

Universidad Nacional de Costa Rica
Sede Regional Chorotega Campus Nicoya

EIF 211

Diseño e Implementación de Bases de Datos

Proyecto

Mini Test Link

Profesor

Francisco Coulon

Integrantes:

Nombre	Cédula
María Carolina González Jiménez	504550553
Ivan Aguero Acosta	504560302
Isaac Villegas Miranda	504560937

I Ciclo 2025

Tabla de contenido

Introducción.....	3
Planteamiento del problema:.....	4
Descripción de equipo:.....	4
Justificación.....	5
Objetivo general.....	5
Objetivo específico.....	5
Descripción de equipo.....	6
Resultados y su discusión.....	6
Modelo entidad relación.....	6
Modelo relacional.....	7
Estrategia de auditoría a utilizar.....	7
Tecnologías utilizadas.....	7
Conclusiones:.....	8
Recomendaciones:.....	8
Referencias.....	8
Anexos.....	8
Scripts Generados.....	8
Consideraciones para una versión final.....	14
Bitácora.....	14

Introducción

Este proyecto tiene como objetivo crear un sistema llamado “Mini test link”, el cual está pensado para ayudar al equipo de desarrollo a organizar mejor las pruebas que se hacen a las tareas antes de entregarlas. Actualmente, no tienen una herramienta que les ayude a manejar los casos de prueba y eso ha causado atrasados en las revisiones y en las entregas del trabajo.

La idea del sistema es que permita crear grupos de pruebas llamados “test suites”, dentro de los cuales se pueden agregar diferentes casos de prueba, cada uno con sus pasos, requisitos y resultados esperados. Los usuarios van a poder ejecutar estas pruebas, repetirlas si es necesario, guardar los resultados y ver un historial por si se necesita volver a una versión anterior.

Para este proyecto aplicamos lo aprendido en clase, como el modelo entidad-relación, el modelo relacional y el uso de SQL Server para crear las tablas, procedimientos, vistas y consultas que necesita el sistema. Este trabajo es una muestra de cómo se pueden aplicar los conocimientos de base de datos en una situación real para ayudar a mejorar el trabajo en equipo y asegurar la calidad del software.

Planteamiento del problema:

Debido a problemas de administración dentro del equipo de trabajo del que somos parte, las entregas de aseguramiento de calidad empiezan a tener problemas al punto que se retrasan las fechas de revisión y de entrega.

La administración dentro de un equipo de trabajo es una parte importante y la ausencia de este afecta el equipo. El director nos ha pedido que desarrollemos una herramienta que ayude a administrar las fases de aseguramiento de calidad, con tal de dar fin a este problema.

Después de una reunión y una lluvia de ideas, se decidió que el problema a solucionar es el siguiente:

- Los trabajos van a ser guardados por categorías.
- Los usuarios del sistema pueden hacer consultas de dichos estados por medio de una base de datos.
- Se tiene que preparar el sistema para seguir los pasos de la aceptación y los valores que se piden.

Descripción de equipo:

El equipo forma parte de una empresa de desarrollo de sistemas computacionales, en el área de aseguramiento de calidad.

Sin embargo esta área ha sido afectada por la alza de cantidad de trabajos que ha sufrido la empresa, haciendo que los trabajos se retrasen.

Para solucionar este problema se considera la creación de un sistema de administración de proyectos.

La idea es mejorar los siguientes puntos:

- Test suit.
- Gestión de tareas para los grupos de trabajo.
- Pasos para la elaboración de trabajo con sus requisitos, pre-requisitos y valores esperados.
- Ejecución de pruebas y su repetición.

Justificación

Es una gran oportunidad debido a que resolvemos un caso realista, además de demostrar los conocimientos adquiridos en bases de datos, modelado de información y desarrollo de sistemas orientados a la mejora de procesos internos. Asimismo, permite abordar problemáticas reales con clientes y empresas mediante soluciones tecnológicas concretas y escalables y dar mejoras a varios procesos que han sido complejos para la empresa tener una solución directa a este problema técnico, Por último con este proyecto se quiere motivar y fomentar el desarrollo de competencias técnicas y metodológicas en los estudiantes y colaboradores del proyectos similares.

