**FACULTAD DE INGENIERÍA INDUSTRIAL**

**E.A.P. INGENIERÍA INDUSTRIAL**

**Análisis y diseño de un sistema de**

**gestión de proyectos de Extensión Universitaria**

Integrantes:

Docente:

Ing Ballón

Asignatura:

Ingeniería de Software

Ciudad Universitaria, Lima. de 2022

**Contenido**

[1 Resumen 6](#_Toc94035076)

[2 Introducción 6](#_Toc94035077)

[3 Aspectos de la investigación 6](#_Toc94035078)

[3.1 Descripción del problema 6](#_Toc94035079)

[3.2 Justificación del proyecto de investigación 6](#_Toc94035080)

[3.2.1 Razones sociales 6](#_Toc94035081)

[3.2.2 Razones económicas 6](#_Toc94035082)

[3.2.3 Razones técnicas 7](#_Toc94035083)

[3.2.4 Razones organizacionales 7](#_Toc94035084)

[3.3 Delimitación 7](#_Toc94035085)

[3.3.1 Espacial 7](#_Toc94035086)

[3.3.2 Cronología 7](#_Toc94035087)

[3.3.3 Conceptual 7](#_Toc94035088)

[3.3.4 Metodología 7](#_Toc94035089)

[3.3.5 Metodología de trabajo 8](#_Toc94035090)

[4 Objetivos 9](#_Toc94035091)

[4.1 Objetivos generales 9](#_Toc94035092)

[4.2 Objetivos específicos 9](#_Toc94035093)

[5 Marco Teórico 9](#_Toc94035094)

[5.1 Antecedentes 9](#_Toc94035095)

[5.2 Ingeneiría de Software 9](#_Toc94035096)

[6 Diseño metodológico 10](#_Toc94035097)

[6.1 Análisis del Sistema 10](#_Toc94035098)

[6.1.1 Metodología para el análisis 10](#_Toc94035099)

[6.2 Diseño Lógico 11](#_Toc94035100)

[6.2.1 Requerimientos 11](#_Toc94035101)

[6.3 Actores del negocio 11](#_Toc94035102)

[6.4 Casos de uso del negocio 12](#_Toc94035103)

[6.1 Especificación de requisitos 16](#_Toc94035104)

[6.1.1 Módulo de administración del Sistema 16](#_Toc94035105)

[6.1.2 Módulo de Matrícula 16](#_Toc94035106)

[6.1.3 Módulo de Gestión de docentes 17](#_Toc94035107)

[6.1.4 Módulo de Mantenimiento de Laboratorio y Equipos 17](#_Toc94035108)

[6.1.5 Módulo de Gestión de CERSEU 17](#_Toc94035109)

[6.1.6 Módulo extensible de ERP 17](#_Toc94035110)

[6.2 Requisitos Funcionales del Sistema 17](#_Toc94035111)

[6.2.1 Requisitos de Información del Sistema 17](#_Toc94035112)

[6.3 Requisitos No Funcionales del Sistema 17](#_Toc94035113)

[6.3.1 Requisitos de Fiabilidad 18](#_Toc94035114)

[6.3.2 Requisitos de Usabilidad 18](#_Toc94035115)

[6.3.3 Requisitos de Eficiencia 18](#_Toc94035116)

[6.3.4 Requisitos de Mantenibilidad 19](#_Toc94035117)

[6.3.5 Requisitos de Portabilidad 19](#_Toc94035118)

[6.3.6 Requisitos de Seguridad 19](#_Toc94035119)

[6.3.7 Otros Requisitos No Funcionales 19](#_Toc94035120)

[6.4 Restricciones Técnicas del Sistema 19](#_Toc94035121)

[6.5 Requisitos de Integración del Sistema 20](#_Toc94035122)

[6.6 Actores del Sistema 20](#_Toc94035123)

[6.7 Casos de Uso del Sistema 21](#_Toc94035124)

[7 Implementación del sistema 34](#_Toc94035125)

[7.1 Creación de la base de datos en SQL Server 2008 34](#_Toc94035126)

[8 Análisis de resultados 36](#_Toc94035127)

[8.1 Codificación del programa 36](#_Toc94035128)

[8.2 Banco de pruebas 36](#_Toc94035129)

[8.2.1 Pruebas de función 36](#_Toc94035130)

[8.2.2 Pruebas modulares 36](#_Toc94035131)

[8.2.3 Pruebas de documentación y ayuda 37](#_Toc94035132)

[8.2.4 Pruebas de seguridad y control 37](#_Toc94035133)

[8.2.5 Pruebas de calidad 38](#_Toc94035134)

[8.3 Informe de pruebas 38](#_Toc94035135)

[8.4 Material revisado 38](#_Toc94035136)

[9 Conclusiones y Recomendaciones 38](#_Toc94035137)

[10 Bibliografía 39](#_Toc94035138)

[11 Anexos 39](#_Toc94035139)

[11.1 Acta de reunión ( Meeting) 39](#_Toc94035140)

[11.2 Acta de actividad de extensión ( CERSEU) 41](#_Toc94035141)

[12 Base de datos 43](#_Toc94035142)

[12.1 Scripts de creación de base de datos 43](#_Toc94035143)

[13 Proyecto en Visual Studio C# 96](#_Toc94035144)

# Resumen

El presente trabajo de investigación es una aplicación de la ingeniería de software a sistemas complejos de gestión y control, teniendo en cuenta los requerimientos y necesidades de la oficina de CERSEU en la facultad de Ingeniería Industrial. El estudio se basa en diseñar un software que optimice el proceso de mantenimiento, creación, anulación y clasificación de cursos y eventos desarrollados por esta oficina.

Para el desarrollo del software se ha utilizado programación por capas en Visual Studio 2015 con conexión a una base de datos SQL Server, mediante el lenguaje de programación C# sumado a componentes de programación ADO.NET. El desarrollo de las clases de Ingeniería de Software en la Facultad de Ingeniería Industrial fue fundamental para el desarrollo de esta aplicación.

# Introducción

# Aspectos de la investigación

## Descripción del problema

## Justificación del proyecto de investigación

### Razones sociales

Entre las razones sociales que existen para la elaboración de este proyecto, se encuentran la importancia de gestionar las reuniones de los docentes para que discuten sobre temas de enseñanza. De esta manera, se incrementaría el compromiso por parte de los docentes en su labor crucial en la educación superior en el país.

### Razones económicas

Se busca crear el software con el fin de que las facultades que lo soliciten puedan adquirirlo a bajos costos, ya que fue realizado por alumnos de manera ad honorem y se utilizó software libre para su creación.

### Razones técnicas

La aplicación podrá ser utilizada en equipos con plataforma Windows, en el lenguaje Visual C# y con bases de datos Microsoft SQL Server 2014 Express Edition, ya que estas herramientas son de la misma casa de software, son compatibles entre ellas y son populares a nivel computacional en la actualidad.

### Razones organizacionales

Las facultades que cuenten con un software como este, tendrán la posibilidad, de agilizar los procesos, de tener una estricta organización de sus docentes. Así mismo, tendrán la facilidad de adquirirlo, pues los bajos costos del software otorgan una ventaja competitiva.

## Delimitación

### Espacial

El proyecto se realiza en las instalaciones de la Facultad de Ingeniería Industrial de la UNMSM, bajo la coordinación con el ingeniero Wiler Ponce docente del curso de Base de Datos y Programación Visual, en donde se entregarán actas semanales para dicho curso.

### Cronología

El proyecto se realiza en un lapso de tiempo de 4 meses. Teniendo presente la presentación de avances periódicos al docente.

### Conceptual

En el proyecto se realiza la investigación sobre las reuniones de docentes en la universidad peruana, así mismo se realiza el manejo de la base de datos Microsoft SQL Server 2014 Express Edition, ya que es una herramienta nueva para el equipo de trabajo.

### Metodología

A nivel metodológico se trabajó el modelo de Análisis y Diseño orientado a Objetos debido a la orientación del proyecto y también por el lenguaje en el cual se realizará, en todo el CVDS (Ciclo de Vida de Desarrollo de Software), dado que esta metodología tiene un conjunto de disciplinas que desarrollan y modelan el software y facilitan la construcción de sistemas complejos a partir de componentes.

Para la construcción y el modelado de todos los componentes del proyecto a nivel de software se utilizó la herramienta UML, las siglas corresponden a Lenguaje Unificado de Modelado (Unified Modeling Language), la definición de UML se puede obtener descomponiendo las palabras que la conforman.

Lenguaje: como todo lenguaje cuenta con una sintaxis y una semántica, por lo tanto, al realizar cualquier modelo es necesario conocer una serie de reglas de cómo es el manejo del elemento s en el lenguaje

Modelado: con él se modelan los aspectos del mundo real, los cuales permiten la correcta interpretación y entendimiento de este.

Unificado: es unificado ya que agrupa varias técnicas de modelado en un solo modelo.

El proyecto se desarrolló a nivel de programación siguiendo un patrón de diseño, modelo vista controlador, el cual es bastante acorde con el modelo de Análisis y diseño orientado a objetos.

**Patrón Modelo-Vista-Controlador**

Este patrón se utiliza en diseño de aplicaciones con sofisticadas interfaces. El fin es realizar un diseño que separe la vista del modelo, esto para aprovechar la reusabilidad de código, de tal forma que los cambios en las vistas no afecten el modelo del negocio y el modelo de datos, por otra parte, al utilizar la herramienta .Net para el proyecto, en futuro se podrá llevar a la web por medio de ASP.Net, utilizando la misma controladora y modelo, para entender mejor cada uno de los componentes del modelo, se describen a continuación.

* El modelo es el responsable de:

Acceder a la capa de almacenamiento de datos. Es ideal que el modelo sea independiente del sistema de almacenamiento. Define las reglas de negocio.

* El controlador es responsable de:

Recibe los eventos de entrada Contiene reglas de gestión de eventos, tanto de la vista como del modelo.

* Las vistas son responsables de:

Recibir datos del modelo y mostrarlos al usuario, así como hacer el proceso contrario recibir los datos del usuario y registrarlos en el modelo para que sean procesados y registrados.

### Metodología de trabajo

Se seguirá la metodología ágil de software SCRUM

El tamaño óptimo del Equipo de Desarrollo es lo suficientemente pequeño como para permanecer ágil y lo suficientemente grande como para completar una cantidad de trabajo significativa. Tener menos de tres miembros en el Equipo de Desarrollo reduce la interacción y resulta en ganancias de productividad más pequeñas. Los Equipos de Desarrollo más pequeños podrían encontrar limitaciones en cuanto a las habilidades necesarias durante un Sprint, haciendo que el Equipo de Desarrollo no pudiese entregar un Incremento que potencialmente se pueda poner en producción. Tener más de nueve miembros en el equipo requiere demasiada coordinación. Los Equipos de Desarrollo grandes generan demasiada complejidad como para que pueda gestionarse mediante un proceso empírico.

