXTVAL;
17  END;
18 ALTER TABLE venta
19 ADD CONSTRAINT venta_ibfk_1 FOREIGN KEY (id_factura)
20 REFERENCES factura (id_factura);
21 ALTER TABLE venta
22 ADD CONSTRAINT venta_ibfk_2 FOREIGN KEY (id_juego)
23 REFERENCES juego (id_juego);

```

Sequence VENTA_SEQ created.

Table VENTA created.

Trigger VENTA_BEFORE_INSERT compiled

Trigger VENTA_BEFORE_INSERT compiled

Table VENTA altered.

Table VENTA altered.

Imagen 12. Tabla Venta

```

1 CREATE TABLESPACE PROYECTO LENGUAJES
2   DATAFILE 'C:\ORACLE\ORADATA\ORCL\PROY LENG01.DBF' SIZE 500M
3   AUTOEXTEND ON NEXT 100M MAXSIZE 800M
4   EXTENT MANAGEMENT LOCAL UNIFORM SIZE 10M
5   SEGMENT SPACE MANAGEMENT AUTO;
6
7
8
9  SELECT file_name, file_id, tablespace_name FROM dba_data_files;
10
11

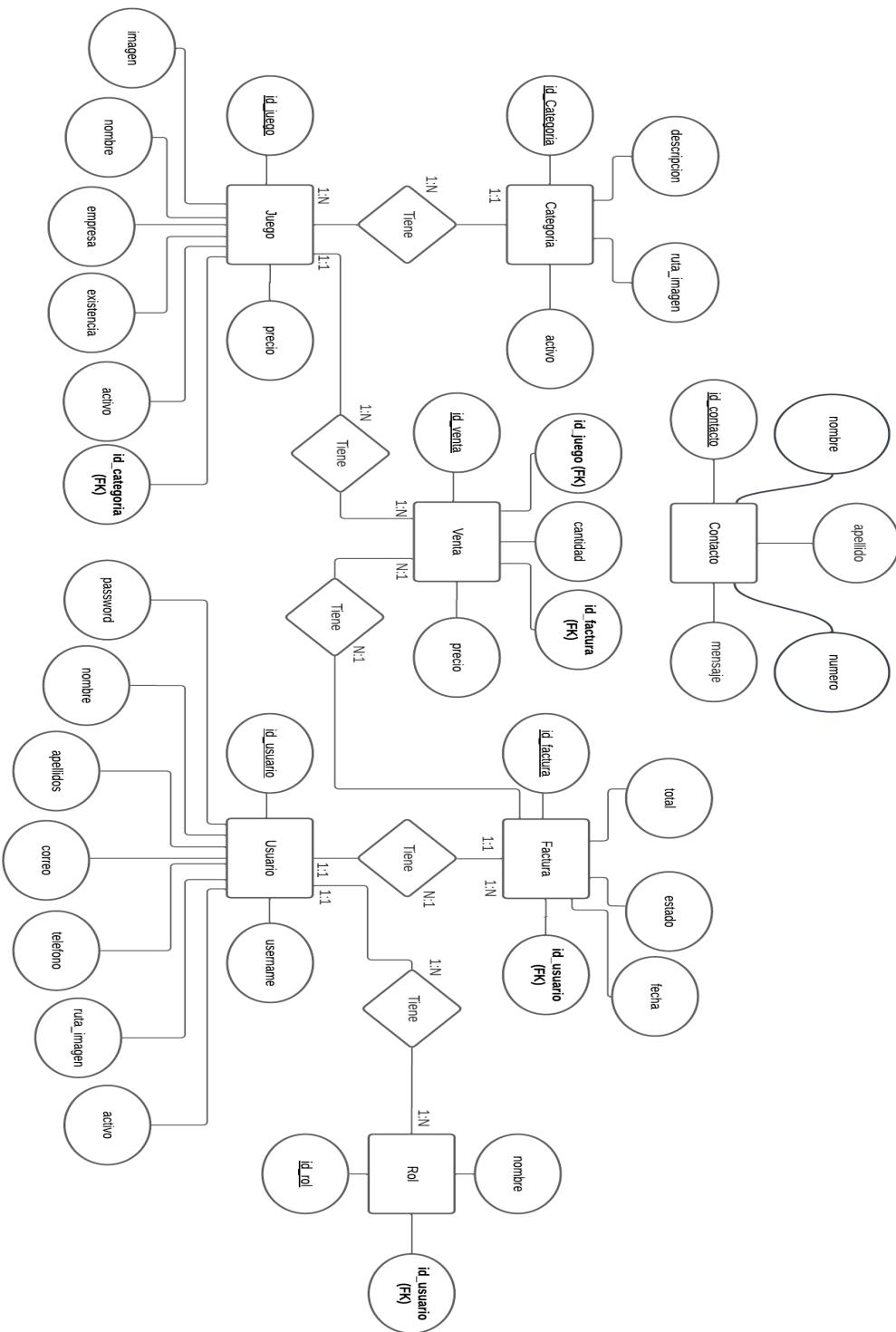
```

TABLESPACE PROYECTO LENGUAJES created.

Imagen 13. Tablespace para proyecto

2.4 Esquema entidad relación

En esta sección se presenta el diagrama entidad relación del prototipo planteado.



2.5 Propuesta de la interfaz

En la siguiente sección se mostrarán capturas de pantalla de la interfaz gráfica del prototipo en el que se basa el proyecto.

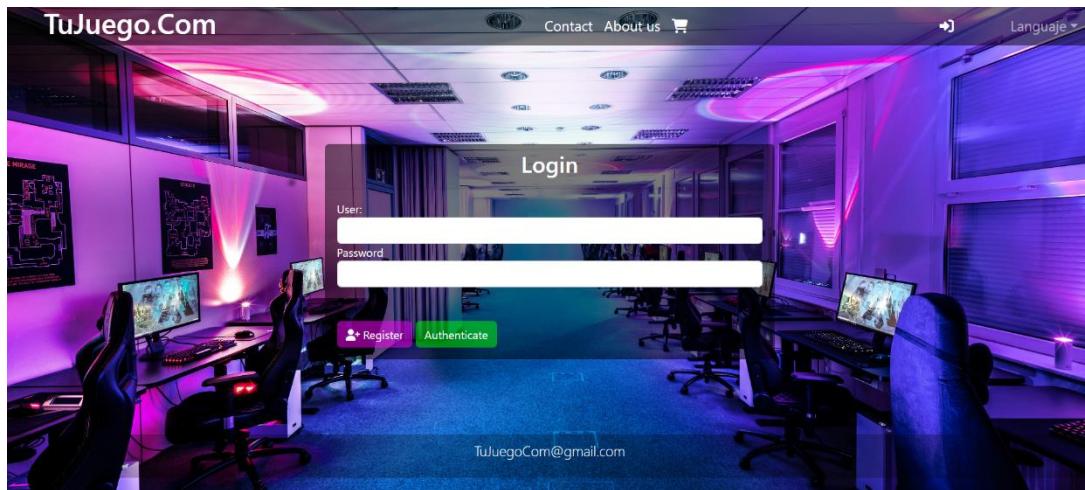


Imagen 14. Login

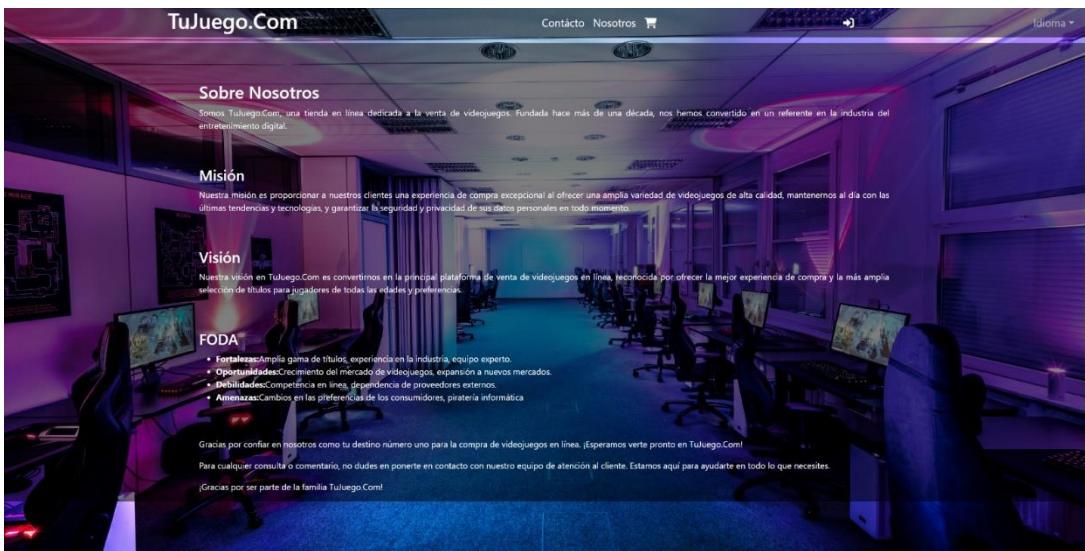


Imagen 15. Nosotros

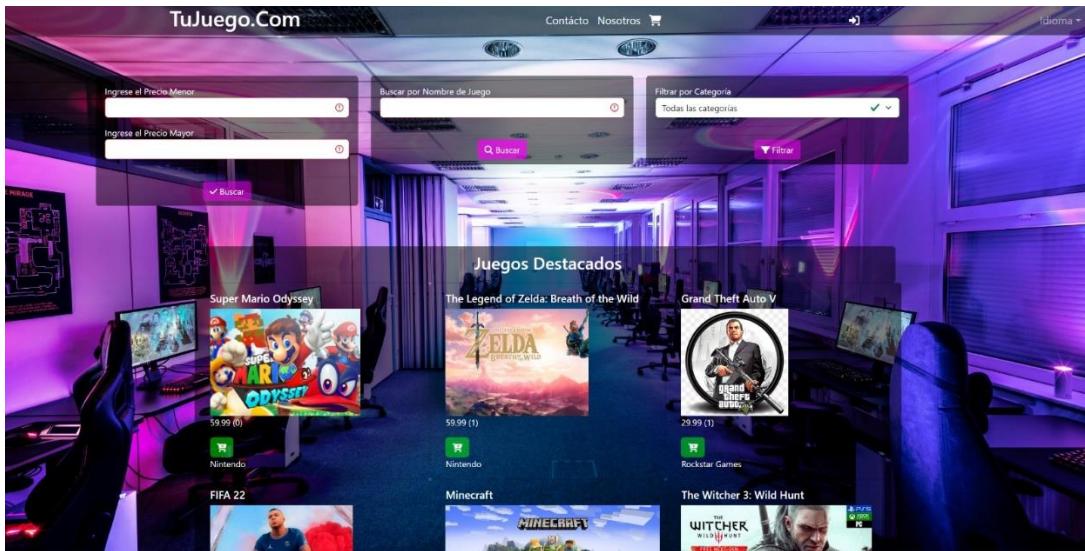


Imagen 16. Index



Imagen 17. Bienvenida

Agregar Comentario

Nombre:

Apellido:

Número de Teléfono:

Mensaje:

Guardar

#	Nombre	Número de Teléfono	Correo
1	TuJuego.com	87900098	TuJuego@gmail.com
2	Servicio al Cliente TuJuego	89762311	ServiciosTuJuego@gmail.com

Comentario

Imagen 18. Contacto

Comentario

#	Nombre	Apellido	Mensaje
1	Miguel	Reyes	Excelente Servicio
2	Alejandro	Rivero	Excelente Aplicacion
3	a	v	c
4	Jose	Sanchez	Soy Jose gracias perros los quiero
5	juan	fernandez	Gracias por el servicio
6	a	v	c
7	Ale	Rivero	AKAKA
8	Begoña	Giner	Muy buen servicio
9	Grand Theft Auto V	Giner	dw adawd

Imagen 19. Comentarios

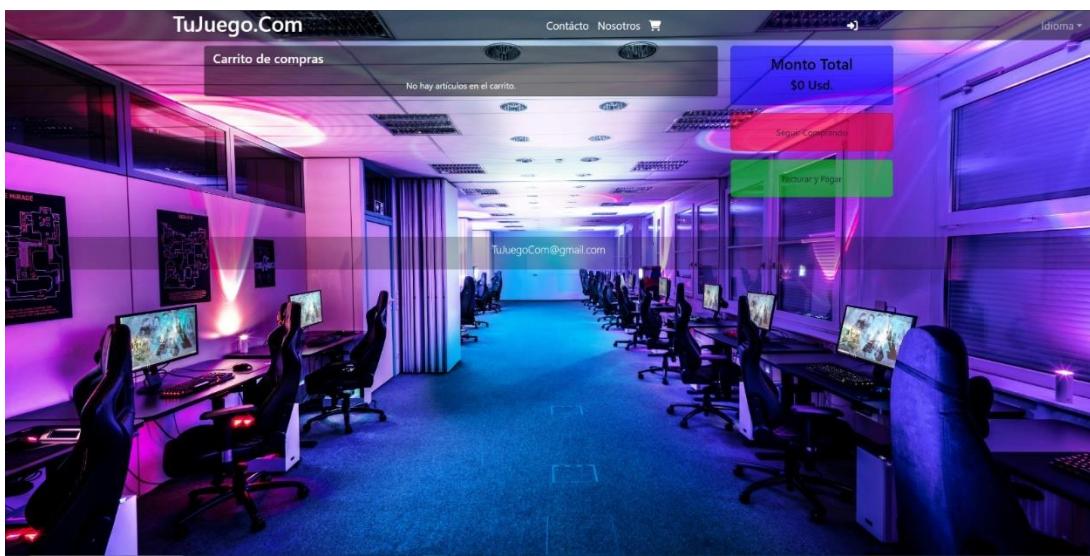


Imagen 20. Carrito

Imagen 21. Tabla de Juegos

#	Category	Image	Active
1	Accion		true
2	Aventura		true
3	Deporte		true

Imagen 22. Tabla de Categorías

#	User	Name	Last name	Delete	Update
1	juan	Juan	Castro Mora	<input type="button" value="Delete"/>	<input type="button" value="Update"/>
2	rebeca	Rebeca	Comereras Mora	<input type="button" value="Delete"/>	<input type="button" value="Update"/>
3	pedro	Pedro	Mesa Corra	<input type="button" value="Delete"/>	<input type="button" value="Update"/>
4	ale	Ale	Rivero	<input type="button" value="Delete"/>	<input type="button" value="Update"/>
5	jose	jose	jose	<input type="button" value="Delete"/>	<input type="button" value="Update"/>
6	kja	maskans	alkla	<input type="button" value="Delete"/>	<input type="button" value="Update"/>
7	JoseM	JoseM	JoseM	<input type="button" value="Delete"/>	<input type="button" value="Update"/>

Imagen 23. Tabla de Usuarios

2.6 Conexión JDBC con Base de Datos

En la siguiente sección se presentan los pasos para la conexión de tipo JDBC con la base de datos Oracle.

```
<dependency>
    <groupId>com.oracle.database.jdbc</groupId>
    <artifactId>ojdbc8</artifactId>
</dependency>
```

Imagen 24. Dependencia para conexión JDBC.



The screenshot shows a code editor window with the title "application.properties". The file contains configuration properties for a Spring application, specifically for a database connection. The properties are annotated with Java code elements such as `url`, `username`, `password`, `driver-class-name`, and `dialect`. The code editor has a dark theme and includes standard navigation and search tools.

```
spring.datasource.url=jdbc:oracle:thin:@localhost:1521/orcl
spring.datasource.username=admin_lenguajes
spring.datasource.password=Oracle2024
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver
# Hibernate dialect
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.Oracle12cDialect
```

Imagen 25. Datos para conexión.

2.7 Diccionario de datos

En la siguiente sección se presentan el diccionario de datos para los objetos creados hasta el momento, como es el caso de las tablas que se presentan a continuación.

TABLA CATEGORIA				
COLUMNA		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_CATEGORIA	NUMBER	NUMERO DE CATEGORIA	NOT NULL
	DESCRIPCION	VARCHAR2(255 CHAR)	EXPLICACION DE LA CATEGORIA	
	RUTA_IMAGEN	VARCHAR2(255 CHAR)	URL DE LA IMAGEN	
	ACTIVO	NUMBER(1)	ESTADO DE LA CATEGORIA	

Imagen 26. Diccionario de datos de la tabla categoría

TABLA CONTACTO				