Objetivo general

Desarrollar un sistema que se encargue de la administración de casos de prueba para que los procesos sean más eficientes y consistentes.

Objetivo específico

Diseñar el sistema para que este pueda hacer ejecutar pruebas que serán repetibles, dichas pruebas generarán datos.

Compartir las pruebas del sistema creado para tener una guía en el proceso de validación.

Establecer un sistema que permita consultas de datos.

Descripción de equipo

Somos el equipo de trabajo MiniLink, la cual nos encargaremos en un proyecto, o una caso de Mini Test Link, los integrantes del equipo técnico somos:

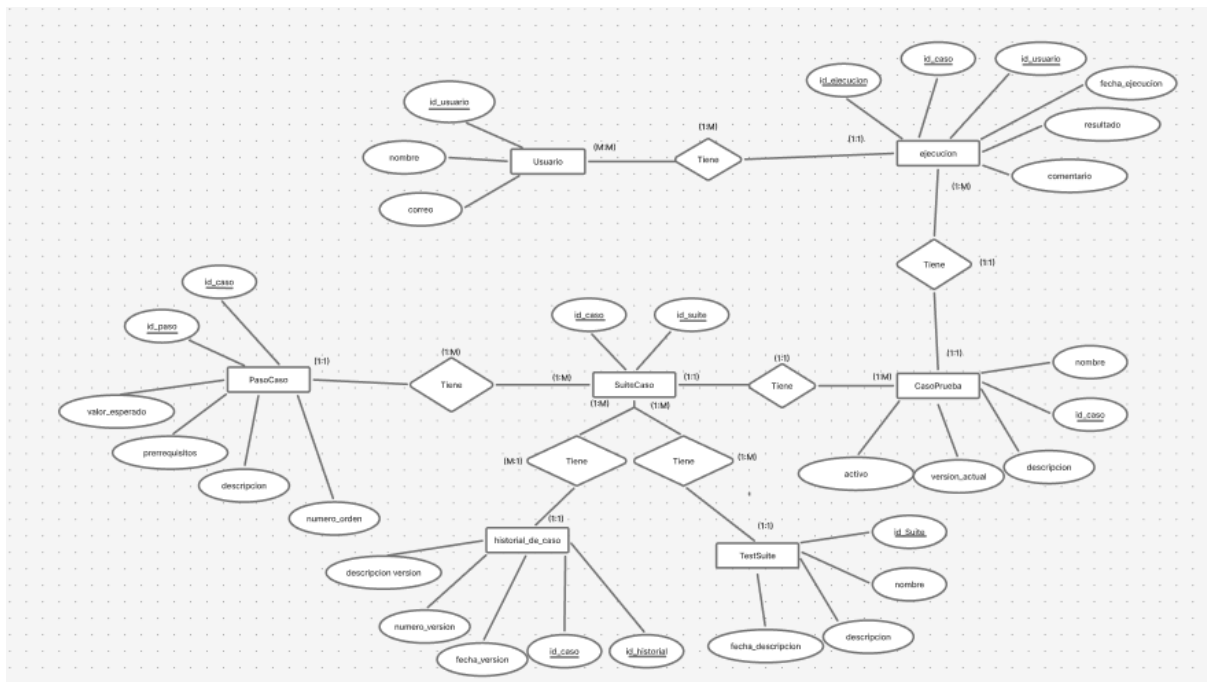
- María Carolina González Jiménez
- Ivan Agüero Acosta
- Isaac Villegas Miranda

Nos dedicamos a solucionar problemas en varias áreas en la informática, por ejemplo reparación de dispositivos electrónicos, creación de páginas web/API, creación de base de datos como nuestra especialidad, finalmente tenemos también somos excelentes en casos que requieran una resolución de lógica impecable.