En el caso particular de este trabajo, el trabajo fue realizado por dos estudiantes.

# Objetivos

## Objetivos generales

* Desarrollar un sistema que gestione las reuniones de docentes universitarios con el fin de promover su participación en la discusión sobre temas de enseñanza.

## Objetivos específicos

* Modelar la gestión de las reuniones.
* Implementar una base de datos que sirva como soporte del sistema.
* Implementar una API (Interfaz de programación de aplicaciones, por sus siglas en inglés) para el sistema.

# Marco Teórico

## Antecedentes

## Ingeneiría de Software

Para el desarrollo del proyecto se utilizó un motor de base de Microsoft SQL Server 2014 Express Edition, versión libre en internet, donde se implementaron la base de datos para el almacenamiento de la información, relacionada con la gestión de reuniones de los docentes.

La aplicación se realizó para plataformas Microsoft. El lenguaje de programación que se utilizó para el desarrollo de la aplicación es Visual C#.

**Microsoft SQL Server 2014 Express Edition**

Está diseñado para brindar una simplicidad de uso, permitiendo instalaciones rápidas. La facilidad de uso comienza con una instalación robusta de la interface del usuario grafica (GUI) que guía al usuario a través del proceso de instalación. Las herramientas GUI que se encuentran incluidas sin cargo con SQL Server Express, incluyen Express Manager (versión Alfa) y Computer Manager. Estas herramientas simplifican las operaciones de bases de datos. El diseño y desarrollo de aplicaciones de bases de datos se vuelve más sencillo con la integración de Visual Studio. Con todas estas ventajas de interface de usuario, en el transcurso del proyecto se presentó la necesidad de ejecutar sentencias en SQL, por complejidad en la realización de procesos.

**Visual Studio .NET**

Visual Studio .NET es un conjunto completo de herramientas de desarrollo para la construcción de aplicaciones Web ASP, servicios Web XML, aplicaciones para escritorio y aplicaciones móviles. Visual Basic .NET, Visual C++ .NET, Visual C# .NET y Visual J# .NET utilizan el mismo entorno de desarrollo integrado (IDE), que les permite compartir herramientas y facilita la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes aprovechan las funciones de .NET Framework, que ofrece acceso a tecnologías clave para simplificar el desarrollo de aplicaciones Web ASP y servicios Web XML.

# Diseño metodológico

El presente proyecto de investigación es de tipo descriptivo, ya que en la investigación se busca describir y especificar las propiedades más importantes de los procesos computacionales para lograr requerimientos óptimos y acordes con las necesidades de la gestión de reuniones de los en la universidad.

Adicionalmente en la investigación, se describirán las características que identifican los diferentes elementos y componentes de la investigación.

## Análisis del Sistema

### Metodología para el análisis

#### Análisis orientado a objetos

#### El presente proyecto de investigación se trabaja con metodología AOO, en donde se brindan los medios para mejorar la reutilización de los componentes software. El objetivo de la programación orientada a objetos es permitir que los sistemas informáticos sean fácilmente extendidos para mejorar su funcionalidad, o reutilizados en otros sistemas que requieran sus servicios.

Las características que integran la Programación orientada a objetos son la: abstracción, encapsulación, polimorfismo y herencia.

La herramienta de diseño será UML, Lenguaje de Modelamiento Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software.

Se utilizan los siguientes diagramas para su conceptualización:

• Diagrama de Casos de Uso: mostrando los procesos relevantes en un escenario con sus respectivos actores.

• Diagrama de Secuencia: Muestran a los diferentes objetos y las relaciones que pueden tener entre ellos, los mensajes que se envían entre ellos.

• Diagrama de Actividades: Es un caso especial del diagrama de estados. Muestra el flujo entre los objetos. Se utilizan para modelar el funcionamiento del sistema y el flujo de control entre objetos.

• Diagramas de Clases: muestran las diferentes clases que componen un sistema y cómo se relacionan unas con otras. Los diagramas de clases son diagramas estáticos porque muestran las clases, junto con sus métodos y atributos, así como las relaciones estáticas entre ella.

## Diseño Lógico

### Requerimientos

El sistema establecerá un mecanismo de gestión para reuniones de docentes. De modo que es necesario identificar cuáles son requerimientos. En ese sentido identificamos los requisitos funcionales.

**Requerimientos funcionales**: Definen qué debe hacer el sistema.[[1]](#footnote-1)

1. Los jefes de Departamento, programarán reuniones periódicas, donde se discutan con los docentes temas relacionados con actividades de enseñanza, aprendizaje y sus resultados.

2. Registrar los acuerdos consensuados en actas para su seguimiento y cumplimiento.

3. Establecer los mecanismos para verificar el cumplimiento de los acuerdos.

## Actores del negocio

A picture containing text, athletic game, sport

Description automatically generated

|  |  |
| --- | --- |
| Actor de negocio | Descripción |
| Alumno |  |
| Docente |  |
| Secretaría |  |
| Jefe de unidad académica |  |
| Jefe de imprenta |  |
| Jefe de CERSEU |  |
| Director de escuela |  |
| Decano de facultad |  |

## Casos de uso del negocio



**Diagrama 2.Diagrama de secuencia de Asignación de tareas**



**Diagrama 3.Diagrama de secuencia de registro de Asistencia**



**Diagrama 4.Diagrama de secuencia de invitación a reuniones**



**Diagrama 5.Diagrama de secuencia de Redacción de documentación de actas**



**Diagrama 6. Diagrama de envío de actas de reuniones al director de escuela**



**Diagrama7. Diagrama de secuencia de Revisión de actas de reuniones por el directo**

Diagram

Description automatically generated

CASOS DE USO DEL SISTEMA

## Especificación de requisitos

### Módulo de administración del Sistema

ECU01 – Login del sistema

ECU02 – CRUD Usuarios del sistema

ECU02 – CRUD Cursos del sistema

ECU03 – Mantenimiento de roles y permisos

### Módulo de Matrícula

ECU04 – Gestión de Matrícula

ECU06 – Gestión de Pagos

ECU07 – Reportes

ECU08 – Información

ECU09 – Programación de Cursos

### Módulo de Gestión de docentes

ECU09 – CRUD Reuniones

ECU09 – CRUD Asistencias de alumnos

### Módulo de Mantenimiento de Laboratorio y Equipos

ECU04 – CRUD Salas

ECU05 – CRUD Equipos

ECU06 – CRUD Actividades de Mantenimiento

### Módulo de Gestión de CERSEU

ECU06 – CRUD Actividades

### Módulo extensible de ERP

ECU06 – Módulo de imprenta

ECU06 – Módulo de CERSEU

## Requisitos Funcionales del Sistema

### Requisitos de Información del Sistema

El sistema debe:

**PRODUCCIÓN**

**Submódulo de Producción de Imprenta**

* Registrar datos para calcular costos de producción
* Actualizar costos de producción
* Anular costos de producción
* Mantenimiento de precios fijos
* Elaborar reporte de costos de producción en Excel y PDF

## Requisitos No Funcionales del Sistema

Para esta aplicación hemos definido los siguientes requerimientos no funcionales:

* Esquema de sincronización de datos entre los niveles del sistema
* Integridad de información entre los sistemas
* Integridad del esquema de sincronización
* Capacidad del sistema de trabajar tanto en un ambiente conectado como en uno no conectado
* Despliegue de la aplicación sobre sistemas de baja capacidad

### Requisitos de Fiabilidad

El control de acceso a la aplicación será controlado por una validación de nombre de usuario y contraseña. De esta manera para que un usuario pueda ingresar al sistema debe haber sido previamente autorizado por el administrador de la aplicación, quien creara un nombre de usuario y una contraseña asociado a un perfil, el cual definirá los permisos y funcionalidades que tendrá el usuario sobre el sistema.

* El sistema debe estar disponible 24x7x365 días al año.
* La duración promedio de una reparación del sistema no debe ser mayor de 60 minutos.

### Requisitos de Usabilidad

La interacción entre el sistema y el usuario será, por lo general, a través de botones de acción que son los puntos de inicio para los casos de uso de la aplicación.

La interfaz de la aplicación será amigable y de fácil entendimiento para el usuario, debido a que muchos de ellos son inexpertos.

* El sistema permitirá a los usuarios el registro de incidencias o llamadas en un tiempo promedio de 45 segundos.
* El sistema permitirá a los usuarios del sistema realizar búsquedas sin entrenamiento previo.
* El sistema se ajustará a los estándares CUA (Common User Access) de IBM. El estándar se podrá encontrar en la carpeta “Estándar” en el Sharepoint de la empresa.
* En caso de error del usuario el sistema informará con el siguiente formato: código del error, el mensaje del error y la solución. Dicha lista de soluciones estará definida en la documentación de la compañía.
* El lenguaje empleado en la interfaz gráfica del sistema respetará los términos usados en el negocio. Dicho glosario se establece en los anexos.

### Requisitos de Eficiencia

Garantizar que el diseño de las consultas u otro proceso no afecte el desempeño de la base de datos, ni considerablemente el tráfico de la red.

1. El sistema permitirá el acceso concurrente de 100 usuarios en promedio.
2. El tiempo de respuesta promedio del sistema para las operaciones involucradas en el proceso de evaluación es de 10 segundos.
3. El sistema debe soportar un promedio de 50 transacciones por minuto.

### Requisitos de Mantenibilidad

Para el soporte de la aplicación, existirá un manual de uso, que procurará ofrecer a los usuarios información entendible, que les ayude cuando no puedan entender alguna funcionalidad del sistema. Cada usuario del sistema debe ser dotado con uno de estos manuales de uso.

La programación orientada a objetos permite además la fácil mejora de la aplicación para el futuro.

### Requisitos de Portabilidad

* El sistema será implantado bajo la plataforma de Windows.

### Requisitos de Seguridad

• Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.

• Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.

• Facilidades y controles para permitir el acceso a la información al personal autorizado a través de Internet, con la intención de consultar y subir información pertinente para cada una de ellas.