COLUMNAS		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_CONTACTO	NUMBER	IDENTIFICADOR DE CONTACTO	NOT NULL
	NOMBRE	VARCHAR2(255 CHAR)	NOMBRE DE USUARIO	NOT NULL
	APELLIDO	VARCHAR2(255 CHAR)	APELLIDO DE USUARIO	NOT NULL
	NUMERO	NUMBER	NUMERO DE TELEFONO	NOT NULL
	MENSAJE	VARCHAR2(255 CHAR)	COMENTARIO EN TEXTO	NOT NULL

Imagen 27. Diccionario de datos de la tabla contacto

TABLA FACTURA				
COLUMNAS		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_FACTURA	NUMBER	IDENTIFICADOR DE FACTURA	NOT NULL
	ID_USUARIO	NUMBER	IDENTIFICADOR DE USUARIO	
	FECHA	TIMESTAMP(6)	FECHA DE FACTURA	
	TOTAL	FLOAT(53)	MONTO TOTAL DE FACTURA	
	ESTADO	NUMBER	ESTADO DE LA FACTURA	

Imagen 28. Diccionario de datos de la tabla factura

TABLA JUEGO				
COLUMNAS		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_JUEGO	NUMBER	IDENTIFICADOR DE JUEGO	NOT NULL
	IMAGEN	VARCHAR2(255 CHAR)	URL DE IMAGEN	NOT NULL
	NOMBRE	VARCHAR2(255 CHAR)	NOMBRE DEL JUEGO	NOT NULL
	EMPRESA	VARCHAR2(255 CHAR)	NOMBRE DE EMPRESA	NOT NULL
	PRECIO	FLOAT(53)	MONTO UNITARIO	NOT NULL
	EXISTENCIAS	NUMBER	CANTIDAD DE JUEGOS	NOT NULL
	ACTIVO	NUMBER	ESTADO DE LA FACTURA	NOT NULL
FK	ID_CATEGORIA	NUMBER	IDENTIFICADOR DE CATEGORIA	

Imagen 29. Diccionario de datos de la tabla juego

TABLA ROL				
COLUMNAS		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_ROL	NUMBER	IDENTIFICADOR DE ROL	NOT NULL
	NOMBRE	VARCHAR2(255 CHAR)	NOMBRE DE ROL	
	ID_USUARIO	NUMBER	IDENTIFICADOR DE USUARIO	

Imagen 30. Diccionario de datos de la tabla rol

TABLA USUARIO				
COLUMNA		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_USUARIO	NUMBER	IDENTIFICADOR DE USUARIO	NOT NULL
	USERNAME	VARCHAR2(255 CHAR)	NOMBRE DE USUARIO	
	NOMBRE	VARCHAR2(255 CHAR)	NOMBRE DE USUARIO	
	APELLIDOS	VARCHAR2(255 CHAR)	APELLIDO DE USUARIO	
	CORREO	VARCHAR2(255 CHAR)	CORREO DE USUARIO	
	TELEFONO	VARCHAR2(255 CHAR)	TELEFONO DE USUARIO	
	RUTA_IMAGEN	VARCHAR2(255 CHAR)	URL DE IMAGEN	
	ACTIVO	NUMBER(1)	ESTADO DE USUARIO	
	PASSWORD	VARCHAR2(255 CHAR)	CONTRASEÑA DE USUARIO	

Imagen 31. Diccionario de datos de la tabla usuario

TABLA VENTA				
COLUMNA		TIPO DE DATO	DESCRIPCION	NULL
PK	ID_VENTA	NUMBER	IDENTIFICADOR DE VENTA	NOT NULL
FK	ID_FACTURA	NUMBER	IDENTIFICADOR DE FACTURA	
FK	ID_JUEGO	NUMBER	IDENTIFICADOR DE JUEGO	
	PRECIO	FLOAT(53)	PRECIO TOTAL	
	CANTIDAD	NUMBER	CANTIDAD TOTAL	

Imagen 32. Diccionario de datos de la tabla venta

Objeto	ADD_CATEGORIA
Entradas	<ul style="list-style-type: none"> - DESCRIP VARCHAR2 - IMG VARCHAR2 - ACT NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que ingresa una nueva categoría al sistema

Imagen 33. Diccionario de datos de procedimiento añadir categoría

Objeto	ADD_CONTACTO
Entradas	<ul style="list-style-type: none"> - NOMBRE VARCHAR2 - APELLIDO VARCHAR2 - NUMERO NUMBER - MENSAJE VARCHAR2
Salidas	Confirmación ejecución
Función	Store procedure que ingresa un nuevo contacto al sistema

Imagen 34. Diccionario de datos de procedimiento añadir contacto

Objeto	ADD_DETALLE_FACTURA
Entradas	<ul style="list-style-type: none"> - IDFACT NUMBER - IDJUEGO NUMBER - PRECIO NUMBER - CANTIDAD NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que ingresa el detalle de factura al sistema

Imagen 35. Diccionario de datos de procedimiento añadir detalle factura

Objeto	ADD_FACTURA
Entradas	<ul style="list-style-type: none"> - USID NUMBER - FECHA DATE - TOTAL NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que ingresa una nueva factura al sistema

Imagen 36. Diccionario de datos de procedimiento añadir factura

Objeto	ADD_JUEGO
Entradas	<ul style="list-style-type: none"> - IMG VARCHAR2 - NOM VARCHAR2 - EMP VARCHAR2 - PREC NUMBER - EXI NUMBER - EST NUMBER - ID_CAT NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que ingresa un nuevo juego al sistema

Imagen 37. Diccionario de datos de procedimiento añadir juego

Objeto	ADD_ROL
Entradas	<ul style="list-style-type: none"> - RNAME VARCHAR2 - IDUSER NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que ingresa un nuevo rol al sistema

Imagen 38. Diccionario de datos de procedimiento añadir rol

Objeto	ADD_USUARIO
Entradas	<ul style="list-style-type: none"> - UNAME VARCHAR2 - PASS VARCHAR2 - UNOM VARCHAR2 - APELL VARCHAR2 - CORRE VARCHAR2 - TEL VARCHAR2 - EST NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que ingresa un nuevo usuario al sistema

Imagen 39. Diccionario de datos de procedimiento añadir usuario

Objeto	GET_CATEGORIAS
Entradas	<ul style="list-style-type: none"> - DATOS OUT SYS_REFCURSOR
Salidas	Variable datos con los valores de la tabla categoría.