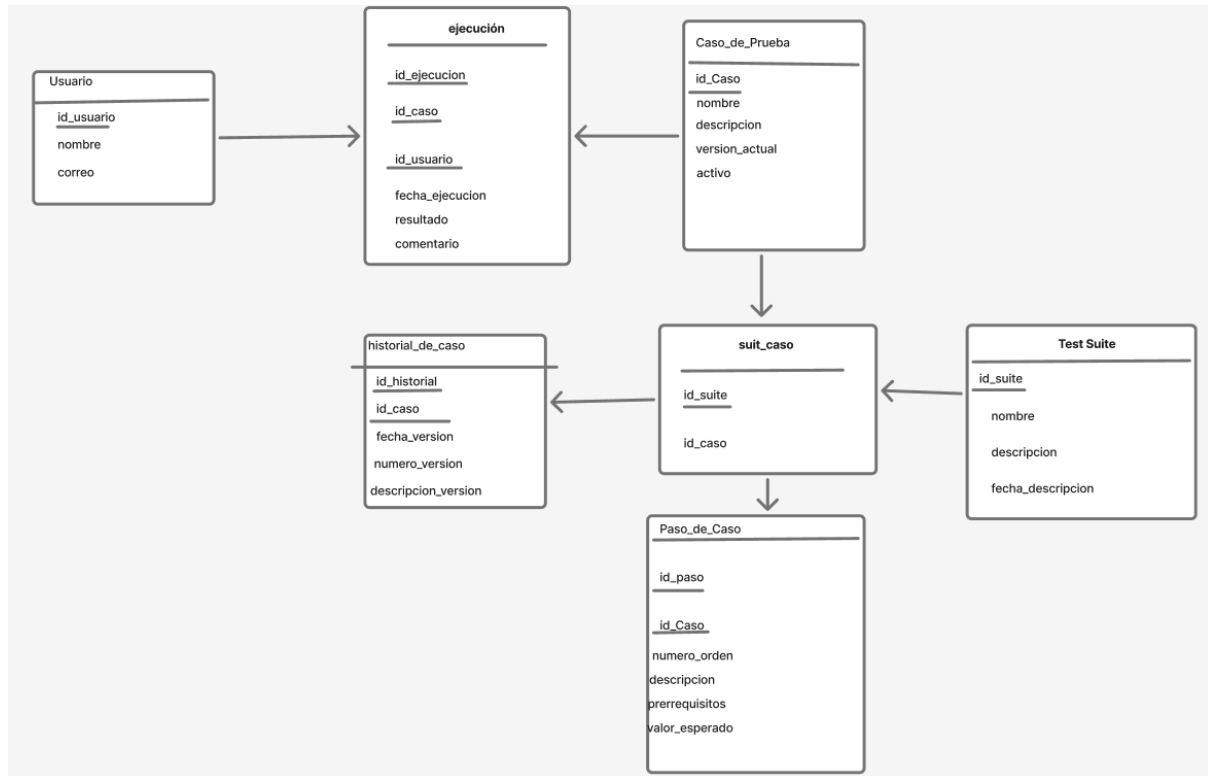
Se visualiza este proyecto como la mejor oportunidad que se nos ha dado para demostrar lo capaces que somos como equipo. Si el resultado del proyecto es un éxito podremos dar nuestro primer paso a convertir nuestro equipo de trabajo en un empresa formalizada a crear proyectos aún más desafiantes y complejas.

Resultados y su discusión

Modelo entidad relación



Modelo relacional



Estrategia de auditoría a utilizar

Para la realización de este trabajo, cada cierto tiempo hacíamos reuniones, en donde discutimos las ideas para crear los distintos instrumentos necesarios y lo dividimos entre los miembros del equipo. A la siguiente reunión, presentamos los adelantos de todos y después de ajustes, vemos cuáles son los siguientes pasos, para dividirlos nuevamente y continuar con el proyecto.

Tecnologías utilizadas

- SQL
- Visual Studio Code
- Figma
- GitHub
- Meet

Conclusiones:

Después de que nuestro equipo desarrollador terminará con la elaboración del sistema mini testlink, consideramos que los problemas que la empresa a la hora de ordenar los proyectos podría tener una baja, ya que con la integración de un sistema de seguimiento y registro de proyectos, la empresa tiene una forma de trabajar más limpia y segura.

