REPUBLICA BOLIVARIANA DE VENEZUELA

UNIVERSIDAD NACIONAL EXPERIMENTAL PARA LAS TELECOMUNICACION E INFORMATICA

VICERRECTORADO ACADEMICO

PROGRAMA NACIONAL DE FORMACION EN INGENIERIA EN INFORMATICA

Diseño de la Seguridad

Autor: Br. Wrallean Javier Brito Vivas

C.I: 24.774.283

Tutor: Prof.

Caracas, 18 septiembre de 2025

Definición de autenticación y autorización

1. **Autenticación**

   La autenticación es el proceso de verificar la identidad de un usuario antes de que pueda acceder a las funcionalidades del sistema. Este proceso garantiza que solo las personas legítimas puedan iniciar sesión.

   - **Método de Identificación**: El sistema requiere que los usuarios se identifiquen con un correo electrónico y una contraseña encriptada.
   - **Requisitos Funcionales Clave:**
     - **RF 001 - Registro y autenticación de usuarios:** Es el requisito fundamental que permite a los usuarios acceder a la aplicación.
     - **RF 005 - Sistema de validación de credenciales:** Asegura que la información de inicio de sesión sea correcta y refuerza la seguridad del acceso.

2. **Autorización y Roles**

   La autorización se refiere al control de acceso que determina qué acciones puede realizar un usuario una vez que ha sido autenticado. Se implementa mediante la asignación de roles, que son un conjunto de permisos predefinidos.

   - **RF 003 - Roles y permisos:** Este requisito es de alta prioridad y establece la base del control de acceso para todas las funcionalidades del sistema.

- **Roles Definidos:**
    - **Administrador del Sistema:** Tiene la máxima autoridad, con permisos para gestionar roles, usuarios y la estabilidad general del sistema.
    - **Líder del Proyecto:** Encargado de la creación, configuración y gestión de colaboradores dentro de proyectos específicos.
    - **Usuario Regular:** Puede registrarse, editar su perfil y participar en proyectos a los que sea invitado.
    - **Visitante:** Un usuario no registrado con acceso limitado, principalmente para ver contenido público.

3. **Gestión de Sesiones**

Una sesión es el estado de la conexión de un usuario con la aplicación. La gestión de sesiones es vital para la seguridad, ya que asegura que la interacción del usuario esté controlada y protegida contra accesos no autorizados.

- **RF 006 - Gestión de sesiones:** Un requisito de alta prioridad que garantiza la seguridad y el control de las sesiones.
- **Atributos de la Sesión:** De acuerdo con el diagrama de clases, una sesión se define por su ID, el ID del usuario, la fecha y hora de inicio y la fecha y hora de cierre.
- **Seguridad:** Las sesiones deben estar protegidas y el sistema debe ser capaz de manejar conflictos, como ediciones simultáneas, para garantizar la integridad de los datos.

Identificación de Vulnerabilidades Comunes

1. Cross-Site Scripting (XSS)

El XSS es una vulnerabilidad que ocurre cuando un atacante inyecta código malicioso en una página web. Este código se ejecuta en el navegador de otros usuarios, lo que puede robar sus datos, como credenciales de sesión.

- **Riesgo en el proyecto:** El sistema de comentarios, los chats grupales y los campos de entrada de texto son puntos potenciales de ataque. Si estos campos no se validan y sanean adecuadamente, un atacante podría insertar un script malicioso que robe la sesión de un Líder de Proyecto o Administrador.
- **Mitigación:**
  - **Validación de entradas:** Implementar un filtro que elimine o escape caracteres peligrosos (<, >, &, ", ').
  - **Codificación de salida:** Asegurarse de que cualquier dato proveniente de los usuarios se muestre como texto plano y no como código ejecutable.

2. Cross-Site Request Forgery (CSRF)

El CSRF es un ataque que obliga a un usuario autenticado a realizar acciones no deseadas en una aplicación web en la que ha iniciado sesión. El atacante engaña al usuario para que envíe una solicitud a

la aplicación, como cambiar su contraseña o realizar una acción de borrado, sin su conocimiento.

- **Riesgo en el proyecto:** Las acciones críticas como crear un proyecto, eliminar una tarea, cambiar la contraseña o modificar permisos de un colaborador son vulnerables. Un atacante podría, por ejemplo, enviar un correo electrónico con un enlace que, al hacer clic, cambie el rol de un usuario sin que se dé cuenta.
- **Mitigación:**
  - **Tokens CSRF:** Implementar tokens únicos y aleatorios que se envíen con cada solicitud de formulario. El servidor verifica el token para asegurarse de que la solicitud es legítima y proviene del usuario.
  - **Verificación de encabezados:** Comprobar el encabezado Referer o Origin en las solicitudes para confirmar que provienen del mismo dominio de la aplicación.

3. Inyección de Código (SQL, NoSQL, etc.)

Las inyecciones de código ocurren cuando un atacante introduce datos maliciosos en un campo de entrada para manipular las consultas de la base de datos o el sistema operativo subyacente. La inyección de SQL es la más común y permite a los atacantes acceder, modificar o eliminar datos de la base de datos.

- **Riesgo en el proyecto:** La aplicación utiliza una base de datos para la gestión de usuarios, proyectos y tareas. Campos de búsqueda, formularios de inicio de sesión o cualquier campo que interactúe directamente con la base de datos son vulnerables. Un atacante podría, por ejemplo, usar una inyección de SQL para eludir el inicio de sesión o acceder a la información de todos los usuarios.
- **Mitigación:**
    - **Consultas parametrizadas (Prepared Statements):** Esta es la defensa más efectiva. Las consultas se construyen de manera que los datos de entrada se envían por separado de la lógica de la consulta, evitando que se interpreten como comandos.
    - **Validación de entradas:** Limpiar y validar todos los datos de entrada del usuario para asegurar que no contengan caracteres especiales o comandos de base de datos.
    - **Principios de mínimo privilegio:** Limitar los permisos de la base de datos para que la aplicación no pueda realizar acciones no autorizadas, incluso si ocurre una inyección.

4. Gestión de Sesiones Insegura

Esta vulnerabilidad surge cuando el manejo de las sesiones de usuario no es robusto, lo que permite a un atacante robar, secuestrar o predecir los identificadores de sesión.

- **Riesgo en el proyecto:** Los documentos mencionan que la Gestión de Sesiones es un requisito de alta prioridad y que las sesiones deben estar protegidas. Un fallo en esta implementación podría permitir a un atacante obtener un identificador de sesión y usarlo para suplantar la identidad de un usuario, como un Líder de Proyecto o incluso un Administrador. Esto le daría acceso total a sus funcionalidades y datos sin necesidad de conocer su contraseña.
- **Mitigación:**
  - **IDs de sesión seguros:** Generar identificadores de sesión largos, aleatorios e impredecibles.
  - **Uso de HTTPS:** Cifrar la comunicación para evitar que los identificadores de sesión sean interceptados en el tráfico de red.
  - **Tiempo de vida limitado:** Establecer un tiempo de caducidad para las sesiones. Por ejemplo, cerrarlas automáticamente después de un período de inactividad.

- **Renovación de sesión:** Cambiar el identificador de sesión después de un inicio de sesión exitoso o al elevar los privilegios del usuario.