Función	Store procedure que devuelve los ID_CATEGORIA y DESCRIPCION de todas las filas de la tabla CATEGORIA.

Imagen 40. Diccionario de datos de procedimiento traer categorías

Objeto	GET_FACTURA_ID
Entradas	<ul style="list-style-type: none"> - FECH IN DATE - DATOS SYS_REFCURSOR
Salidas	Variable datos con los valores ID de la tabla FACTURA.
Función	Store procedure que devuelve los ID_FACTURA de la tabla FACTURA para una fecha específica proporcionada como entrada

Imagen 41. Diccionario de datos de procedimiento traer factura por id

Objeto	GET_JUEGOS
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR
Salidas	Variable datos con los valores de la tabla JUEGOS.
Función	Store procedure que devuelve todos los datos que se encuentran en la tabla JUEGOS

Imagen 42. Diccionario de datos de procedimiento traer juegos

Objeto	GET_JUEGOSBYCATEGORIA
Entradas	<ul style="list-style-type: none"> - CID IN NUMBER - DATOS OUT SYS_REFCURSOR
Salidas	Variable datos con los valores de la tabla JUEGOS .
Función	Store procedure que devuelve todos los datos de juego cuando el ID_CATEGORIA es igual al valor ingresado.

Imagen 43. Diccionario de datos de procedimiento traer juegos por categoría

Objeto	UPDATE_CATEGORIA
Entradas	<ul style="list-style-type: none"> - CID IN NUMBER - DESCRIP IN VARCHAR2 - IMG IN VARCHAR2 - ACT IN NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que se encarga de actualizar la descripción, ruta de la imagen y estado cuando el CID es igual a ID_CATEGORIA.

Imagen 44. Diccionario de datos de procedimiento actualizar categorías

Objeto	UPDATE_JUEGO
Entradas	<ul style="list-style-type: none"> - JID IN NUMBER - IMG IN VARCHAR2 - NOM IN VARCHAR2 - EMP IN VARCHAR2 - PREC IN NUMBER - EXI IN NUMBER - EST IN NUMBER - ID_CAT IN NUMBER
Salidas	Confirmación ejecución
Función	Store procedure que se encarga de actualizar la imagen, nombre, empresa, precio, existencia, estado y el id de categoría de un juego cuando el JID es igual a ID_JUEGO.

Imagen 45. Diccionario de datos de procedimiento actualizar juegos

Objeto	GET_ONE_JUEGO
Entradas	<ul style="list-style-type: none"> - JID IN NUMBER - DATOS SYS_REFCURSOR
Salidas	Variable datos con los valores de la tabla JUEGOS.
Función	Store procedure que se encarga de devolver todos los datos de un juego cuando el JID es igual al ID_JUEGO.

Imagen 46. Diccionario de datos de procedimiento traer un juego

Objeto	SP_UPDATE_USUARIO
Entradas	<ul style="list-style-type: none"> - USERID NUMBER - USNAM VARCHAR2 - CONTRAS VARCHAR2 - NOMBR VARCHAR2 - APELLI VARCHAR2 - MAIL VARCHAR2 - PHONE VARCHAR2 - ACTV NUMBER
Salidas	Confirmación ejecución
Función	Store Procedure que actualiza los datos de usuario

Imagen 47. Diccionario de datos de procedimiento actualizar usuario

Objeto	SP_UPDATE_FACTURA
Entradas	<ul style="list-style-type: none"> - FID NUMBER - TOTL NUMBER
Salidas	Confirmación ejecución
Función	Store Procedure que actualiza los datos de factura

Imagen 48. Diccionario de datos de procedimiento actualizar factura

Objeto	SP_GET_USUARIOS
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR
Salidas	Variable datos con usuarios cargados
Función	Store Procedure que carga datos de usuarios

Imagen 49. Diccionario de datos de procedimiento traer usuarios

Objeto	SP_GET_USUARIOBYID
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR - UID NUMBER
Salidas	Variable datos con usuarios cargados
Función	Store Procedure que carga datos de usuarios por id

Imagen 50. Diccionario de datos de procedimiento traer usuarios por id

Objeto	SP_GET_USUARIOBYUSERNAME
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR - UNAME VARCHAR2
Salidas	Variable datos con usuarios cargados
Función	Store Procedure que carga datos de usuarios por USERNAME

Imagen 51. Diccionario de datos de procedimiento traer usuarios por username

Objeto	SP_GET_USERNAMEANDPASSWORD
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR - UNAME VARCHAR2 - PASS VARCHAR2
Salidas	Variable datos con usuarios cargados
Función	Store Procedure que carga datos de usuarios por username y password

Imagen 52. Diccionario de datos de procedimiento traer usuarios por username y password

Objeto	SP_GET_USERNAMEORCORREO
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR - UNAME VARCHAR2 - CORRE VARCHAR2
Salidas	Variable datos con usuarios cargados
Función	Store Procedure que carga datos de usuarios por username o correo

Imagen 53. Diccionario de datos de procedimiento traer usuarios por username o correo

Objeto	FN_USUARIO_EXISTE
Entradas	<ul style="list-style-type: none"> - UNAME VARCHAR2 - CORRE VARCHAR2
Salidas	TRUE o FALSE / 1 o 0
Función	Función que verifica si existe el username y correo

Imagen 54. Diccionario de datos de función que verifica existencia de username y correo

Objeto	SP_GET_ROLES
Entradas	<ul style="list-style-type: none"> - DATOS SYS_REFCURSOR - UID NUMBER
Salidas	Variable datos con roles cargados
Función	Store Procedure que carga datos de roles por id de usuario

Imagen 55. Diccionario de datos de procedimiento traer usuarios por username o correo

Objeto	DELETE_CATEGORIA
Entradas	<ul style="list-style-type: none"> - ID_CATEGORIA
Salidas	Confirmación de Ejecución
Función	Store procedure que elimina una categoría por medio de su id.

Imagen 56. Diccionario de datos de procedimiento eliminar categoría

Objeto	DELETE_CONTACTO
Entradas	- ID_CONTACTO
Salidas	Confirmación de Ejecución
Función	Store procedure que elimina un contacto por medio de su id.

Imagen 57. Diccionario de datos de procedimiento eliminar contacto

Objeto	DELETE_JUEGO
Entradas	- ID_JUEGO
Salidas	Confirmación de Ejecución
Función	Store procedure que elimina un juego por medio de su id.

Imagen 58. Diccionario de datos de procedimiento eliminar juego

Objeto	DELETE_ROL
Entradas	- ID_ROL
Salidas	Confirmación de Ejecución
Función	Store procedure que elimina un rol por medio de su id.

Imagen 59. Diccionario de datos de procedimiento eliminar rol

Objeto	DELETE_USUARIO
Entradas	- ID_USUARIO
Salidas	Confirmación de Ejecución
Función	Store procedure que elimina un usuario y su rol asignado, por medio de su id.

Imagen 60. Diccionario de datos de procedimiento eliminar usuario

Objeto	GET_CONTACTO
Entradas	- DATOS SYS_REFCURSOR
Salidas	Variable con los datos cargados
Función	Store procedure que obtiene los registros de la tabla contactos

Imagen 61. Diccionario de datos de procedimiento traer contacto

Objeto	GET_CATEGORIA
Entradas	- DATOS SYS_REFCURSOR
Salidas	Variable con los datos cargados
Función	Store procedure que obtiene los registros de la tabla categoría.