Con este sistema funcionando, la empresa tiene acceso a la información de los proyectos, casos, suits, historial de proyectos y además a un registro que guarda todos los pasos que se debe seguir para realizar los casos adecuadamente.

La realización de este trabajo fue para el equipo uno de los más enriquecedores, debido a la poca información y formación que teníamos acerca de tecnologías de bases de datos antes de las respectivas clases. Tuvimos que pasar por una curva de aprendizaje grande, pero como equipo estamos felices con el resultado que se llegó.

Recomendaciones:

1. Se recomienda al usuario primeramente, insertar en la base de datos que se vaya a utilizar, todas las tablas, SPs y Triggers que fueron planteados, para el buen funcionamiento del sistema.
2. Además se recomienda respetar los tipos de datos que se esperan, escritos anteriormente en los parámetros de las tablas.
3. Finalmente se recomienda cambiar el nombre de la base de datos en el backup si este es distinto a el nombre de la base de datos que se está utilizando

Referencias

Francisco Coulon Ollivier (comunicación personal 2025).

Anexos

Scripts Generados

Creación de tabla "CasoPrueba"

```
CREATE TABLE CasoPrueba (  
    id_caso INT NOT NULL PRIMARY KEY IDENTITY(1,1),
```



```
nombre NVARCHAR(100),  
descripcion NVARCHAR(255),  
version_actual INT,  
activo BIT DEFAULT 1  
);
```

Creación de tabla "Ejecucion"

```
CREATE TABLE Ejecucion (  
    id_ejecucion INT PRIMARY KEY IDENTITY(1,1),  
    id_caso INT FOREIGN KEY REFERENCES CasoPrueba(id_caso),  
    id_usuario INT FOREIGN KEY REFERENCES Usuario(id_usuario),  
    fecha_ejecucion DATETIME DEFAULT GETDATE(),  
    resultado NVARCHAR(50),  
    comentarios NVARCHAR(255)  
);
```

Creación de tabla "historialCaso"

```
CREATE TABLE historialCaso (  
    id_historial INT PRIMARY KEY IDENTITY(1,1),  
    id_caso INT FOREIGN KEY REFERENCES CasoPrueba(id_caso),  
    fecha_version DATETIME DEFAULT GETDATE(),  
    descripcion_version NVARCHAR(255),  
    numero_version INT  
);  
go
```

Creación de tabla "PasoCaso"

```
CREATE TABLE PasoCaso (  
    id_paso INT PRIMARY KEY IDENTITY(1,1),  
    id_caso INT FOREIGN KEY REFERENCES CasoPrueba(id_caso),  
    numero_orden INT,  
    descripcion NVARCHAR(255),  
    prerequisitos NVARCHAR(255),  
    valor_esperado NVARCHAR(255),  
);
```

Creación de tabla "SuiteCaso"

```
CREATE TABLE SuiteCaso (  
    id_suite INT FOREIGN KEY REFERENCES TestSuite(id_suite),  
    id_caso INT FOREIGN KEY REFERENCES CasoPrueba(id_caso),
```

```
PRIMARY KEY (id_suite, id_caso)
);
```

Creación de tabla "TestSuite"

```
CREATE TABLE TestSuite (
    id_suite INT PRIMARY KEY IDENTITY (1,1),
    nombre NVARCHAR(100),
    descripcion NVARCHAR(255),
    fecha_creacion DATETIME DEFAULT GETDATE()
);
```

Creación de tabla "usuario"

```
CREATE TABLE usuario (
    id_usuario INT PRIMARY KEY IDENTITY(1,1),
    nombre NVARCHAR(50),
    primer_apellido NVARCHAR(50),
    segundo_apellido NVARCHAR(50),
    correo NVARCHAR(50)
);
```