### Otros Requisitos No Funcionales

## Restricciones Técnicas del Sistema

|  |  |
| --- | --- |
| **<id>999** | *<nombre descriptivo>* |
| **[Versión]** | *<nº versión>(<fecha de versión>)* |
| **[Dependencias]** | * *<requisitos generales de los que depende>* * *<otros requisitos de los que depende>* * *...* |
| **Descripción** | El sistema deberá respetar la siguiente restricción técnica: *<descripción de la restricción técnica del sistema>* |
| **[Importancia]** | *<importancia de la restricción técnica para el cliente>* |
| **[Prioridad]** | *<prioridad dela restricción técnica para la dirección del proyecto>* |
| **[Estado]** | *<estado dela restricción técnica según el ciclo de vida adoptado por el proyecto>* |
| **Comentarios** | *<comentarios adicionales sobre la restricción técnica>* |

Tabla 17: Restricciones técnicas del sistema.

## Requisitos de Integración del Sistema

<Introduzca contenido, cumplimente tabla y borre cuadro>

Esta sección debe contener los requisitos de integración que se hayan identificado, especificados mediante las plantillas para requisitos de integración que se muestran a continuación.

Estos requisitos deben identificar aquellos servicios disponibles en el entorno tecnológico de producción o componentes software (por ejemplo, librerías enlazables) cuya funcionalidad sea relevante para el sistema a desarrollar y deban ser consumidos por el mismo.

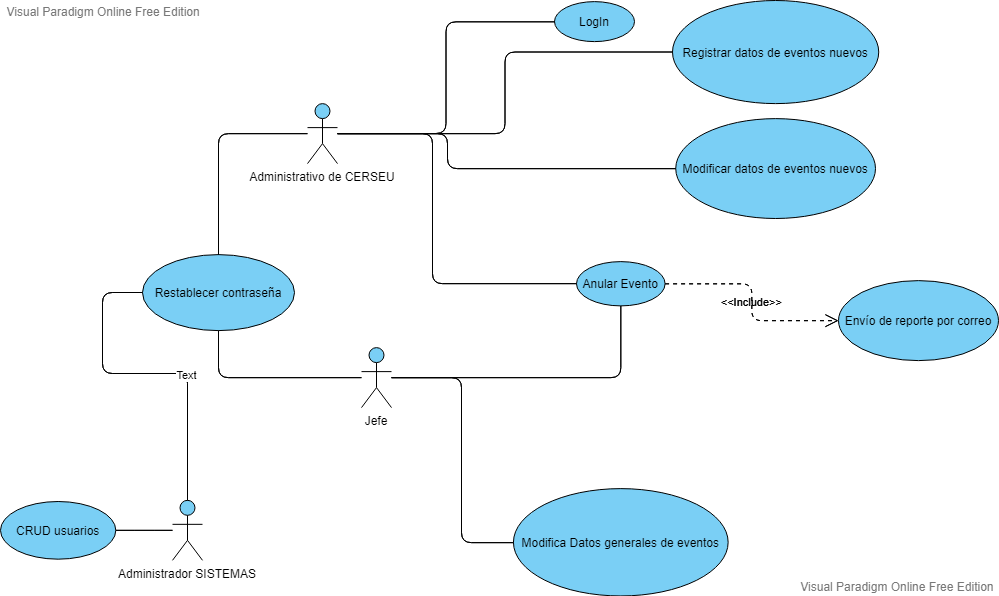
|  |  |
| --- | --- |
| **<id>999** | *<nombre descriptivo>* |
| **[Versión]** | *<nº versión>(<fecha de versión>)* |
| **[Dependencias]** | * *<requisitos generales de los que depende>* * *<otros requisitos de los que depende>* * *...* |
| **Descripción** | El sistema deberá utilizar el {servicio, componente software} *<nombre del elemento a integrar> para aquellos aspectos relacionados con <funcionalidad prestada por el elemento a integrar>* |
| **[Importancia]** | *<importancia del requisito para el cliente>* |
| **[Prioridad]** | *<prioridad del requisito para la dirección del proyecto>* |
| **[Estado]** | *<estado del requisito según el ciclo de vida adoptado por el proyecto>* |
| **Comentarios** | *<comentarios adicionales sobre el requisito>* |

Tabla 18: Requisitos de integración del sistema.

Los atributos entre corchetes son opcionales

## Actores del Sistema

## Casos de Uso del Sistema



|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Login* | | |
| **Precondición** | *La precondición es que haya cargado el formulario del login* | | |
| **Descripción** | *Se digita usuario y contraseña* | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El usuario digitador ingresa usuario y contraseña* | |
| 2 | Se validan los datos | |
|  | 3 | Se muestra resultados | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si no existe usuario o contraseña | |
| *…* |  |
| *E.m* | *Se imprime: “ Usuario y/o contraseña incorrecto”* |
| *…* | *...* | |
| **Frecuencia** | *1 vez* | | |
| **Comentarios** |  | | |

Tabla 12: Login

Diagram

Description automatically generated

Tabla 12: Diagrama de Secuencia de CU Login

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Parametrización de usuarios* | | |
| **Precondición** | *La precondición es que haya cargado el formulario del login* | | |
| **Descripción** | *Actualizar los datos de usuarios. El usuario que tenga rol ADMINISTRADOR podrá modificar otros usuarios.* | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El usuario digitador ingresa usuario y contraseña* | |
| 2 | Validación de contraseña admisible | |
|  | 3 | Cambio de contraseña | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si la contraseña no cumple con las políticas de seguridad | |
| *…* |  |
| *E.m* | *Se imprime: “Contraseña no válida”* |
| *…* | *...* | |
| **Frecuencia** | *N veces* | | |
| **Comentarios** |  | | |

Tabla 12: Parametrización de usuarios

CRUD de Actividades

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Registro de actividades* | | |
| **Precondición** | *La precondición es que haya cargado el formulario del login* | | |
| **Descripción** | sistema debera permitir registrar nuevos eventos de CERSEU | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El usuario digitador ingresa usuario y contraseña* | |
| 2 | Validación de contraseña admisible | |
|  | 3 | Cambio de contraseña | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si la contraseña no cumple con las políticas de seguridad | |
| *…* |  |
| *E.m* | *Se imprime: “Contraseña no válida”* |
| *…* | *...* | |
| **Frecuencia** | *N veces* | | |
| **Comentarios** |  | | |

Tabla 12: Registro de actividades

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Registro de actividades* | | |
| **Precondición** | *La precondición es que haya cargado el formulario del login* | | |
| **Descripción** | sistema debera permitir registrar nuevos eventos de CERSEU | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El usuario digitador ingresa usuario y contraseña* | |
| 2 | Validación de contraseña admisible | |
|  | 3 | Cambio de contraseña | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si la contraseña no cumple con las políticas de seguridad | |
| *…* |  |
| *E.m* | *Se imprime: “Contraseña no válida”* |
| *…* | *...* | |
| **Frecuencia** | *N veces* | | |
| **Comentarios** |  | | |

Tabla 12: Actualización de actividades

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Anulación de actividades* | | |
| **Precondición** | *La precondición es que haya cargado el formulario del login* | | |
| **Descripción** | El sistema debe anular eventos de CERSEU | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El usuario consulta el código de actividad que sea anular* | |
| 2 | El usuario presiona botón anular | |
|  | 3 | Actividad en estado N | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si la contraseña no cumple con las políticas de seguridad | |
| *…* |  |
| *E.m* | *Se imprime: “Contraseña no válida”* |
| *…* | *...* | |
| **Frecuencia** | *N veces* | | |
| **Comentarios** |  | | |

Tabla 12: Anulación de actividades

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Registrar Costos* | | |
| **Precondición** | *<precondición del caso de uso del sistema>* | | |
| **Descripción** | *Se registran los costos de un lote de a producir* | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El usuario digitador ingresa al sistema* | |
| 2 | Se registra un código y fecha de registro | |
|  | 3 | Se realizan los cálculos del CIF y del total | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si *<condición de excepción>* | |
| *…* |  |
| *E.m* | *{*El caso de uso continua,Se cancela el caso de uso*}* |
| *…* | *...* | |
| **Frecuencia** | *1 vez* | | |
| **Comentarios** |  | | |

Tabla 12: Registrar Costos

Chart

Description automatically generated with low confidence

Tabla 12: Diagrama de Secuencia de CU Registro de Costos

Diagram

Description automatically generated

Tabla 12: Diagrama de Actividades de CU Registrar Costos

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Actualizar Costos* | | |
| **Precondición** | *Debe existir un lote con el código digitado , de lo contrario se imprimirá el mensaje : “ No existen un lote con el código ingresado o se encuentra inactivo”.* | | |
| **Descripción** |  | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *{*El actor *<actor del sistema>,* El sistema*}<acción/es realizada/s por el actor del sistema>* | |
| 2 | Se realiza el *<caso de uso del sistema>* | |
| **Postcondición** | *<postcondición del caso de uso del sistema>* | | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si *<condición de excepción>* | |
| *…* |  |
| *E.m* | *{*El caso de uso continua,Se cancela el caso de uso*}* |
| *…* | *...* | |
| **Rendimiento** | **Paso** | **Cota de tiempo** | |
| *q* | *k<unidad de tiempo>* | |
| *…* | *...* | |
| **Frecuencia** | *1 vez* | | |
| **Comentarios** | *<comentarios adicionales sobre el caso de uso del sistema>* | | |

Tabla 12: Actualizar Costos

Diagram

Description automatically generated

Tabla 12: Diagrama de Actividades de CU Actualizar Costos

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Anular Costos* | | |
| **Precondición** | *Debe existir un lote con el código digitado , de lo contrario se imprimirá el mensaje : “ No existen un lote con el código ingresado o se encuentra inactivo”. El lote de producción debe existir y debe estar en estado ACTIVADO* | | |
| **Descripción** | *S* | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | *El digitado del área de ventas debe ingresar login* | |
| **Postcondición** | *Esta anulación debe enviar una notificación al jefe de área* | | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si *<condición de excepción>* | |
| *…* |  |
| *E.m* | *{*El caso de uso continua,Se cancela el caso de uso*}* |
| *…* | *...* | |
| **Comentarios** | *Este caso de uso es similar al caso de uso “Actualizar Costos”* | | |

Tabla 12: Anular Costos

Diagram

Description automatically generated

Tabla 12: Diagrama de Actividades de CU Anular Costos

|  |  |  |  |
| --- | --- | --- | --- |
| **<id>999** | *Generar Reporte de los costos* | | |
| **Precondición** | *El lote de producción debe existir y debe estar en estado ACTIVADO* | | |
| **Descripción** | *Los botones para generar reportes imprimirán un documento de Excel o pdf, según se desee.* | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | Click en el botón exportar | |
| 2 | El usuario selecciona la carpeta donde desea guardar el documento | |
|  | 3 | Se imprime el mensaje “documento exportado con éxito” | |
| **Postcondición** | *Ninguna* | | |
| **Excepciones** | **Paso** | **Acción** | |
| *P* | Si *ocurre un error al momento de exportar* | |
| *…* | *Se imprimirá el detalle del error* |
| **Comentarios** | *Se utilizará las librerías PDFSharp para generar PDF* | | |

Tabla 12: Generar Reporte de los costos

# Implementación del sistema

## Creación de la base de datos en SQL Server 2008

Posteriormente, se creó la base de datos en el motor SQL Server 2008.Se crearon tablas[[2]](#footnote-2). El diagrama de relaciones en SQL Server Management Studio (Figura 1) detalla los campos de cada tabla.