Imagen 62. Diccionario de datos de procedimiento traer categoría

Objeto	GET_CATEGORIA_ONE
Entradas	- ID_CATEGORIA - DATOS SYS_REFCURSOR
Salidas	Variable con los datos cargados
Función	Store procedure que obtiene el registro de la tabla categoría que coincide con el id proporcionado.

Imagen 63. Diccionario de datos de procedimiento traer categoría por id

Objeto	CATEGORIA_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 64. Diccionario de datos de secuencia categoría

Objeto	TRIGGER CATEGORIA_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 65. Diccionario de datos de trigger categoría

Objeto	CONTACTO_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 66. Diccionario de datos de secuencia contacto

Objeto	TRIGGER CONTACTO_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 67. Diccionario de datos de trigger contacto

Objeto	DETALLE_FACTURA_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 68. Diccionario de datos de secuencia detalle factura

Objeto	TRIGGER DETALLE_FACTURA_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 69. Diccionario de datos de trigger detalle factura

Objeto	FACTURA_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 70. Diccionario de datos de secuencia factura

Objeto	TRIGGER FACTURA_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 71. Diccionario de datos de trigger factura

Objeto	JUEGO_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 72. Diccionario de datos de secuencia juego

Objeto	TRIGGER JUEGO_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 73. Diccionario de datos de trigger juego

Objeto	ROL_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 74. Diccionario de datos de secuencia rol

Objeto	TRIGGER ROL_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 75. Diccionario de datos de trigger rol

Objeto	USUARIO_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 76. Diccionario de datos de secuencia usuario

Objeto	TRIGGER USUARIO_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 77. Diccionario de datos de trigger usuario

Objeto	ESTADO_SEQ
Entradas	No aplica
Salidas	Numero
Función	Incrementar en 1

Imagen 78. Diccionario de datos de secuencia estado

Objeto	TRIGGER ESTADO_BEFORE_INSERT
Entradas	Id=null
Salidas	Id según secuencia
Función	Validar si id a ingresar es null

Imagen 79. Diccionario de datos de trigger estado

Objeto	TRIGGER CATEGORIA_TRG
Entradas	n/a
Salidas	FECHA_CREACION/FECHA_MODIFICACION, CREADO_POR/ MODIFICADO_POR, ACCION.
Función	Actualiza los campos para la auditoria de categoria.

Imagen 80. Diccionario de datos de trigger auditoria de categoría.

Objeto	TRIGGER JUEGO_TRG
Entradas	n/a
Salidas	FECHA_CREACION/FECHA_MODIFICACION, CREADO_POR/ MODIFICADO_POR, ACCION.
Función	Actualiza los campos para la auditoria de juegos.

Imagen 81. Diccionario de datos de trigger auditoria de juegos.

Objeto	TRIGGER USUARIO_TRG
Entradas	n/a
Salidas	FECHA_CREACION/FECHA_MODIFICACION, CREADO_POR/ MODIFICADO_POR, ACCION.
Función	Actualiza los campos para la auditoria de usuarios.

Imagen 82. Diccionario de datos de trigger auditoria de usuarios.

3 RESULTADOS

En este apartado se muestra el ingreso de datos vía formularios de frontend, ademas de la actualización de dichos datos en la base de datos para las funciones de crear, actualizar y borrar.

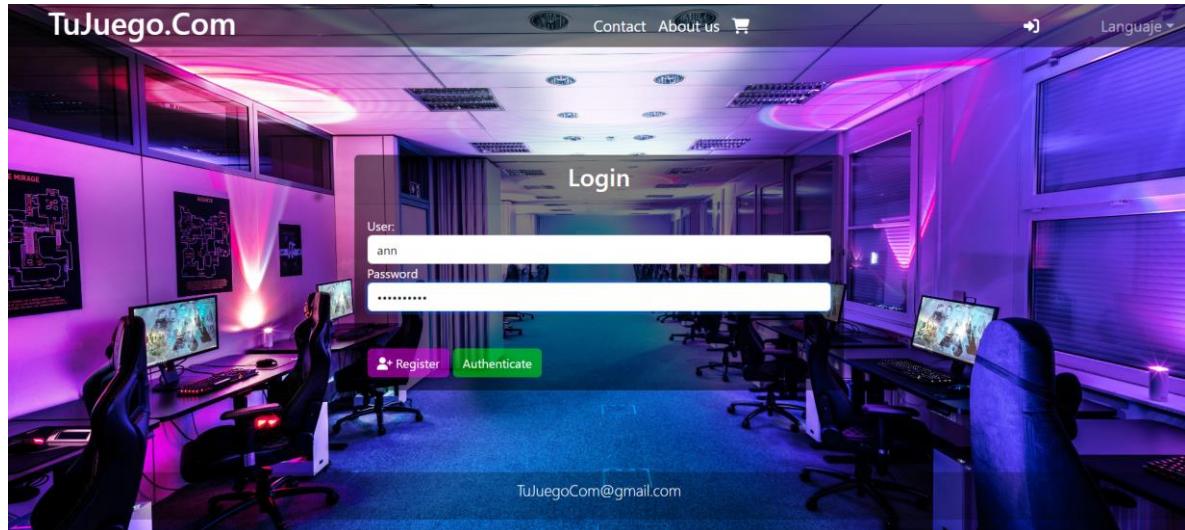


Imagen 83. Datos para inicio de sesión

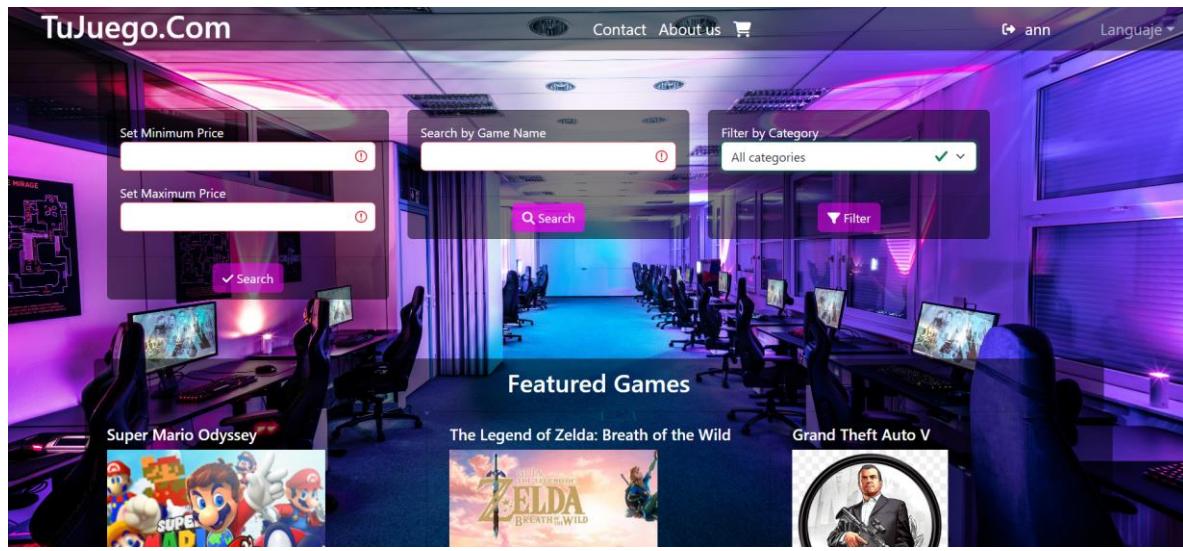


Imagen 84. Inicio de sección exitoso.

ID_USUARIO	USERNAME	NOMBRE	APELLIDOS	CORREO	TELEFONO	RUTA_IMAGEN	ACTIVO	PASSWORD
1	5 ann	ana	vargas	ana@gmail.com	88114617	(null)	1	1\$2a\$10\$WKe628BrT4jIBEyhbawba.9zt8nvcF6xOWBybarbXeRIZJ7pVt1Fi
2	1hhernan	Hector	Hernandez	hhernan93vargas@gmail.com	72063542	/ruta/imagen.jpg	1	1\$2a\$12\$U6JJq1R.Srnwq3Ok6Hwved5v8sd1gQxLsC.VeG1lPrzpBkn5Hfm

Imagen 85. Usuarios de prueba.