Creación de SP para agregar usuario:

```
CREATE PROCEDURE sp_crear_usuario (@nombre NVARCHAR(50),
@primer_apellido NVARCHAR(50), @segundo_apellido NVARCHAR(50), @correo
NVARCHAR(50))
AS
BEGIN
INSERT INTO usuario (nombre, primer_apellido, segundo_apellido, correo)
VALUES (@nombre, @primer_apellido, @segundo_apellido, @correo);
Select SCOPE_IDENTITY() AS NuevoUsuarioId;
END;

-----Para utilizar SP-----
EXEC sp_crear_usuario
    @nombre = '',
    @primer_apellido = '',
    @segundo_apellido = '',
    @correo = '';
```

Creación de SP para agregar Caso de prueba:

```
CREATE PROCEDURE sp_crear_casoPrueba (@nombre NVARCHAR(100),
@descripcion NVARCHAR(255), @version_actual INT, @activo BIT)
AS
BEGIN
INSERT INTO CasoPrueba (nombre, descripcion, version_actual, activo)
VALUES (@nombre, @descripcion, @version_actual, @activo);
Select SCOPE_IDENTITY() AS NuevoCasodPruebaId;
END;

-----Para utilizar SP-----
EXEC sp_crear_casoPrueba
    @nombre = '',
    @descripcion = '',
    @version_actual = 7,
    @activo = 0;
```

Creación de SP para actualizar caso de prueba:

```
CREATE PROCEDURE sp_actualizar_prueba
    @id_caso INT,
    @nombre NVARCHAR(100),
    @descripcion NVARCHAR(255),
    @activo BIT
AS
BEGIN
    IF EXISTS (SELECT 1 FROM CasoPrueba WHERE id_caso = @id_caso)
    BEGIN
        UPDATE CasoPrueba
        SET
            nombre = @nombre,
            descripcion = @descripcion,
            version_actual = version_actual + 1,
            activo = @activo
        WHERE id_caso = @id_caso;

        SELECT 'Caso actualizado correctamente' AS mensaje;
    END
    ELSE
    BEGIN
        SELECT 'El caso no existe' AS mensaje;
    END
END
```

```
END  
END;  
GO  
  
-----Para utilizar SP-----  
EXEC sp_actualizar_caso  
    @id_caso = 1,  
    @nombre = '',  
    @descripcion = '';
```

Creación de SP para agregar colección:

```
CREATE PROCEDURE sp_CrearTestSuite (@nombre NVARCHAR(100), @descripcion  
NVARCHAR(255))  
AS  
BEGIN  
INSERT INTO TestSuite (nombre,descripcion)  
VALUES (@nombre, @descripcion);  
END;  
  
-----Para utilizar SP-----  
EXEC sp_CrearTestSuite  
    @nombre = '',  
    @descripcion = '';
```

Creación de SP para actualizar coleccion:

```
CREATE PROCEDURE sp_ActualizarTestSuite (@id_suite INT, @nuevo_nombre  
NVARCHAR(100), @nueva_descripcion NVARCHAR(255))  
AS  
BEGIN  
UPDATE TestSuite  
SET nombre = @nuevo_nombre,  
descripcion = @nueva_descripcion  
WHERE id_suite = @id_suite;  
END;  
  
-----Para utilizar SP  
EXEC sp_ActualizarTestSuite  
    @id_suite = 1,  
    @nuevo_nombre = '',  
    @nueva_descripcion = '';
```

Trigger para historial de caso:

```
CREATE TRIGGER tr_crear_historial
ON casoPrueba
AFTER INSERT, UPDATE
AS
BEGIN
    INSERT INTO HistorialCaso (id_caso, descripcion_version,
numero_version)
    SELECT
        i.id_caso,
        'Nueva versión del caso: ' + CAST(i.id_caso AS NVARCHAR(10)),
        ISNULL(
            (SELECT MAX(h.numero_version) FROM HistorialCaso h WHERE
h.id_caso = i.id_caso),
            0
        ) + 1
    FROM inserted i;
END;
GO
```

Trigger para backup de base de datos:

```
CREATE TRIGGER BackupOnChange
ON DATABASE
FOR CREATE_TABLE, ALTER_TABLE, DROP_TABLE
AS
BEGIN
    EXEC('BACKUP DATABASE mini_testlink
        TO DISK = 'C:\Backups\mini_testlink.bak'
        WITH FORMAT,
        NAME = 'Backup completo de mini_testlink';')
END;
```