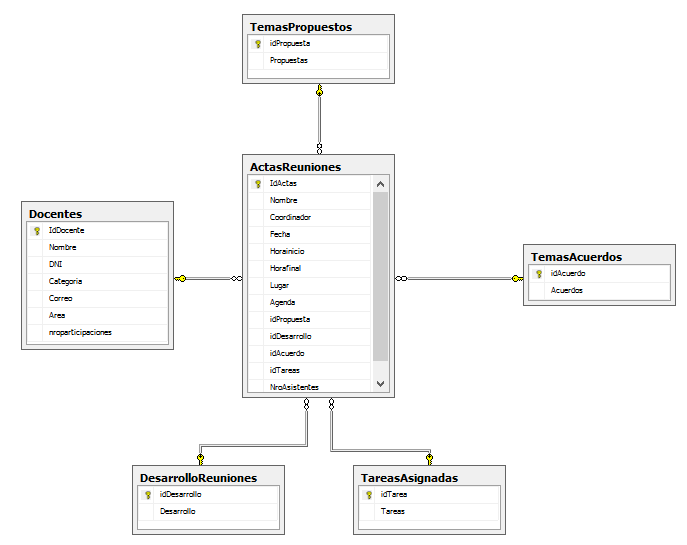


Figura 2. Diagrama de relaciones en SQL Server Management Studio (Elaboración propia)

# Análisis de resultados

## Codificación del programa

Es el proceso de trascripción del diseño funcional realizado en UML (Diagramas de Casos de Usos, Diagramas de Secuencia) a Visual C#.NET, para la creación de las interfaces, diseño y pruebas del monitoreo del aplicativo.

## Banco de pruebas

Las pruebas realizadas al desarrollo al finalizar fueron pruebas de caja negra debido a la carencia de un agente de pruebas de funcionamiento del desarrollo y una entidad de implementación del desarrollo, donde se harían pruebas de integración.

### Pruebas de función

Objetivo: Verificar que cada una de las funciones cumpla con el propósito para el cual fueron diseñados.

Criterio de decisión: De acuerdo a los requerimientos internacionales que exige el modelo del CVDS, dentro de su primera prueba de funcionalidad, exige la verificación de la validación de los caracteres de digitación, entre ellos están las letras, los números y los espacios en blanco en cada uno de los campos que se consideran obligatorios en el diseño de los respectivos formularios.

|  |  |  |
| --- | --- | --- |
| PRUEBAS | SÍ | NO |
| La función de validación de letras es correcta | X |  |
| La función de validación de números es correcta | X |  |
| La rutina de validación de espacios en blanco es correcta | X |  |

### Pruebas modulares

Objetivo: Detección de errores y verificación de la integración de los menús.

Criterio de decisión: Las pruebas modulares corresponden al segundo ítem mencionado en la aplicación de pruebas de la metodología del CVDS, y su función consiste en evaluar cada una de las rutinas que se integran al todo de la aplicación y que exista interacción adecuada con el usuario.

**MENÚ APLICACIÓN**

|  |  |  |
| --- | --- | --- |
| PRUEBAS | SÍ | NO |
| Se generan errores al momento de cargar un Submenú |  | X |
| Los Submenús son complejos |  | X |
| La interacción con el usuario es fácil | X |  |
| La selección de menús y submenús funciona | X |  |
| Realiza el proceso de inserción de nuevo entorno | X |  |
| Realiza el proceso de salir del entorno actual | X |  |

El siguiente pantallazo muestra la funcionalidad de los menús y los submenús los cuales son de fácil acceso al usuario y divididos según la funcionalidad a solicitar, anexo a los submenús están un cuadro de botones que tienen un texto de ayuda para saber a qué funcionalidad se está accediendo al hacer clic en el botón.

### Pruebas de documentación y ayuda

Objetivo: Verificar el contenido de la documentación, manuales y ayudas.

Criterio de decisión: Para las pruebas de documentación, igualmente se tomó la referencia de la metodología del CVDS, en donde el código debe ser explicativo como referencia para que el desarrollador se ubique dentro del entorno del análisis.

|  |  |  |
| --- | --- | --- |
| PRUEBAS | SÍ | NO |
| Existe una documentación interna (código) | X |  |
| Los nombres de las variables y etiquetas son significativos | X |  |
| Los comentarios son fáciles de entender | X |  |
| Existe una documentación externa | X |  |
| La ayuda es fácil de entender y manejar | X |  |

### Pruebas de seguridad y control

Objetivo: Verificación del control de acceso, validación de información y la realización correcta de las acciones sobre la aplicación.

|  |  |  |
| --- | --- | --- |
| PRUEBAS | SÍ | NO |
| Existe perfiles de usuario para el funcionamiento del sistema | X |  |
| Solicita contraseña al ingresar a la aplicación | X |  |

### Pruebas de calidad

Objetivo: Verificar la capacidad operativa y de adaptación a los cambios en la aplicación.

|  |  |  |
| --- | --- | --- |
| PRUEBAS | SÍ | NO |
| Existe portabilidad |  | X |
| El código es reutilizable | X |  |
| Es fácil de utilizar | X |  |
| Existe integridad | X |  |

## Informe de pruebas

* Comprobar la validación de información.
* Comprobar la lógica de la aplicación.
* Comprobar el nivel de integración de los menús.
* Probar el manejo de errores.

## Material revisado

**Formulario principal**

**Menú Archivos**

**Menú Docentes**

# Conclusiones y Recomendaciones

* Las herramientas de trabajo Visual C# .NET 2015 y SQL server 2010 Express, se interrelacionan y complementan bien en el desarrollo de proyectos de software, generando confianza en el proceso de construcción, existe buena documentación con respecto a ambas en Internet y por parte de Microsoft, son de fácil uso y manejo, el reporte de errores y al documentación de ellos es bastante buena y hace que el trabajo del desarrollador sea ágil al encontrar solución rápida y oportuna a los inconvenientes que se puedan presentar.
* Se propone este sistema, en aras de mejorar con la calidad educativa en distintas facultades de la universidad peruana.
* La experiencia específica en la UNMSM señala que en diversas facultades no existe documentación de reuniones de los docentes.
* Se recomienda, reutilizar los sistemas informáticos en las distintas facultades (Por ejemplo, Campus Virtual FII).

# Bibliografía

AEC.*Asociación Española para la Calidad*. Obtenido de Modelos de Calidad: https://www.aec.es/web/guest/centro-conocimiento/modelos-de-calidad

Ferguson,J. Patterson,B.Beres, J.Boutquin,P.Gupta,M. (2003). La Biblia de C# . Edit Anaya Multimedia. Madrid,España.

Kendall, K.Kendall,J.(2005). Análisis y diseño de sistemas.6ta. Ed., Edit. Pearson, México.

MINEDU.*Ministerio de Educación.* Obtenido de Estándares de Calidad Educativa: https://educacion.gob.ec/wp-content/uploads/downloads/2013/03/estandares\_2012.pdf

Parodi, C. (2013). Calidad de la Educación en el Perú.Diario *Gestión*.

SINEACE. (2016). *SINEACE.* Obtenido de Modelo de Acreditación para Programas de: https://www.sineace.gob.pe/wp-content/uploads/2014/08/Anexo-1-nuevo-modelo-programas-Resolucion-175.pdf

# Anexos

## Acta de reunión ( Meeting)

|  |  |  |  |
| --- | --- | --- | --- |
| **ACTA No. 24**  **COMITÉ DE CALIDAD** | **FECHA:**  **8 de julio de 2017** | **HORA INICIO**: 8 a.m.  **HORATERMINACIÓN:**  **10: 10 a.m.** | **LUGAR: FII. Piso 2°. Salón N° 1.** |
| **OBJETIVO DE LA REUNIÓN:**   * 1. A   2. B   3. C | | | |
| ***COORDINADOR DE LA REUNION:*** *Ing.Wiler Ponce* | | | |

CONVOCADOS / ASISTENTES

|  |  |  |  |
| --- | --- | --- | --- |
| **NOMBRES Y APELLIDOS** | **FIRMA** | **ASISTIO** | |
| **SI** | **NO** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |
| --- |
| **AGENDA** |
| 1. Verificación del quórum. 2. Lectura y aprobación del acta anterior. 3. Presentación situación actual del problema a tratar 4. Propuesta cronograma. 5. Revisión y aprobación de procedimientos. 6. Proposiciones y varios. |

|  |
| --- |
| **DESARROLLO DE LA AGENDA** |
| **1. Verificación del quórum.**  **2. Lectura y aprobación de las Actas # 22 y # 23 del 24 y 25 de Junio de 2017.**  SITUACIÓN ACTUAL:  PLAN DE ACCION PROPUESTO DEL PROBLEMA X:  Estrategia 1:  Estrategia 2:  Estrategia 3:  Estrategia 4:  **6. Revisión** y aprobación de procedimientos.  **7.** Proposiciones y varios. |
| **ACUERDOS** |

**TAREAS Y COMPROMISOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **TAREA / ACTIVIDAD** | **RESPONSABLE** | ***ENTREGA*** |
| **1.** |  | *Dr.Omar Álvarez* | *Inmediata.* |

Fecha de aprobación del acta: Julio de 2017

ELABORÓ: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

REVISÓ: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

En constancia firma:

|  |  |
| --- | --- |
| *NOMBRE – ROL* | *NOMBRE – ROL* |
| ***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*** | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* |
| *XXXXX* | *XXXX* |
| *Coordinador* | *Director* |

## Acta de actividad de extensión ( CERSEU)

**ESCUELA:**

**CURSO:**

**CICLO:**

**AULA: FECHA:**

**PROFESOR:**

1. **NOMBRE DE ACTIVIDAD:**

**RESUMEN DE ACTIVIDAD A REALIZAR:**

**RESULTADOS ESPERADOS:**

**INTEGRANTES DEL GRUPO:**

|  |  |
| --- | --- |
| **NOMBRE DEL ALUMNO** | **CÓDIGO** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Base de datos