ID_ROL	NOMBRE	ID_USUARIO
1	3 ROLE_USER	1
2	5 ROLE_USER	5
3	1 ROLE_ADMIN	1

Imagen 86. Roles de usuarios

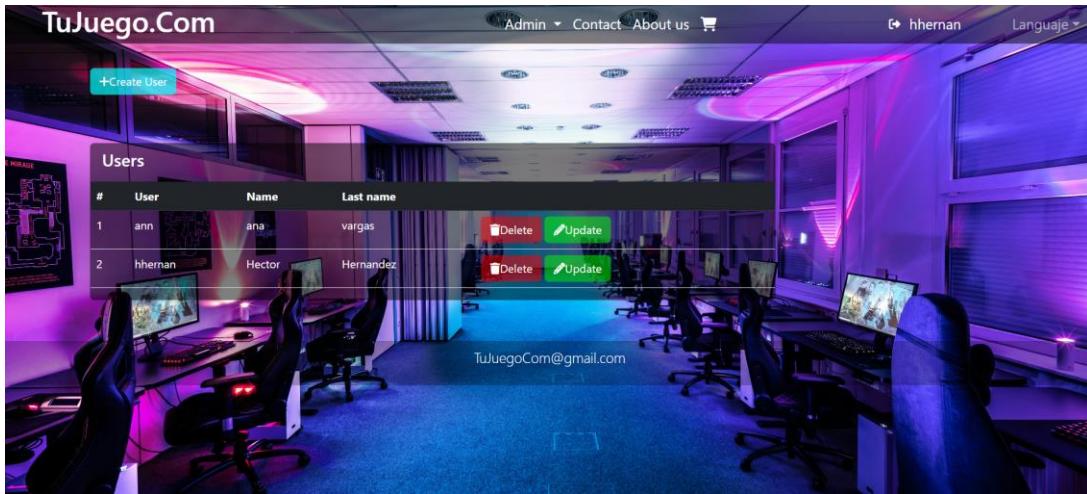


Imagen 87. Sección para creación de usuarios

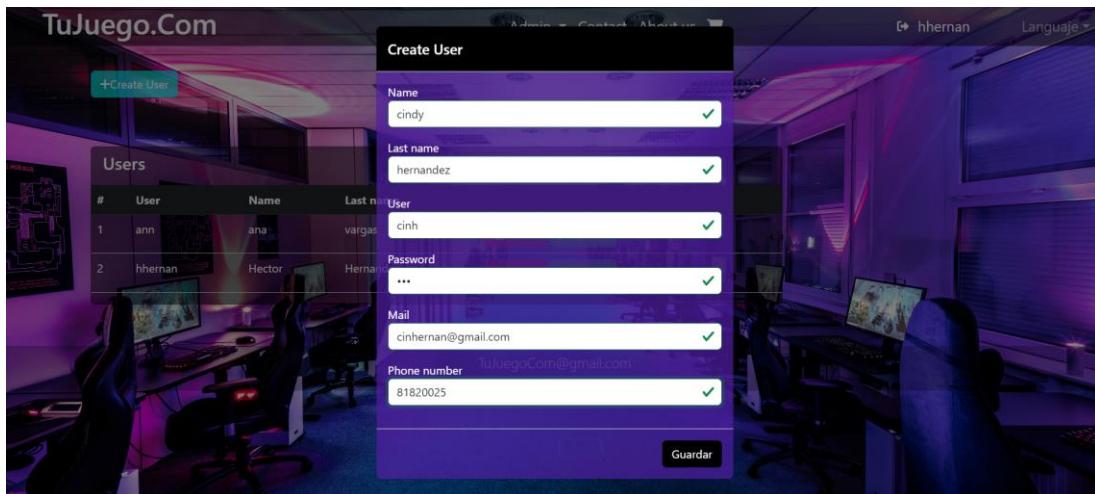


Imagen 88. Creación de usuario

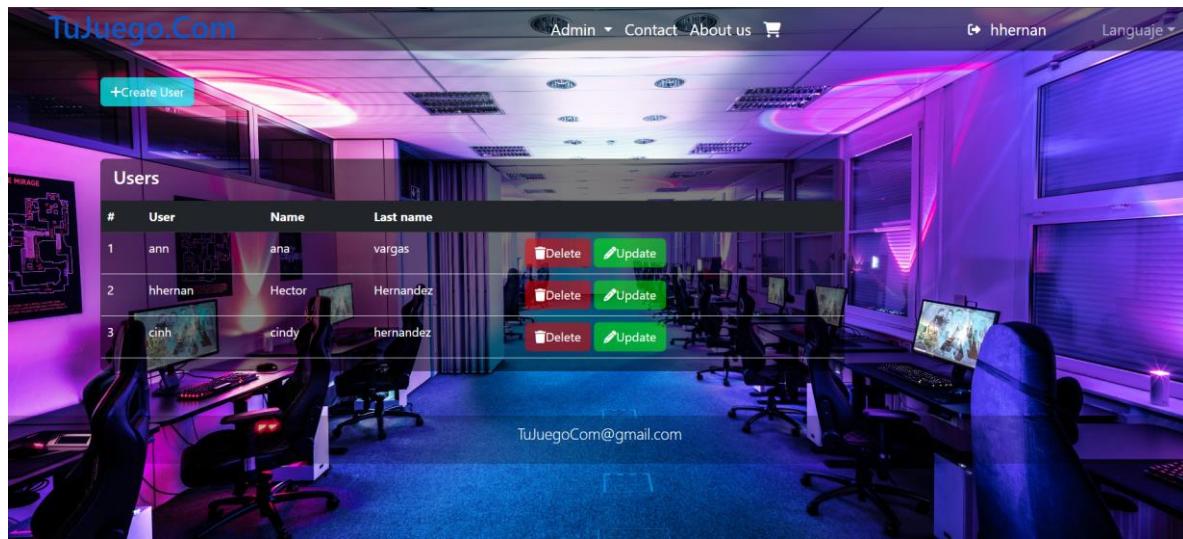


Imagen 89. Actualización de usuario en la tabla usuarios

ID_USUARIO	USERNAME	NOMBRE	APELLIDOS	CORREO	TELEFONO	RUTA_IMAGEN	ACTIVO	PASSWORD
1	S ann	ana	vargas	ana@gmail.com	88114617	(null)	1	€2a\$10gWKe628BrT4jIBEyhBnwba.9zt8nvcF6xOWBybarbXeRizJ7pVt1Ff
2	l hhernan	Hector	Hernandez	hhernan93vargas@gmail.com	72063542	/ruta/imagen.jpg	1	€2a\$12gU6JJq1R.Srnwq3h0k6Hwved5vsds1gQxLsC.VeG1lPrzBkn5Hfm
3	6 cindh	cindy	hernandez	cinhernan@gmail.com	81820025	(null)	1	€2a\$10gHnRTYUNoCZq30/air5bHubpkb4/cED4zG/SbcNPvuHFch6oxryOG

Imagen 90. Actualización de usuario en la base de datos

ID_ROL	NOMBRE	ID_USUARIO
1	3 ROLE_USER	1
2	5 ROLE_USER	5
3	1 ROLE_ADMIN	1
4	6 ROLE_USER	6

Imagen 91. Actualización de rol de nueva usuario

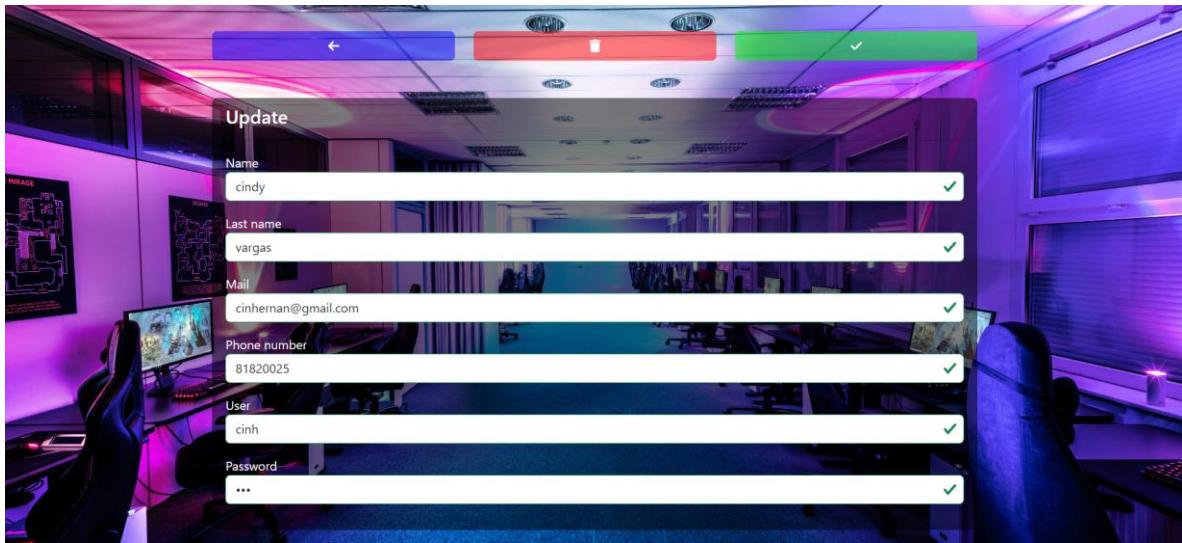


Imagen 92. Actualización de datos de usuario

Users					
#	User	Name	Last name		
1	ann	ana	vargas		
2	hhernan	Hector	Hernandez		
3	cinh	cindy	vargas		

Imagen 93. Actualización de datos de usuario exitosa

ID_USUARIO	USERNAME	NOMBRE	APPELLIDOS	CORREO	TELEFONO	RUTA_IMAGEN	ACTIVO	PASSWORD
1	5 ann	ana	vargas	ana@gmail.com	88114617	(null)	1 \$2a\$10\$WKe620BrT4jIBEyBnwba.9zt8nvxF6xOWBybarbXeRIZj7pVt1Fi	
2	1 hhernan	Hector	Hernandez	hhernan53vargas@gmail.com	72063542	/ruta/imagen.jpg	1 \$2a\$12\$U6JJqiR.Srmwq3hOk6Hwved5v8sd1gQxLsC.VeG1lPrzpBkn5Hfm	
3	6 cinh	cindy	vargas	cinhernan@gmail.com	81820025	(null)	1 \$2a\$10\$Z.0Mt/jXOKKAzZDtj6gMR.kc2VxANnpP5ftIG7s8iwDNj7I3.YLha	

Imagen 94. Actualización de datos de usuario exitosa en base de datos

Users			
#	User	Name	Last name
1	ann	ana	vargas
2	hhernan	Hector	Hernandez

Imagen 95. Eliminación de usuario

ID_USUARIO	USERNAME	NOMBRE	APELLIDOS	CORREO	TELEFONO	RUTA_IMAGEN	ACTIVO	PASSWORD
1	5 ann	ana	vargas	ana@gmail.com	88114617	(null)	1\$2a\$10\$WKe620BrT4jIBEyhBnwba.9zt8nvcF6xOWBybarbXeRIZJ7pVT1Fi	
2	1 hhernan	Hector	Hernandez	hhernan53vargas@gmail.com	72063542	/ruta/imagen.jpg	1\$2a\$12\$U6JJqiR.Srnwq3hOk6Hwvevd5v8sd1gQxLsC.VeGllPrzpBknSHfm	

Imagen 96. Eliminación de usuario en la base de datos

Category List			
#	Category	Image	Active
1	Accion		true
2	Aventura		true

Imagen 97. Sección para agregar categorías de juegos.