Vista de colección

```
CREATE VIEW vista_suit_caso
AS
SELECT *
FROM SuiteCaso;
GO

-----Para activar en cosola-----
SELECT * FROM vista_suit_caso;
```

Vista de últimos dos resultados:

```
CREATE VIEW vista_ultimas_2_ejecuciones  
AS  
SELECT TOP 2 *  
FROM Ejecucion  
ORDER BY id_ejecucion DESC;  
GO
```

```
-----Para activar en cosola-----  
SELECT * FROM vista_usuarios_activos;
```

Backup de datos:

```
BACKUP DATABASE mini_testlink  
TO DISK = 'C:\Backups\mini_testlink.bak'  
WITH FORMAT,  
    NAME = 'Backup completo de mini_testlink';  
  
-----Para restaurar-----  
RESTORE DATABASE mini_testlink  
FROM DISK = 'C:\Backups\mini_testlink.bak';
```

Consideraciones para una versión final

Este proyecto se pensó que puede llegar a mejorarse y ser escalable para la empresa, la cual se consideraría algunas cosas para una versión final de este proyecto. Por ejemplo utilizar esta base de datos para almacenar todas las pruebas, y ejecuciones teniendo una conexión a una API creada para tener un registro de todas las pruebas que realizará el personal de la empresa. También que esté conectada a la red, para que en caso de que no funcione el sistema localmente, poder realizar pruebas de una manera remota y sencilla para que todo funcione en perfectas condiciones. Por último darle mantenimiento por un tiempo para verificar que el sistema sea utilizado correctamente, y cumpla con las expectativas del grupo de trabajo y personal de la empresa.

Bitácora

Fecha	Contenido	Asistencia
1/5/2025	Nuestra primera reunión fue directa con el objetivo de dividir las tareas, las cuales	Isaac Villegas Miranda ✓ Ivan Agüero Acosta ✓ María Carolina González

	fueron asignadas de forma equitativa con todos los miembros del grupo. Esta reunión fue realmente muy veloz debido a que aún no teníamos el conocimiento suficiente para trabajar en scripts y la programación principal del proyecto Mini Test Link	Jiménez ✓
8/5/2025	En horario lectivo tuvimos una reunión y conversación relevante para decidir el nombre del equipo, cómo visualizamos el proyecto dando opiniones variadas sobre el caso asignado.	Isaac Villegas Miranda ✓ Ivan Agüero Acosta ✓ María Carolina González Jiménez ✓
22/5/2025	Tuvimos una reunión la cual fue concisa debido a que fue después de clase. La cual se designó las tareas de investigación y estudio, debido se necesita estar preparado para realizar el proyecto de base de datos de una manera eficaz.	Isaac Villegas Miranda ✓ Ivan Agüero Acosta ✓ María Carolina González Jiménez ✓
3/6/2025	Tuvimos una reunión, la cual discutimos sobre las tablas que necesitamos y todos los requisitos que se nos impusieron para crear la base de datos, desafiando nuestros conocimientos. La cual usamos una herramienta para que el poder visualizar las tablas sea más sencillo y poder plasmarlo en "SQLCMD". Creamos las tablas en un Figma donde todos aportando en cómo se	Isaac Villegas Miranda ✓ Ivan Agüero Acosta ✓ María Carolina González Jiménez ✓

	puede dividir el proyecto en tablas.	
8/6/2025	Una de las últimas reuniones que tuvimos fue el 8/6/2025 la cual no asistió Carolina debido a problemas con el internet de su zona, pero pudimos ponernos de acuerdo por medio de mensajes vía WhatsApp, con las últimas tareas para finalizar el proyecto y tener finalizado lo más pronto posible este proyecto.	Isaac Villegas Miranda ✓ Ivan Agüero Acosta ✓ María Carolina González Jiménez No pudo asistir por problemas con el Internet y nos pusimos de acuerdo vía mensajería WhatsApp