## Scripts de creación de base de datos

USE [master]

GO

/\*\*\*\*\*\* Object: Database [bdgeneric] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

CREATE DATABASE [bdgeneric]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'bdcongreso', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLSERVER2019\MSSQL\DATA\bdcongreso.mdf' , SIZE = 2097152KB , MAXSIZE = 8388608KB , FILEGROWTH = 2097152KB )

LOG ON

( NAME = N'bdcongreso\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLSERVER2019\MSSQL\DATA\bdcongreso\_log.ldf' , SIZE = 1048576KB , MAXSIZE = 2097152KB , FILEGROWTH = 10%)

WITH CATALOG\_COLLATION = DATABASE\_DEFAULT

GO

ALTER DATABASE [bdgeneric] SET COMPATIBILITY\_LEVEL = 140

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [bdgeneric].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [bdgeneric] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [bdgeneric] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [bdgeneric] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [bdgeneric] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [bdgeneric] SET ARITHABORT OFF

GO

ALTER DATABASE [bdgeneric] SET AUTO\_CLOSE OFF

GO

ALTER DATABASE [bdgeneric] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [bdgeneric] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [bdgeneric] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [bdgeneric] SET CURSOR\_DEFAULT GLOBAL

GO

ALTER DATABASE [bdgeneric] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [bdgeneric] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [bdgeneric] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [bdgeneric] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [bdgeneric] SET DISABLE\_BROKER

GO

ALTER DATABASE [bdgeneric] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [bdgeneric] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [bdgeneric] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [bdgeneric] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [bdgeneric] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [bdgeneric] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [bdgeneric] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [bdgeneric] SET RECOVERY SIMPLE

GO

ALTER DATABASE [bdgeneric] SET MULTI\_USER

GO

ALTER DATABASE [bdgeneric] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [bdgeneric] SET DB\_CHAINING OFF

GO

ALTER DATABASE [bdgeneric] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [bdgeneric] SET TARGET\_RECOVERY\_TIME = 60 SECONDS

GO

ALTER DATABASE [bdgeneric] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [bdgeneric] SET ACCELERATED\_DATABASE\_RECOVERY = OFF

GO

EXEC sys.sp\_db\_vardecimal\_storage\_format N'bdgeneric', N'ON'

GO

ALTER DATABASE [bdgeneric] SET QUERY\_STORE = OFF

GO

USE [bdgeneric]

GO

/\*\*\*\*\*\* Object: Table [dbo].[actividades] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[actividades](

[idactividad] [int] IDENTITY(1,1) NOT NULL,

[codigo] [nchar](10) NOT NULL,

[nombre] [nchar](30) NOT NULL,

[descripcion] [nvarchar](100) NOT NULL,

[fecha\_creacion] [datetime] NULL,

[rol\_creacion] [nchar](10) NOT NULL,

[estado] [nchar](2) NULL,

[idlistaeq] [int] NULL,

[idorganiz] [int] NULL,

[idsala] [int] NULL,

[fec\_inicio] [datetime] NULL,

[fec\_fin] [datetime] NULL,

[responsable] [nchar](50) NULL,

[lugar] [nvarchar](200) NULL,

[justificacion] [nvarchar](100) NULL,

[objetivos] [nvarchar](100) NULL,

[horas] [int] NULL,

[idambiente] [int] NULL,

[idsede] [int] NULL,

[inscripcion] [nchar](2) NULL,

[certificado] [nchar](2) NULL,

[tipoevento] [nchar](30) NULL,

CONSTRAINT [PK\_\_activida\_\_7E768FFAD3BCB078] PRIMARY KEY CLUSTERED

(

[idactividad] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[alumnos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[alumnos](

[idalumno] [int] IDENTITY(1,1) NOT NULL,

[ap\_paterno] [varchar](30) NOT NULL,

[ap\_materno] [varchar](30) NOT NULL,

[nombre] [nvarchar](30) NOT NULL,

[DNI] [nchar](8) NOT NULL,

[codigo] [nchar](8) NULL,

[telefono] [nchar](20) NULL,

[sexo] [nchar](1) NOT NULL,

[correo] [nvarchar](80) NOT NULL,

[carrera] [nvarchar](80) NULL,

[direccion] [nvarchar](80) NULL,

[estado] [nchar](2) NULL,

[tipo\_alumno] [nchar](10) NOT NULL,

[idusuario] [int] NULL,

[fec\_nac] [date] NULL,

[correo\_2] [nvarchar](80) NULL,

CONSTRAINT [PK\_\_particip\_\_8E1C3FB4CF6198CD] PRIMARY KEY CLUSTERED

(

[idalumno] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY],

CONSTRAINT [IX\_participantes] UNIQUE NONCLUSTERED

(

[DNI] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ambiente] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ambiente](

[idambiente] [int] IDENTITY(1,1) NOT NULL,

[nombre] [varchar](20) NOT NULL,

[direccion] [nvarchar](100) NOT NULL,

[estado] [nchar](3) NULL,

PRIMARY KEY CLUSTERED

(

[idambiente] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[aud\_audit\_modif] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[aud\_audit\_modif](

[id] [int] IDENTITY(1,1) NOT NULL,

[codigo] [nchar](20) NULL,

[observaciones] [nvarchar](300) NULL,

[tabla] [nchar](30) NULL,

[valor\_antes] [nvarchar](100) NULL,

[valor\_despues] [nvarchar](100) NULL,

[usuario] [varchar](50) NULL,

[usuario\_bd] [varchar](100) NULL,

[fecha] [datetime] NULL,

PRIMARY KEY CLUSTERED

(

[id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ciclo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ciclo](

[IdCiclo] [int] IDENTITY(1,1) NOT NULL,

[FecInicio] [datetime] NOT NULL,

[FecTermino] [datetime] NOT NULL,

[estado] [nchar](1) NULL,

CONSTRAINT [PK\_ciclo] PRIMARY KEY CLUSTERED

(

[IdCiclo] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[costos\_prod\_imprenta] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[costos\_prod\_imprenta](

[idprodimprenta] [int] IDENTITY(1,1) NOT NULL,

[codigo] [nchar](10) NOT NULL,

[nro\_libros] [int] NOT NULL,

[nro\_paginas] [int] NOT NULL,

[largo] [decimal](18, 2) NOT NULL,

[ancho] [decimal](18, 2) NOT NULL,

[nro\_pliegos] [int] NOT NULL,

[nro\_resmas] [int] NOT NULL,

[nro\_placas] [int] NOT NULL,

[estado] [nchar](2) NULL,

[doblez] [decimal](18, 2) NOT NULL,

[alce] [decimal](18, 2) NOT NULL,

[fecha\_creacion] [date] NULL,

[corte\_refilado] [decimal](18, 2) NOT NULL,

[encolado] [decimal](18, 2) NOT NULL,

[movilidad] [decimal](18, 2) NOT NULL,

[manejo\_archivo] [decimal](18, 2) NOT NULL,

[best\_color] [decimal](18, 2) NOT NULL,

[cif] [decimal](18, 2) NOT NULL,

[total] [decimal](18, 2) NOT NULL,

[tinta] [decimal](10, 2) NULL,

[titulo] [nchar](50) NULL,

CONSTRAINT [PK\_\_costos\_p\_\_1CD69F466FC07AE5] PRIMARY KEY CLUSTERED

(

[idprodimprenta] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[curso] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[curso](

[idcurso] [int] NOT NULL,

[IdTarifa] [int] IDENTITY(1,1) NOT NULL,

[malla] [nchar](5) NULL,

[nombreCurso] [varchar](80) NOT NULL,

[estado] [nchar](1) NOT NULL,

[idCiclo] [int] NULL,

[tipo] [nchar](2) NULL,

CONSTRAINT [PK\_\_curso\_\_085F27D6ED07770A] PRIMARY KEY CLUSTERED

(

[idcurso] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[cursoProgramado] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[cursoProgramado](

[idcursoprog] [int] IDENTITY(1,1) NOT NULL,

[idcurso] [int] NOT NULL,

[Vacantes] [numeric](3, 0) NOT NULL,

[Matriculados] [numeric](3, 0) NOT NULL,

[PreCursoProg] [money] NOT NULL,

[Horario] [varchar](30) NOT NULL,

[Activo] [numeric](1, 0) NOT NULL

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[detalle\_list\_equipos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[detalle\_list\_equipos](

[intdetalleeq] [int] IDENTITY(1,1) NOT NULL,

[estado] [nchar](2) NULL,

[idEquipo] [int] NULL,

[codigo] [nchar](10) NOT NULL,

CONSTRAINT [PK\_\_detalle\_\_\_4D0657328F738DC1] PRIMARY KEY CLUSTERED

(

[intdetalleeq] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[detalle\_organizadores] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[detalle\_organizadores](

[iddetorg] [int] IDENTITY(1,1) NOT NULL,

[estado] [nchar](2) NULL,

[DNI] [nchar](8) NOT NULL,

[codigo] [nchar](5) NOT NULL,

CONSTRAINT [PK\_detalle\_organizadores] PRIMARY KEY CLUSTERED

(

[iddetorg] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[empleado] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[empleado](

[IdEmpleado] [char](6) NOT NULL,

[Rol] [nchar](4) NOT NULL,

[ApePatEmpleado] [nvarchar](30) NOT NULL,

[ApeMatEmpleado] [nvarchar](30) NOT NULL,

[NomEmpleado] [nvarchar](40) NOT NULL,

[Cargo] [varchar](25) NOT NULL,

[DirEmpleado] [varchar](100) NULL,

[TelEmpleado] [varchar](10) NULL,

[EmailEmpleado] [varchar](30) NOT NULL,

[Fecha\_ini] [date] NOT NULL,

[Estado] [nchar](1) NOT NULL,

[idusuario] [int] NULL

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[equipo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[equipo](

[idEquipo] [int] IDENTITY(1,1) NOT NULL,

[Nombre] [nchar](20) NULL,

[SO] [nchar](10) NULL,

[Procesador] [nchar](30) NULL,

[RAM] [nchar](10) NULL,

[TarjetaMadre] [nchar](10) NULL,

[estado] [nchar](2) NOT NULL,

CONSTRAINT [PK\_Equipo] PRIMARY KEY CLUSTERED

(

[idEquipo] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[listaequipos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[listaequipos](

[idlistaeq] [int] IDENTITY(1,1) NOT NULL,

[estado] [nchar](2) NOT NULL,

[codigo] [nchar](10) NOT NULL,

CONSTRAINT [PK\_\_listaequ\_\_B3879B681BE2E6FA] PRIMARY KEY CLUSTERED

(

[idlistaeq] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY],

CONSTRAINT [IX\_listaequipos\_1] UNIQUE NONCLUSTERED

(

[codigo] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[organizadores] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[organizadores](