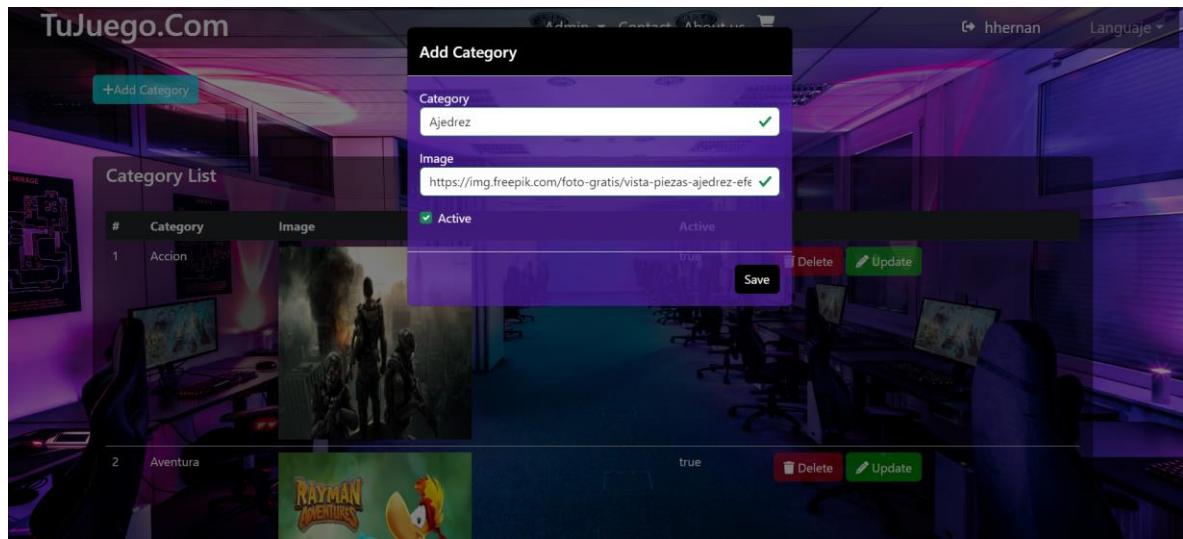


Imagen 98. Agregar categoría

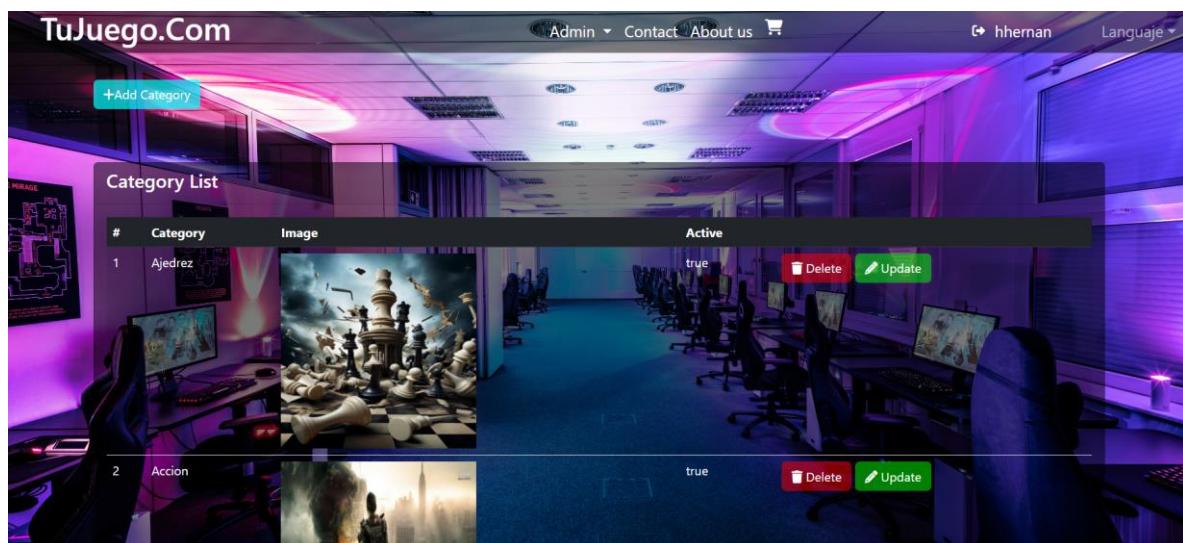


Imagen 99. Categoría agregada

ID_CATEGORIA	DESCRIPCION	RUTA_IMAGEN	ACTIVO
1	Accion	https://wallpaperaccess.com/full/5485133.jpg	1
2	Aventura	https://miracomosehace.com/wp-content/uploads/2021/06/rayman-adventures.jpg	1
3	Deporte	https://www.fantasymundo.com/wp-content/uploads/2021/02/videojuegos-deporte-696x380.jpg	1
4	Estrategia	https://media.vandal.net/m/1-2019/2019181558171_1.jpg	1
5	RPG	https://www.mundodeportivo.com/alfabeta/hero/2020/09/juegos-rpg-pc-2.jpg?width=1200	1
6	Puzzle	https://i.blogs.es/50cfef/juegos-de-puzzles/1366_2000.jpg	1
7	Ajedrez	https://img.freepik.com/foto-gratis/vista-piezas-ajedrez-efecto-rotura_23-2150829093.jpg	1

Imagen 100. Categoría agregada a la base de datos

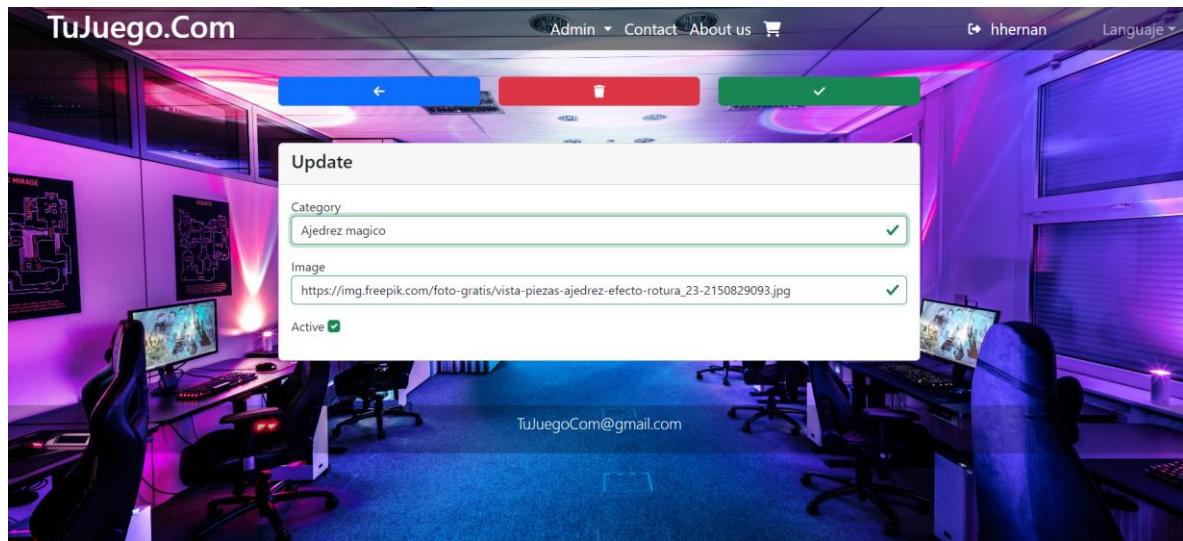


Imagen 101. Modificación de categoría

Category List			
#	Category	Image	Active
1	Ajedrez magico		true Delete Update

Imagen 102. Modificación exitosa

ID_CATEGORIA	DESCRIPCION	RUTA_IMAGEN	ACTIVO
1	1 Accion	https://wallpaperaccess.com/full/5485133.jpg	1
2	2 Aventura	https://miracomosehace.com/wp-content/uploads/2021/06/rayman-adventures.jpg	1
3	3 Deporte	https://www.fantasymundo.com/wp-content/uploads/2021/02/videojuegos-deporte-696x380.jpg	1
4	4 Estrategia	https://media.vandal.net/m/1-2019/2019181558171_1.jpg	1
5	5 RPG	https://www.mundodeportivo.com/alfabeta/heroe/2020/09/juegos-rpg-pc-2.jpg?width=1200	1
6	6 Puzzle	https://i.blogs.es/50cfef/juegos-de-puzzles/1366_2000.jpg	1
7	13 Ajedrez magico	https://img.freepik.com/foto-gratis/vista-piezas-ajedrez-efecto-rotura_23-2150829093.jpg	1

Imagen 103. Modificación exitosa en la base de datos

Imagen 104. Eliminación de categoría.