[idorganiz] [int] IDENTITY(1,1) NOT NULL,

[codigo] [nchar](10) NULL,

[estado] [nchar](2) NULL,

[fec\_registro] [date] NULL,

[iddetorg] [int] NULL,

CONSTRAINT [PK\_\_asistenc\_\_4B2DBC1D9FD9EBFF] PRIMARY KEY CLUSTERED

(

[idorganiz] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[parametros] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[parametros](

[idparametros] [int] IDENTITY(1,1) NOT NULL,

[nombre] [nchar](20) NOT NULL,

[codigo] [nchar](20) NOT NULL,

[concepto] [nchar](20) NOT NULL,

[valor] [decimal](10, 2) NOT NULL,

[estado] [nchar](1) NULL,

PRIMARY KEY CLUSTERED

(

[idparametros] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[sala] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[sala](

[idsala] [int] IDENTITY(1,1) NOT NULL,

[nombre] [nvarchar](80) NOT NULL,

[ubicacion] [nvarchar](80) NOT NULL,

[capacidad] [int] NULL,

[fecha\_creacion] [date] NULL,

[estado] [nchar](2) NULL,

[rol\_creacion] [nchar](10) NOT NULL,

[tipo\_sala] [char](10) NULL,

PRIMARY KEY CLUSTERED

(

[idsala] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[sede] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[sede](

[idsede] [int] IDENTITY(1,1) NOT NULL,

[nombre] [varchar](20) NOT NULL,

[direccion] [nvarchar](100) NOT NULL,

[estado] [nchar](3) NULL,

PRIMARY KEY CLUSTERED

(

[idsede] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tarifa] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tarifa](

[IdTarifa] [int] IDENTITY(1,1) NOT NULL,

[PrecioVenta] [money] NOT NULL,

[Descripcion] [varchar](50) NOT NULL,

[Horas] [int] NOT NULL,

[PagoHora] [money] NOT NULL,

CONSTRAINT [PK\_Tarifa] PRIMARY KEY CLUSTERED

(

[IdTarifa] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[usuarios] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[usuarios](

[idusuario] [int] IDENTITY(1,1) NOT NULL,

[usuario] [nchar](10) NULL,

[clave] [nchar](10) NULL,

[estado] [nchar](2) NULL,

[foto] [image] NULL,

PRIMARY KEY CLUSTERED

(

[idusuario] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[actividades] ADD CONSTRAINT [DF\_\_actividad\_\_fecha\_\_51EF2864] DEFAULT (getdate()) FOR [fecha\_creacion]

GO

ALTER TABLE [dbo].[actividades] ADD CONSTRAINT [DF\_\_actividad\_\_estad\_\_52E34C9D] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[alumnos] ADD CONSTRAINT [DF\_\_participa\_\_telef\_\_3CF40B7E] DEFAULT ('--') FOR [telefono]

GO

ALTER TABLE [dbo].[alumnos] ADD CONSTRAINT [DF\_\_participa\_\_estad\_\_3DE82FB7] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[ambiente] ADD DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[ciclo] ADD CONSTRAINT [DF\_\_ciclo\_\_estado\_\_3C89F72A] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[costos\_prod\_imprenta] ADD CONSTRAINT [DF\_\_costos\_pr\_\_estad\_\_6A85CC04] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[costos\_prod\_imprenta] ADD CONSTRAINT [DF\_\_costos\_pr\_\_fecha\_\_6B79F03D] DEFAULT (getdate()) FOR [fecha\_creacion]

GO

ALTER TABLE [dbo].[curso] ADD CONSTRAINT [DF\_curso\_estado] DEFAULT (N'S') FOR [estado]

GO

ALTER TABLE [dbo].[cursoProgramado] ADD CONSTRAINT [DF\_\_CursoProg\_\_Vacan\_\_2B2A60FE] DEFAULT ((20)) FOR [Vacantes]

GO

ALTER TABLE [dbo].[cursoProgramado] ADD CONSTRAINT [DF\_\_CursoProg\_\_Matri\_\_2C1E8537] DEFAULT ((0)) FOR [Matriculados]

GO

ALTER TABLE [dbo].[cursoProgramado] ADD CONSTRAINT [DF\_\_CursoProg\_\_Activ\_\_2D12A970] DEFAULT ((1)) FOR [Activo]

GO

ALTER TABLE [dbo].[detalle\_list\_equipos] ADD CONSTRAINT [DF\_detalle\_list\_equipos\_estado] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[detalle\_organizadores] ADD CONSTRAINT [DF\_\_detalle\_a\_\_estad\_\_6F7F8B4B] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[empleado] ADD DEFAULT ('S') FOR [Estado]

GO

ALTER TABLE [dbo].[equipo] ADD CONSTRAINT [DF\_\_Equipo\_\_estado\_\_4B422AD5] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[listaequipos] ADD CONSTRAINT [DF\_\_listaequi\_\_estad\_\_4E1E9780] DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[sala] ADD DEFAULT ((0)) FOR [capacidad]

GO

ALTER TABLE [dbo].[sala] ADD DEFAULT (getdate()) FOR [fecha\_creacion]

GO

ALTER TABLE [dbo].[sala] ADD DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[sede] ADD DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[usuarios] ADD DEFAULT ('S') FOR [estado]

GO

ALTER TABLE [dbo].[actividades] WITH CHECK ADD CONSTRAINT [FK\_\_actividad\_\_idamb\_\_5C02A283] FOREIGN KEY([idambiente])

REFERENCES [dbo].[ambiente] ([idambiente])

GO

ALTER TABLE [dbo].[actividades] CHECK CONSTRAINT [FK\_\_actividad\_\_idamb\_\_5C02A283]

GO

ALTER TABLE [dbo].[actividades] WITH CHECK ADD CONSTRAINT [FK\_\_actividad\_\_idasi\_\_54CB950F] FOREIGN KEY([idorganiz])

REFERENCES [dbo].[organizadores] ([idorganiz])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[actividades] CHECK CONSTRAINT [FK\_\_actividad\_\_idasi\_\_54CB950F]

GO

ALTER TABLE [dbo].[actividades] WITH CHECK ADD CONSTRAINT [FK\_\_actividad\_\_idsal\_\_55BFB948] FOREIGN KEY([idsala])

REFERENCES [dbo].[sala] ([idsala])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[actividades] CHECK CONSTRAINT [FK\_\_actividad\_\_idsal\_\_55BFB948]

GO

ALTER TABLE [dbo].[actividades] WITH CHECK ADD CONSTRAINT [FK\_\_actividad\_\_idsed\_\_5CF6C6BC] FOREIGN KEY([idsede])

REFERENCES [dbo].[sede] ([idsede])

GO

ALTER TABLE [dbo].[actividades] CHECK CONSTRAINT [FK\_\_actividad\_\_idsed\_\_5CF6C6BC]

GO

ALTER TABLE [dbo].[actividades] WITH CHECK ADD CONSTRAINT [FK\_actividades\_listaequipos] FOREIGN KEY([idlistaeq])

REFERENCES [dbo].[listaequipos] ([idlistaeq])

GO

ALTER TABLE [dbo].[actividades] CHECK CONSTRAINT [FK\_actividades\_listaequipos]

GO

ALTER TABLE [dbo].[alumnos] WITH CHECK ADD CONSTRAINT [FK\_\_participa\_\_idusu\_\_7FB5F314] FOREIGN KEY([idusuario])

REFERENCES [dbo].[usuarios] ([idusuario])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[alumnos] CHECK CONSTRAINT [FK\_\_participa\_\_idusu\_\_7FB5F314]

GO

ALTER TABLE [dbo].[curso] WITH CHECK ADD CONSTRAINT [FK\_\_curso\_\_idCiclo\_\_7C6F7215] FOREIGN KEY([idCiclo])

REFERENCES [dbo].[ciclo] ([IdCiclo])

GO

ALTER TABLE [dbo].[curso] CHECK CONSTRAINT [FK\_\_curso\_\_idCiclo\_\_7C6F7215]

GO

ALTER TABLE [dbo].[curso] WITH CHECK ADD CONSTRAINT [FK\_\_curso\_\_IdTarifa\_\_7D63964E] FOREIGN KEY([IdTarifa])

REFERENCES [dbo].[tarifa] ([IdTarifa])

GO

ALTER TABLE [dbo].[curso] CHECK CONSTRAINT [FK\_\_curso\_\_IdTarifa\_\_7D63964E]

GO

ALTER TABLE [dbo].[cursoProgramado] WITH CHECK ADD FOREIGN KEY([idcurso])

REFERENCES [dbo].[curso] ([idcurso])

GO

ALTER TABLE [dbo].[detalle\_list\_equipos] WITH CHECK ADD CONSTRAINT [FK\_detalle\_list\_equipos\_Equipo] FOREIGN KEY([idEquipo])

REFERENCES [dbo].[equipo] ([idEquipo])

GO

ALTER TABLE [dbo].[detalle\_list\_equipos] CHECK CONSTRAINT [FK\_detalle\_list\_equipos\_Equipo]

GO

ALTER TABLE [dbo].[detalle\_list\_equipos] WITH CHECK ADD CONSTRAINT [FK\_detalle\_list\_equipos\_listaequipos] FOREIGN KEY([codigo])

REFERENCES [dbo].[listaequipos] ([codigo])

GO

ALTER TABLE [dbo].[detalle\_list\_equipos] CHECK CONSTRAINT [FK\_detalle\_list\_equipos\_listaequipos]

GO

ALTER TABLE [dbo].[detalle\_organizadores] WITH CHECK ADD CONSTRAINT [FK\_dni] FOREIGN KEY([DNI])

REFERENCES [dbo].[alumnos] ([DNI])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[detalle\_organizadores] CHECK CONSTRAINT [FK\_dni]

GO

ALTER TABLE [dbo].[empleado] WITH CHECK ADD FOREIGN KEY([idusuario])

REFERENCES [dbo].[usuarios] ([idusuario])

GO

ALTER TABLE [dbo].[organizadores] WITH CHECK ADD CONSTRAINT [FK\_\_organizad\_\_iddet\_\_71F1E3A2] FOREIGN KEY([iddetorg])

REFERENCES [dbo].[detalle\_organizadores] ([iddetorg])

GO

ALTER TABLE [dbo].[organizadores] CHECK CONSTRAINT [FK\_\_organizad\_\_iddet\_\_71F1E3A2]

GO

ALTER TABLE [dbo].[cursoProgramado] WITH CHECK ADD CONSTRAINT [CK\_\_CursoProg\_\_Activ\_\_2E06CDA9] CHECK (([Activo]=(1) OR [Activo]=(0)))

GO

ALTER TABLE [dbo].[cursoProgramado] CHECK CONSTRAINT [CK\_\_CursoProg\_\_Activ\_\_2E06CDA9]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_\_parametros\_valor\_xnombre] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_\_parametros\_valor\_xnombre]

--cabecera

@nombre nchar(10)

AS

SELECT \*

FROM parametros

WHERE nombre= @nombre

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_actividad\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_actividad\_codigo]

@id int,

@descripcion nvarchar(80),

@idlistaeq int,

@idsala int,

@idorganiz int,

@Nombre nchar(30),

@fec\_inicio datetime,

@fec\_fin datetime,

@responsable nchar(50),

@codigo nchar(10)

AS

UPDATE actividades

SET descripcion = @descripcion,

Nombre = @Nombre,

idorganiz = @idorganiz,

idlistaeq = @idlistaeq,

idsala = @idsala,

fec\_inicio = @fec\_inicio,

fec\_Fin = @fec\_fin,

responsable = @responsable

where ( codigo = @codigo and estado ='S')

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_actividad\_id] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_actividad\_id]

@id int,

@descripcion nvarchar(80),

@idlistaeq int,

@idsala int,

@idorganiz int,

@Nombre nchar(5),

@fec\_inicio datetime,

@fec\_fin datetime,

@responsable nchar(50)

AS

UPDATE actividades

SET descripcion = @descripcion,

Nombre = @Nombre,

idorganiz = @idorganiz,

idlistaeq = @idlistaeq,

idsala = @idsala,

fec\_inicio = @fec\_inicio,

fec\_Fin = @fec\_fin,

responsable = @responsable

where ( idactividad = @id and estado ='S')

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_alumno\_dni] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_alumno\_dni]

@ap\_paterno varchar(30),

@ap\_materno varchar(30),

@nombre varchar(30),

@telefono nchar(20),

@sexo nchar(3),

@correo nvarchar(80),

@DNI nchar(8),

@dirrecion nvarchar(80),

@tipo nchar(10)

AS

UPDATE alumnos

SET ap\_paterno = @ap\_paterno,

ap\_materno = @ap\_materno,

nombre = @nombre,

telefono = @telefono,

sexo = @sexo,

correo = @correo,

DNI = @DNI,

direccion = @dirrecion,

tipo\_alumno = @tipo

where ( DNI = @DNI and estado ='S')

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_costosimprenta] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_costosimprenta]

@codigo nchar(10),

@nro\_libros int,

@nro\_paginas int,

@largo decimal,

@ancho decimal,

@nro\_pliegos int,

@nro\_resmas int,

@nro\_placas int,

@doblez decimal,

@alce decimal,

@corte\_refilado decimal,

@encolado decimal,

@movilidad decimal,

@manejo\_archivo decimal,

@best\_color decimal,

@cif decimal,

@tinta decimal(10,2),

@total decimal,

@titulo NCHAR(50)

AS

UPDATE costos\_prod\_imprenta

SET nro\_libros =@nro\_libros, nro\_paginas =@nro\_paginas,

largo =@largo, ancho =@ancho, nro\_pliegos =@nro\_pliegos, nro\_resmas =@nro\_resmas,

nro\_placas =@nro\_placas, doblez =@doblez, alce =@alce, corte\_refilado =@corte\_refilado, encolado =@encolado, movilidad =@movilidad, manejo\_archivo =@manejo\_archivo, best\_color =@best\_color, cif =@cif, total =@total, tinta =@tinta, titulo = @titulo

where (codigo = @codigo)

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_equipo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_equipo]

@idEquipo int,

@Nombre nchar(20),

@SO nchar(10),

@Procesador nchar(30),

@RAM nchar(10),

@TarjetaMadre nchar(10)

AS

UPDATE Equipo

SET Nombre = @Nombre,

SO = @SO,

Procesador = @Procesador,

RAM = @RAM,

TarjetaMadre = @TarjetaMadre

where (idEquipo = @idEquipo

and estado ='S')

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_sala\_id] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_sala\_id]

@idsala int,

@nombre nvarchar(80),

@ubicacion nvarchar(80),

@capacidad int,

@tipo\_sala char(10)

AS

UPDATE sala

SET nombre = @nombre,

ubicacion = @ubicacion,

capacidad = @capacidad,

tipo\_sala = @tipo\_sala

where ( idsala = @idsala)

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_usuario] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_usuario]

@usuario nchar(10),

@clave nchar(10)

AS

UPDATE usuarios

SET usuario = @usuario,

clave = @clave

where (usuario = @usuario)

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_usuario\_user] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_usuario\_user]

@usuario nchar(10),

@clave nchar(10),

@foto image

AS

UPDATE usuarios

SET clave = @clave,

foto = @foto

where ( usuario = @usuario and estado ='S')

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_actualizar\_valor\_param] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_actualizar\_valor\_param]

@valor decimal(10,2),

@nombre nchar(20)

AS

UPDATE parametros

SET valor = @valor

where (nombre = @nombre and estado ='S')

RETURN 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_borrar\_det\_list\_equipos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_borrar\_det\_list\_equipos]

-- @DNI NCHAR(8),

@codigo NCHAR(5)

AS

delete detalle\_list\_equipos where codigo = @codigo

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_borrar\_det\_organizadores] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_borrar\_det\_organizadores]

-- @DNI NCHAR(8),

@codigo NCHAR(5)

AS

delete detalle\_organizadores where codigo = @codigo

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_buscar\_actividad\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_buscar\_actividad\_codigo]

@codigo nchar(10)

AS

SELECT \*

FROM actividades

WHERE codigo= @codigo

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_buscar\_alumno\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_buscar\_alumno\_codigo]

@codigo nchar(8)

AS

SELECT \*

FROM alumnos

WHERE estado ='S'

and codigo =@codigo

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_buscar\_alumno\_dni] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_buscar\_alumno\_dni]

@DNI nchar(8)

AS

SELECT \*

FROM alumnos

WHERE estado ='S'

and DNI =@DNI

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_buscar\_alumno\_id] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_buscar\_alumno\_id]

@id int

AS

SELECT \*

FROM alumnos

WHERE estado ='S'

and idalumno =@id

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_buscar\_usuario\_nombre] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Raúl Armas

-- Create date: 03/11/19

-- Description: Lista todas las actividades

-- =============================================

CREATE PROCEDURE [dbo].[usp\_buscar\_usuario\_nombre]

@usuario nchar(10)

AS

BEGIN

select \* from usuarios

where usuario =@usuario and estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_BusquedaActividadforDescripcion] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Procedure [dbo].[usp\_BusquedaActividadforDescripcion]

@descripcion nvarchar(80)

As

Select \* from Actividades Where

descripcion = @descripcion

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_BusquedaActividadforFecha] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE Procedure [dbo].[usp\_BusquedaActividadforFecha]

@fec1 Date

As

Select \* from actividades Where

fecha\_creacion = @fec1

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_costos\_prod\_imprenta\_datos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_costos\_prod\_imprenta\_datos]

--cabecera

@codigo nchar(10)

AS

SELECT \*

FROM costos\_prod\_imprenta

WHERE codigo= @codigo

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_actividad\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_actividad\_codigo]

@codigo nchar(10)

AS

BEGIN

update actividades set estado ='N'

WHERE codigo = @codigo

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_actividad\_id] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_actividad\_id]

@id int

AS

BEGIN

update actividades set estado ='N'

WHERE idactividad = @id

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_alumno\_dni] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_alumno\_dni]

@DNI VARCHAR(10)

AS

BEGIN

UPDATE alumnos

set estado ='N'

WHERE DNI = @DNI

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_alumno\_id] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_alumno\_id]

@idalumno int

AS

BEGIN

UPDATE alumnos

set estado ='N'

WHERE idalumno = @idalumno

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_costosimprenta] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_costosimprenta]

@codigo nchar(10)

AS

BEGIN

update costos\_prod\_imprenta

set estado ='N'

WHERE (codigo = @codigo

and estado ='S')

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_equipo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_equipo]

@idEquipo int

AS

BEGIN

update Equipo

set estado ='N'

WHERE (idEquipo = @idEquipo

and estado ='S')

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_sala] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_sala]

@idsala int

AS

BEGIN

update sala

set estado ='N'

WHERE idsala = @idsala

and estado ='S'

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_eliminar\_usuario] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_eliminar\_usuario]

@idusuario VARCHAR(10)

AS

BEGIN

update usuarios

set estado ='N'

WHERE idusuario = @idusuario

and estado ='S'

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_equipo\_datos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_equipo\_datos]

@idEquipo int

AS

SELECT \*

FROM Equipo

WHERE idEquipo= @idEquipo

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_equipo\_datos\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_equipo\_datos\_codigo]

@Nombre nchar(20)

AS

SELECT \*

FROM Equipo

WHERE Nombre= @Nombre

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_equipo\_listar\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_equipo\_listar\_all]

AS

BEGIN

select \* from Equipo

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_generarcodigo\_actividad] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_generarcodigo\_actividad]

@codigox NCHAR(10) output

AS

SELECT @codigox = max(codigo) FROM actividades

return @codigox + 1

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_generarcodigo\_organizadores] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_generarcodigo\_organizadores]

@codigox NCHAR(10) output

AS

SELECT @codigox = max(codigo) FROM organizadores

return @codigox + 1

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_lista\_equipos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_lista\_equipos]

@codigo nchar(8)

AS

--QUERY QUE MUESTRA LISTA DE EQUIPOS

select l.idlistaeq, e.idEquipo,l.codigo, e.Nombre,e.Procesador,e.RAM,e.SO,e.TarjetaMadre from listaequipos l , detalle\_list\_equipos d , Equipo e

where l.codigo = d.codigo

and d.idEquipo = e.idEquipo

and l.estado ='S'

and d.estado ='S'

and e.estado ='S'

and l.codigo=@codigo

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_lista\_equipos\_oficial] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_lista\_equipos\_oficial]

@codigo nchar(8)

AS

--QUERY QUE MUESTRA LISTA DE EQUIPOS

select l.idlistaeq, e.idEquipo,l.codigo, e.Nombre,e.Procesador,e.RAM,e.SO,e.TarjetaMadre from listaequipos l , detalle\_list\_equipos d , Equipo e

where l.codigo = d.codigo

and d.idEquipo = e.idEquipo

and l.estado ='S'

and d.estado ='S'

and e.estado ='S'

and l.codigo=@codigo

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listaequipo\_datos\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listaequipo\_datos\_codigo]