ID_CAT...	DESCRIPCION	RUTA_IMAGEN	ACTIVO
1	1 Accion	https://wallpaperaccess.com/full/5485133.jpg	1
2	2 Aventura	https://miracosehace.com/wp-content/uploads/2021/06/rayman-adventures.jpg	1
3	3 Deporte	https://www.fantasymundo.com/wp-content/uploads/2021/02/videojuegos-deporte-696x380.jpg	1
4	4 Estrategia	https://media.vandal.net/m/1-2019/2019181558171_1.jpg	1
5	5 RPG	https://www.mundodeportivo.com/alfabeta/hero/2020/09/juegos-rpg-pc-2.jpg?width=1200	1
6	6 Puzzle	https://i.blogs.es/50cfef/juegos-de-puzzles/1366_2000.jpg	1

Imagen 105. Eliminación de categoría en la base de datos

Imagen 106. Sección para agregar juegos

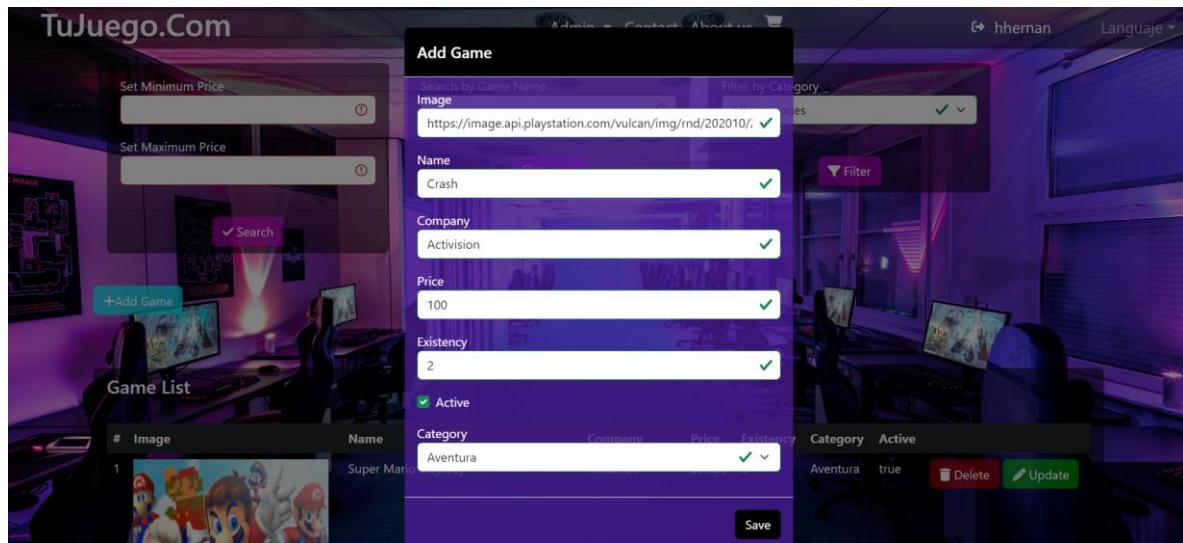


Imagen 107. Agregar nuevo juego

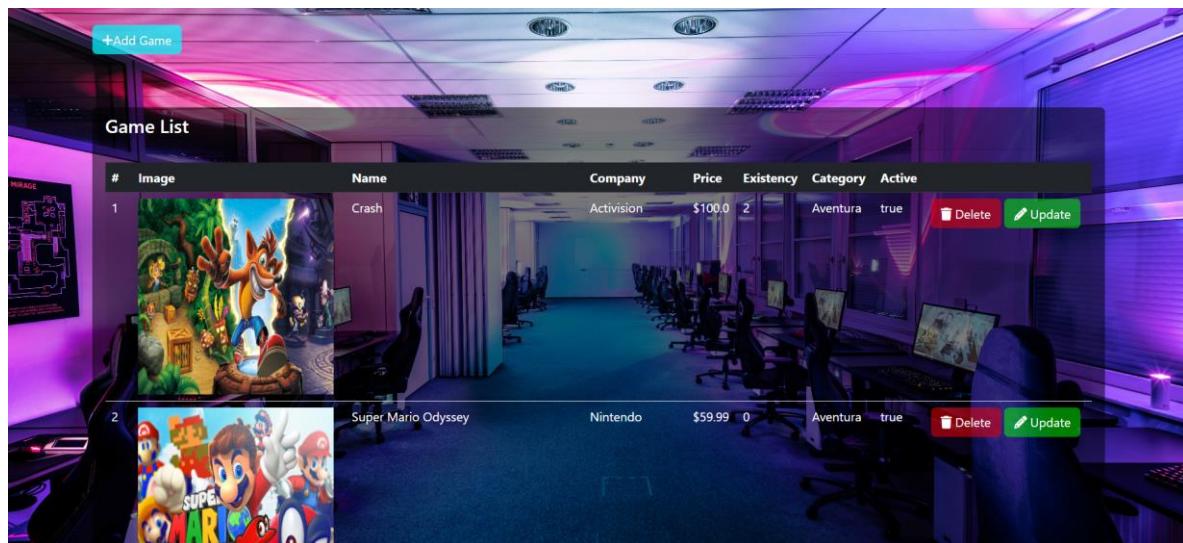


Imagen 108. Juego agregado con éxito

ID_JUEGO	IMAGEN	NOMBRE	EMPRESA	PRECIO	EXISTENCIAS	ACTIVO	ID_CATEGORIA
1	14 https://im...	Crash	Activision	100	2	1	2
2	1 https://me...	Super Mario Odyssey	Nintendo	59.99	0	1	2
3	3 https://me...	The Legend of Zelda: Breath of the Wild	Nintendo	59.99	1	1	2
4	4 https://el...	Grand Theft Auto V	Rockstar Games	29.99	1	1	1
5	5 https://fu...	FIFA 22	EA Sports	49.99	1	1	3
6	6 https://im...	Minecraft	Mojang Studios	29.99	1	1	2
7	7 https://me...	The Witcher 3: Wild Hunt	CD Projekt	39.99	1	1	4
8	8 https://i....	Call of Duty: Modern Warfare	Activision	59.99	1	1	1
9	9 https://fs...	Among Us	InnerSloth	4.99	1	1	5
10	10 https://e0...	League of Legends	Riot Games	0	0	1	6

Imagen 109. Juego agregado con éxito en base de datos

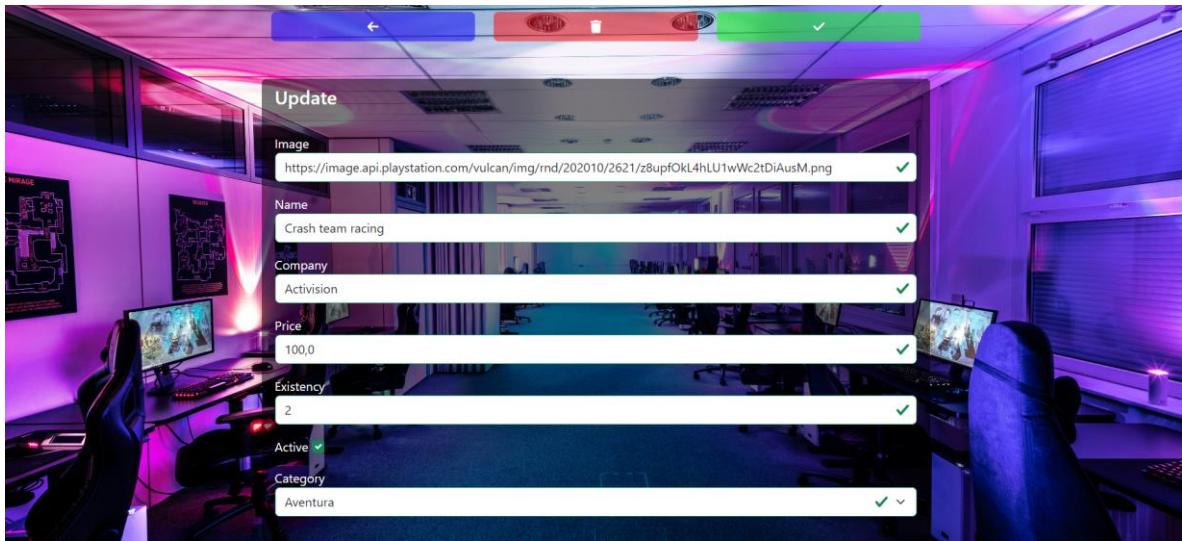


Imagen 110. Modificación de datos de juego

Game List							
#	Image	Name	Company	Price	Existency	Category	Active
1		Crash team racing	Activision	\$100.0	2	Aventura	true

Imagen 111. Modificación de datos de juego con éxito

ID_JUEGO	IMAGEN	NOMBRE	EMPRESA	PRECIO	EXISTENCIAS	ACTIVO	ID_CATEGORIA
1	14 https://im...	Crash team racing	Activision	100	2	1	2
2	1 https://me...	Super Mario Odyssey	Nintendo	59.99	0	1	2
3	3 https://me...	The Legend of Zelda: Breath of the Wild	Nintendo	59.99	1	1	2
4	4 https://el...	Grand Theft Auto V	Rockstar Games	29.99	1	1	1
5	5 https://fu...	FIFA 22	EA Sports	49.99	1	1	3
6	6 https://im...	Minecraft	Mojang Studios	29.99	1	1	2
7	7 https://me...	The Witcher 3: Wild Hunt	CD Projekt	39.99	1	1	4
8	8 https://i...	Call of Duty: Modern Warfare	Activision	59.99	1	1	1
9	9 https://fs...	Among Us	InnerSloth	4.99	1	1	5
10	10 https://e0...	League of Legends	Riot Games	0	0	1	6

Imagen 112. Modificación de datos de juego en base de datos con éxito

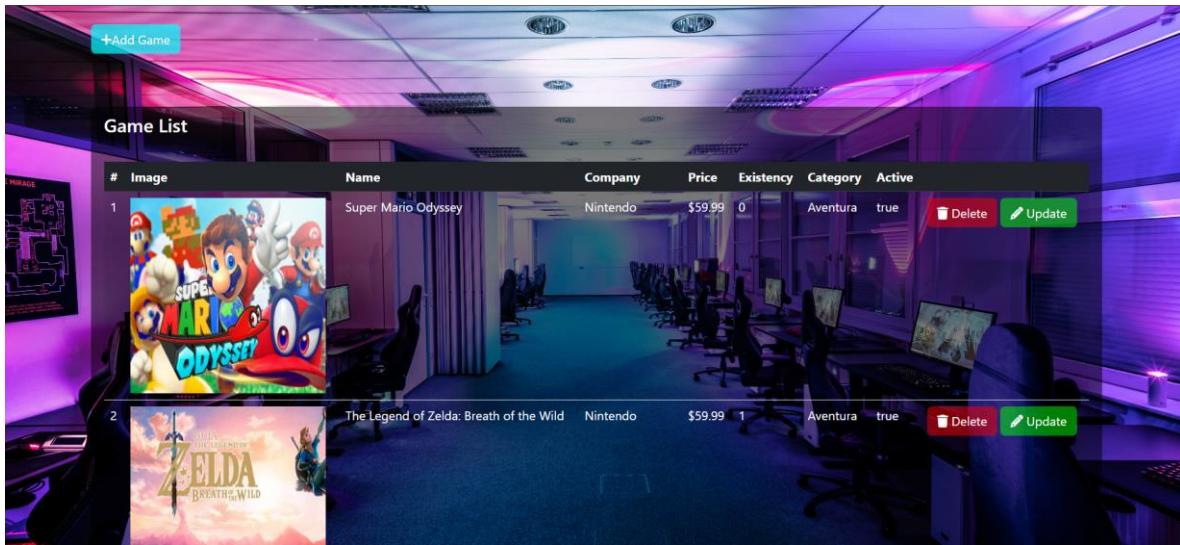


Imagen 113. Eliminación de juego

ID_JUEGO	IMAGEN	NOMBRE	EMPRESA	PRECIO	EXISTENCIAS	ACTIVO	ID_CATEGORIA
1	1 https://me...	Super Mario Odyssey	Nintendo	59.99	0	1	2
2	3 https://me...	The Legend of Zelda: Breath of the Wild	Nintendo	59.99	1	1	2
3	4 https://el...	Grand Theft Auto V	Rockstar Games	29.99	1	1	1
4	5 https://fu...	FIFA 22	EA Sports	49.99	1	1	3
5	6 https://im...	Minecraft	Mojang Studios	29.99	1	1	2
6	7 https://me...	The Witcher 3: Wild Hunt	CD Projekt	39.99	1	1	4
7	8 https://i....	Call of Duty: Modern Warfare	Activision	59.99	1	1	1
8	9 https://fs...	Among Us	InnerSloth	4.99	1	1	5
9	10 https://e0...	League of Legends	Riot Games	0	0	1	6

Imagen 114. Eliminación de juego en base de datos con éxito

Imagen 115. Sección para enviar comentario a la página de video juegos

Add comment

Name
Ana

Last name
Vargas

Phone number
88114617

Message
Tienen crash team racing??