--cabecera

@codigo nchar(10)

AS

SELECT \*

FROM listaequipos

WHERE codigo= @codigo

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listaequipos\_listar\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listaequipos\_listar\_all]

--cabecera

@codigo nchar(5)

AS

SELECT \*

FROM listaequipos

--WHERE codigo= @codigo

where estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_actividades\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Raúl Armas

-- Create date: 03/11/19

-- Description: Lista todas las actividades

-- =============================================

CREATE PROCEDURE [dbo].[usp\_listar\_actividades\_all]

AS

BEGIN

select \* from actividades

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_alumnos\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Raúl Armas

-- Create date: 03/11/19

-- Description: Lista todos los participantes

-- =============================================

CREATE PROCEDURE [dbo].[usp\_listar\_alumnos\_all]

AS

BEGIN

select \* from alumnos

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_alumnos\_tipo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Raúl Armas

-- Create date: 03/11/19

-- Description: Lista todos los participantes estudiantes

-- =============================================

CREATE PROCEDURE [dbo].[usp\_listar\_alumnos\_tipo]

@tipo\_alumno nchar(10)

AS

select \* from alumnos where tipo\_alumno = @tipo\_alumno

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_ambientes\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listar\_ambientes\_all]

AS

BEGIN

select \* from ambiente

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_curso\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listar\_curso\_all]

@tipo nchar(2)

AS

BEGIN

IF @tipo ='0'

select \* from curso

where estado ='S'

ELSE

select \* from curso

where estado ='S'

and tipo = @tipo

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_curso\_tipo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listar\_curso\_tipo]

@tipo nchar(2)

AS

BEGIN

select \* from curso

where estado ='S'

and tipo = @tipo

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_salas\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listar\_salas\_all]

AS

BEGIN

select \* from sala

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_sede\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listar\_sede\_all]

AS

BEGIN

select \* from sede

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_listar\_tipoeventos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_listar\_tipoeventos]

AS

BEGIN

select \* from parametros

where estado ='S'

and nombre = 'TIPOEVENTO'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_organizadores\_datos\_codigo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_organizadores\_datos\_codigo]

--cabecera

@codigo nchar(5)

AS

SELECT \*

FROM organizadores

WHERE codigo= @codigo

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_organizadores\_oficial] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_organizadores\_oficial]

@codigo nchar(8)

AS

--QUERY QUE MUESTRA LISTA DE ORGANIZADORES

select p.DNI,p.ap\_paterno, p.ap\_materno, a.codigo,a.idorganiz from organizadores a , detalle\_organizadores d, alumnos p

where a.codigo = d.codigo

and d.DNI = p.DNI

and d.estado ='S'

and a.estado ='S'

and p.estado ='S'

and a.codigo=@codigo

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_organizadores\_oficial\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_organizadores\_oficial\_all]

--cabecera

@codigo nchar(5)

AS

SELECT \*

FROM organizadores

WHERE codigo= @codigo

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_parametros\_valor\_xnombre] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_parametros\_valor\_xnombre]

--cabecera

@nombre nchar(20)

AS

SELECT \*

FROM parametros

WHERE nombre= @nombre

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_participantes\_datos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_participantes\_datos]

@DNI nchar(8)

AS

SELECT \*

FROM participantes

WHERE DNI= @DNI

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_actividad\_oficial] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_actividad\_oficial]

@nombre nchar(30),

@codigo nchar(10),

@descripcion NVARCHAR(100),

@responsable nchar(50),

@rol\_creacion NCHAR(10),

@fec\_inicio datetime,

@fec\_fin datetime,

@tipoevento nchar(30),

@lugar nvarchar(200),

@certificado nchar(2), --puede ser tres tipos

@inscripcion nchar(2), -- puede ser tres tipos

@idlistaeq integer,

@idsala integer,

@idsede integer,

@idambiente integer,

@justificacion nvarchar(100),

@objetivos nvarchar(100),

@horas integer,

@idorganiz integer

AS

INSERT INTO actividades(nombre,codigo,descripcion,fec\_inicio,fec\_fin,responsable,rol\_creacion,idorganiz,idlistaeq,idsala,objetivos,justificacion,horas,tipoevento,certificado,inscripcion,idsede,idambiente,lugar)

VALUES (@nombre,@codigo,@descripcion,@fec\_inicio,@fec\_Fin,@responsable,@rol\_creacion,@idorganiz,@idlistaeq,@idsala,@objetivos,@justificacion,@horas,@tipoevento,@certificado,@inscripcion,@idsede,@idambiente,@lugar)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_alumno] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_alumno]

-- @IdDocente int ,

@ap\_paterno VARCHAR(30),

@ap\_materno VARCHAR(30),

@nombre VARCHAR(30),

@telefono NCHAR(20),

@sexo NCHAR(1),

@correo NVARCHAR(80),

@DNI NCHAR(8),

@carrera NVARCHAR(80),

@direccion NVARCHAR(80),

@tipo\_alumno NCHAR(10)

AS

INSERT INTO alumnos(ap\_paterno,ap\_materno,nombre,telefono,sexo,correo,DNI,carrera,direccion,tipo\_alumno ) VALUES

(@ap\_paterno,@ap\_materno,@nombre,@telefono,@sexo,@correo,@DNI,@carrera,@direccion,@tipo\_alumno)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_costos\_prod\_imprenta] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_costos\_prod\_imprenta]

@codigo NCHAR(5),

@nro\_libros int,

@nro\_paginas int,

@largo decimal,

@ancho decimal,

@nro\_pliegos int,

@nro\_resmas int,

@nro\_placas int,

@doblez decimal,

@alce decimal,

@corte\_refilado decimal,

@encolado decimal,

@movilidad decimal,

@manejo\_archivo decimal,

@best\_color decimal,

@cif decimal,

@tinta decimal(10,2),

@total decimal,

@titulo NCHAR(50)

AS

INSERT INTO costos\_prod\_imprenta(codigo,nro\_libros,nro\_paginas,largo,ancho,nro\_pliegos,nro\_resmas,nro\_placas,doblez,alce,corte\_refilado,encolado

,movilidad,manejo\_archivo,best\_color,cif,tinta,total,titulo)

VALUES (@codigo,@nro\_libros,@nro\_paginas,@largo,@ancho,@nro\_pliegos,@nro\_resmas,@nro\_placas,@doblez,@alce,@corte\_refilado,@encolado

,@movilidad,@manejo\_archivo,@best\_color,@cif,@tinta,@total,@titulo)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_curso] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_curso]

@Nombre NVARCHAR(80),

--@SO NVARCHAR(80),

-- @Procesador NVARCHAR(30),

@tipo nchar(2)

AS

INSERT INTO curso(nombreCurso,tipo,IdTarifa) VALUES (@Nombre,@tipo,1)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_det\_list\_equipos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_det\_list\_equipos]

@idEquipo int,

@codigo NCHAR(5)

AS

INSERT INTO detalle\_list\_equipos(idEquipo,codigo)

VALUES (@idEquipo,@codigo)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_det\_org] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_det\_org]

@DNI NCHAR(8),

@codigo NCHAR(5)

AS

INSERT INTO detalle\_organizadores(DNI,codigo)

VALUES (@DNI,@codigo)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_equipo] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_equipo]

@Nombre NVARCHAR(80),

@SO NVARCHAR(80),

@Procesador NVARCHAR(30),

@RAM NVARCHAR(10),

@TMADRE NVARCHAR(10)

AS

INSERT INTO Equipo(Nombre,SO,Procesador,RAM,TarjetaMadre) VALUES (@Nombre,@SO,@Procesador,@RAM,@TMADRE)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_lista\_equipos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_lista\_equipos]

@codigo NCHAR(5)

AS

INSERT INTO listaequipos(codigo)

VALUES (@codigo)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_organizadores] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_organizadores]

@codigo NCHAR(5),

@iddetorg int,

@idorganiz int output

AS

INSERT INTO organizadores(codigo,fec\_registro,iddetorg)

VALUES (@codigo,GETDATE(),@iddetorg)

set @idorganiz=@@IDENTITY

return @idorganiz

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_sala] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_sala]

-- @IdDocente int ,

@nombre NVARCHAR(80),

@ubicacion NVARCHAR(80),

@capacidad INTEGER,

@rol\_creacion NCHAR(10),

@tipo\_Sala char(10)

AS

insert into sala(nombre,ubicacion,capacidad,rol\_creacion,tipo\_sala)

values ( @nombre,@ubicacion,@capacidad , @rol\_creacion,@tipo\_Sala )

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_sede] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_sede]

@nombre NVARCHAR(20),

@direccion NVARCHAR(100)

as

insert into sede(nombre,direccion)

values ( @nombre,@direccion)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_registrar\_usuario] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_registrar\_usuario]

@usuario nchar(10),

@clave nchar(10)

AS

INSERT INTO usuarios(usuario,clave) values (@usuario,@clave)

--actividades(descripcion,fecha) VALUES (@descripcion,@fecha)

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_sala\_datos] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[usp\_sala\_datos]

@nombre nvarchar(80)

AS

SELECT \*

FROM sala

WHERE nombre= @nombre

and estado ='S'

RETURN

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_usuarios\_listar\_all] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Raúl Armas

-- Create date: 03/11/19

-- Description: Lista todas las actividades

-- =============================================

CREATE PROCEDURE [dbo].[usp\_usuarios\_listar\_all]

AS

BEGIN

select \* from usuarios

where estado ='S'

END

return 0

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_ValidaUsuario] Script Date: 25/01/2022 20:06:30 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- En el Stored Procedure

CREATE PROCEDURE [dbo].[usp\_ValidaUsuario]

@Usuario nvarchar(10),

@Clave nvarchar(10)

AS

begin

If (SELECT count(1) From usuarios WHERE usuario=@Usuario and clave=@Clave and estado ='S') > 0

select \* from usuarios where usuario=@Usuario and estado ='S'

end

GO

USE [master]

GO

ALTER DATABASE [bdgeneric] SET READ\_WRITE

GO

# Proyecto en Visual Studio C#

**Login**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Formulario Inicio**

Pantalla de computadora encendida

Descripción generada automáticamente con confianza media

**Actualización de datos de usuario**

Captura de pantalla de computadora

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

**Mantenedor de actividades de extensión universitaria**

Gráfico, Gráfico de barras

Descripción generada automáticamente

**Reporte de actividad en PDF**

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

1. [↑](#footnote-ref-1)
2. Cabe resaltar que para la creación de las tablas se pasó por el proceso de normalización. [↑](#footnote-ref-2)