Save

Imagen 116. Envío de comentario a la pagina

#	Name	Last name	Message
1	Hector	Hernandez	Hola, tienen COD???
2	Ana	Vargas	Tienen crash team racing??

Imagen 117. Envío de comentario exitoso

ID_CONTACTO	NOMBRE	APELLIDO	NUMERO	MENSAJE
1	1 Hector	Hernandez	72063542	Hola, tienen COD???
2	4 Ana	Vargas	88114617	Tienen crash team racing??

Imagen 118. Comentario guardado con éxito en la base de datos

#	Name	Last name	Phone number	Message	
1	Hector	Hernandez	72063542	Hola, tienen COD???	
2	Ana	Vargas	88114617	Tienen crash team racing??	

Imagen 119. Tabla de comentarios vista administrador

Comments			
#	Name	Last name	Phone number
1	Ana	Vargas	88114617
Tienen crash team racing??			<input type="button" value="Delete"/>

Imagen 120. Eliminación de comentarios vista administrador

ID_CONTACTO	NOMBRE	APELLIDO	NUMERO	MENSAJE
1	4 Ana	Vargas	88114617	Tienen crash team racing??

Imagen 121. Eliminación de comentarios vista administrador en base de datos

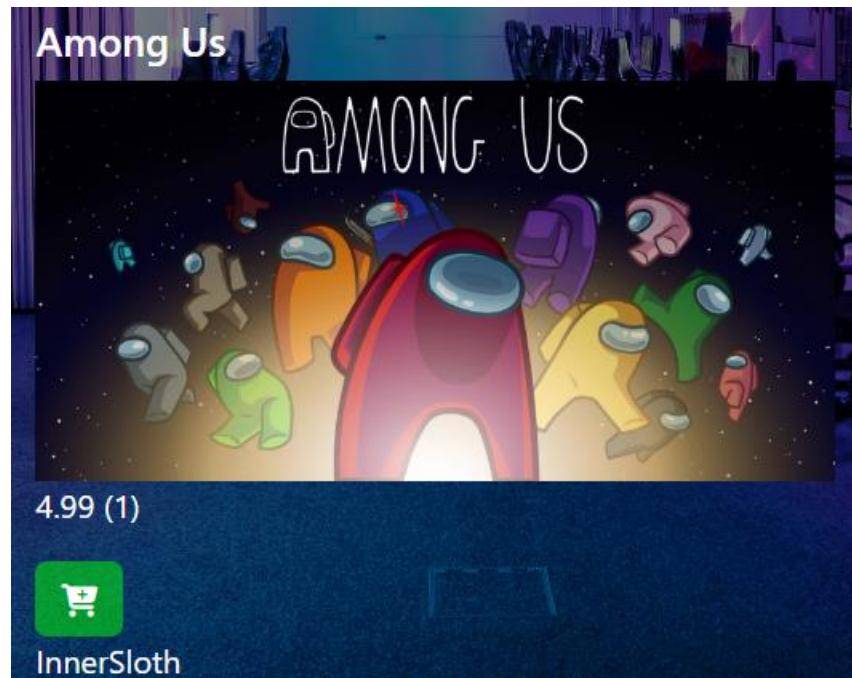


Imagen 122. Botón para agregar productos al carrito



Imagen 123. Botón para ir al carrito

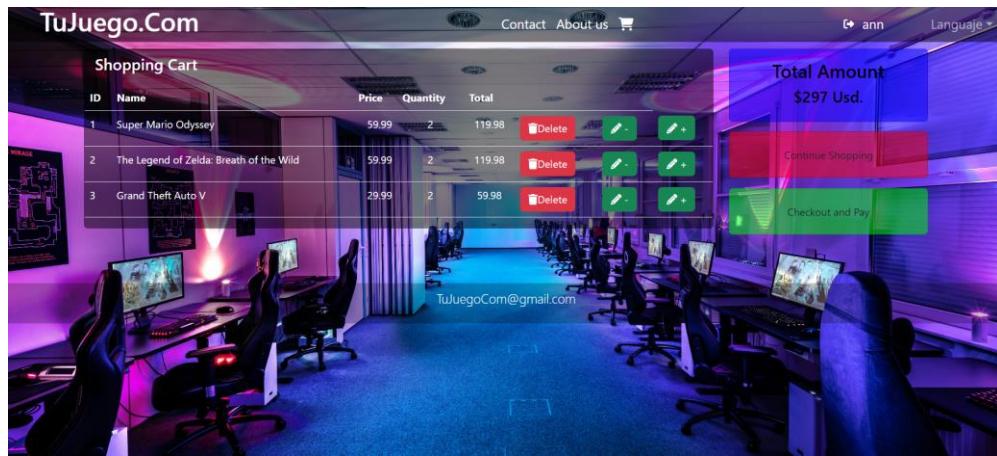


Imagen 124. Carrito

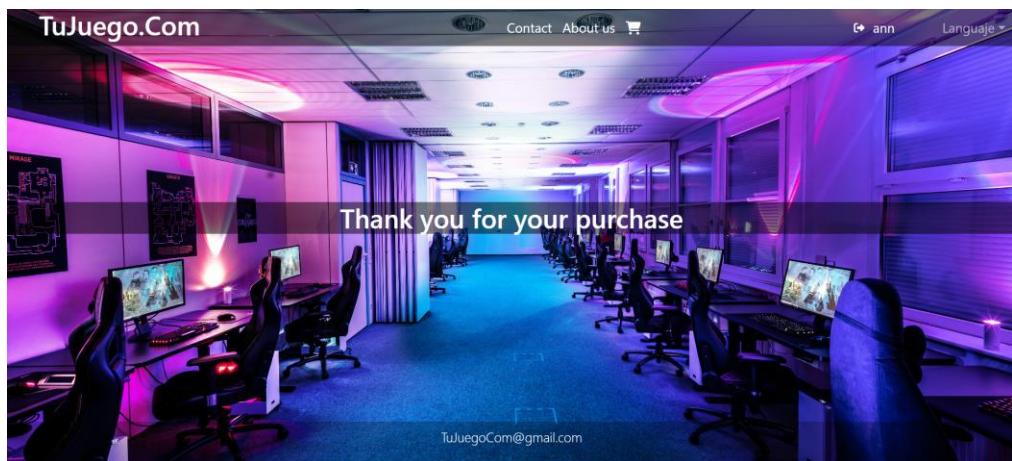


Imagen 125. Posterior a realizar el pago

ID_FACTURA	ID_USUARIO	FECHA	TOTAL	ESTADO
1	1	17-JUL-24 09.16.35.625000000 PM	59.99	1
2	3	19-JUL-24 10.44.13.233000000 PM	299.94	1
3	2	19-JUL-24 10.05.19.085000000 PM	89.98	1

Imagen 126. Total de factura

ID_VENTA	ID_FACTURA	ID_JUEGO	PRECIO	CANTIDAD
1	1	1	59.99	1
2	4	3	59.99	2
3	5	3	59.99	2
4	6	3	29.99	2
5	2	2	59.99	1
6	3	2	29.99	1

Imagen 127. Detalles de factura

4 CONCLUSIONES Y RECOMENDACIONES

- Una plataforma visualmente atractiva y fácil de navegar fue creada mediante el uso de HTML y CSS para crear un diseño de interfaz de usuario. Esto mejora la apariencia de la tienda de videojuegos en línea y facilita la interacción con los clientes, lo que aumenta la satisfacción y la probabilidad de realizar transacciones exitosas.
- La interfaz gráfica y la lógica del servidor se conectaron de manera fluida y efectiva gracias a la implementación de una arquitectura web basada en Spring Framework. Esto garantiza que las acciones de los usuarios en la interfaz se reflejen adecuadamente en el sistema, lo que permite una experiencia de usuario constante y sin interrupciones.
- El desarrollo de un modelo de datos alineado con las necesidades operativas de la aplicación permitió estructurar la base de datos de manera que soportara el volumen de usuarios y transacciones previstas. Esto no solo mejora la eficiencia en la gestión de datos, sino que también asegura la escalabilidad futura de la tienda.
- Con Java y Oracle, se creó un ambiente seguro para la manipulación de datos al establecer una conexión segura entre la aplicación y la base de datos. Esto es fundamental para asegurar la integridad de los datos almacenados en la base de datos y proteger la información confidencial de los usuarios.

Recomendaciones

- Es fundamental asegurar una documentación detallada y una planificación clara desde el inicio del proyecto. Esto facilita la comunicación entre los equipos, la gestión del proyecto y permite un desarrollo más ordenado y efectivo.
- Se necesita un ciclo de pruebas iterativas y continuas para detectar y corregir los errores a lo largo del desarrollo. Esto mejora la calidad del software y garantiza que la base de datos y la interfaz funcionen correctamente en diversas condiciones de uso.
- Para asegurar un rendimiento ideal, es fundamental concentrarse en la estructura de la base de datos y en la optimización de las consultas y procedimientos en Oracle. La gestión efectiva de los recursos del sistema, la optimización de las consultas más utilizadas y el diseño adecuado de índices son parte de esto.

5 BIBLIOGRAFÍA

Spring framework6.1.8 (2005) Spring Framework. Available at: <https://spring.io/projects/spring-framework> (Accessed: 12 June 2024).

¿Qué es una base de datos? (no date) *Qué es una base de datos / Oracle España.* Available at: <https://www.oracle.com/es/database/what-is-database/> (Accessed: 12 June 2024).