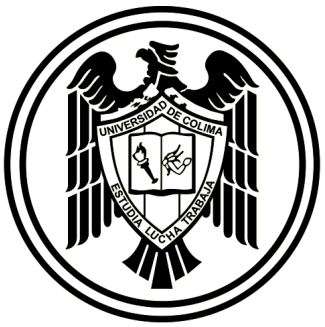

UNIVERSIDAD DE COLIMA

Facultad de Telemática

Ingeniería en Software

Actividad 19.- Proyecto MVP (Producto Mínimo Viable) de software

Desarrollo de Software Seguro

Integrantes del equipo:

Bañuelos Munguía Roberto Antonio

Rodríguez Reyes Miguel Jordán

Sanchez Medina Julian

6°E

21 de mayo de 2023.

Sistema de Gestión de Laboratorios de Enfermería

Descripción del proyecto

El objetivo del proyecto es brindar una solución tecnológica para la administración de los laboratorios de la facultad de enfermería, en el cual se hace uso de diversos materiales que se tienen que inventariar y llevar el control de su uso en las diversas prácticas de laboratorio con los alumnos, así mismo debe permitir crear informes de lo gastado en cada práctica y poder programar sesiones en las que los alumnos se puedan registrar.

En términos generales es un sistema de control de inventario donde existen entradas y salidas; Además se complementa como un gestor de prácticas de laboratorio.

Requerimientos

- La aplicación deberá cumplir con las normas y regulaciones aplicables al manejo de productos farmacéuticos y químicos en el país en el que se utilizará.
- Entre estas normas se encuentran las regulaciones sobre almacenamiento, manipulación, transporte y distribución de productos, así como la gestión de residuos peligrosos.
- El sistema deberá registrar cualquier actividad realizada en la aplicación, incluyendo el acceso de usuarios, la modificación de datos, la generación de informes y cualquier otra acción que se realice.
- Los registros deberán ser almacenados en una base de datos segura y accesible únicamente para personal autorizado.

Roles de usuario

Adecuandonos al principio de seguridad informática del mínimo privilegio tenemos para esta aplicación web contemplados 5 tipos de usuarios, cada uno con permisos diferentes los cuales se muestran en la figura 1.

MODULOS Y PERMISOS	Usuario Administrador	Usuario Gestor	Usuario Vendedor	Usuario Responsable	Usuario Alumno
Modulo Usuarios					
Puede crear, editar, eliminar, inactivar, activar usuarios administradores.	✓	✗	✗	✗	✗
Puede crear, editar, eliminar, inactivar, activar usuarios gestores.	✓	✓	✗	✗	✗
Puede crear, editar, eliminar, inactivar, activar usuarios vendedores.	✓	✓	✓	✗	✗
Puede crear, editar, eliminar, inactivar, activar usuarios responsables.	✓	✓	✓	✓	✗
Puede crear, editar, eliminar, inactivar, activar usuarios alumnos.	✓	✓	✓	✓	✗
Modulo Salidas					
Puede crear una nueva salida. Asignando un responsable que tendrá que validarla.	✓	✓	✓	✗	✗
Puede subir un archivo a una salida. En caso de que esté aprobada y sea quien la creó.	✓	✓	✓	✗	✗
Modulo Configuración					
Puede editar el encabezado de los PDF, la cantidad de semestres y las asistencias y faltas requeridas para cada semestre.	✓	✗	✗	✗	✗
Modulo Movimientos					
Puede ver y aprobar o rechazar las salidas en las que esté como responsable.	✓	✓	✓	✓	✗
Puede ver todas los movimientos por aprobar pero solo aprobar o rechazar las salidas en las que esté como responsable.	✓	✓	✗	✗	✗
Modulo Materiales					
Puede ver y editar los datos de los productos en el inventario.	✓	✓	✓	✗	✗
Puede editar el Stock actual y pasar productos a caducados.	✓	✗	✗	✗	✗
Modulo Prácticas					
Puede crear, editar, inactivar, eliminar plantillas.	✓	✓	✓	✓	✗
Puede crear, editar, inactivar, eliminar prácticas.	✓	✓	✓	✓	✗
Puede nombrar lista en prácticas que estén listas.	✓	✓	✓	✓	✗
Pueden ver las prácticas que están disponibles para tomarse.	✓	✓	✓	✓	✓
Pueden registrarse para asistir a las prácticas.	✗	✗	✗	✗	✓
Modulo Alumnos					
Puede crear, editar, eliminar, inactivar, activar, en general realizar acciones.	✓	✓	✗	✗	✗
Pueden ver el módulo.	✓	✓	✓	✓	✗
Modulo Salidas					
Puede crear una nueva entrada.	✓	✓	✓	✗	✗
Puede subir un archivo a una entrada. En caso de que esté aprobada y sea quien la creó.	✓	✓	✓	✗	✗
Modulo Reportes					
Tiene acceso al módulo.	✓	✗	✗	✗	✗

Figura 1: Tabla de permisos de usuarios.

Desarrollo del Proyecto

Matriz de Activos de Información

Podemos encontrar el inventario de activos aquí.

Arquitectura de Seguridad

Patrón de diseño

Una de las maneras en las que se puede mejorar la seguridad de un desarrollo WEB es utilizando algún patrón de diseño, en este caso implementaremos el patrón MVC (Modelo, Vista, Controlador), para tener dividido el código y tener las menores vulnerabilidades posibles, cabe mencionar que el desarrollo se realizará principalmente en PHP.

FrontEnd

Para la creación del frontend se utilizará el framework de CSS BOOTSTRAP, combinado con las bibliotecas de FPDF para visualizar reportes en PDF y ALERTIFY JS para mostrar alertas, todo esto conjunto con CSS nativo. Así mismo dentro del funcionamiento del frontend se incluirán validaciones para evitar la entrada de datos erróneos.

BackEnd

Para la creación del backend se utilizará PHP nativo y JavaScrip con la librería de JQuery. La API contará con validaciones para evitar la entrada de datos erróneos o maliciosos, asimismo cuidará solo mandar al frontend los datos requeridos para evitar fugas de información.

Funciones del sistema

Modulo Usuarios

En este módulo se podrán registrar nuevos usuarios, desactivarlos o activarlos, eliminarlos o editarlos. Cabe mencionar que según el rol que tengas serán los usuarios que se muestran, por ejemplo los administradores tienen poder sobre todos los usuarios, pero los gestores no pueden ver los administradores.

Todo esto con el fin de garantizar el buen funcionamiento del sistema y poder evitar que un usuario inferior le otorgue mayores privilegios a otro o le quite privilegios a usuarios superiores.

Módulo Salidas

En este módulo se podrán registrar todas las salidas que se realicen del inventario, para ello pondrán el código del producto y lo irán agregando al carrito para finalmente realizar la salida, cabe destacar que por petición del cliente se tiene que asignar una descripción de la salida y un responsable que valide la transacción, con el fin de evitar que se haga mal uso del sistema. Más adelante en el módulo de movimientos veremos como funciona la validación.

Así mismo en este módulo podremos ver el historial de transacciones que se han realizado y el estado en el que se encuentra, así como ver el PDF generado de la transacción. Cuando una transacción esté aprobada podrá subir un archivo que valide la salida.

Módulo Configuración

Este módulo es muy sencillo y solo está disponible para los administradores del sistema. Este nos permite modificar la configuración de los reportes como dirección, facultad, etc.

Módulo Movimientos

Este módulo complementa el de salida, aquí podremos ver las salidas generadas en las cuales el usuario está como responsable. Tendrá 3 funciones, aprobar la salida, rechazar la salida y ver el reporte de la salida.

Si el usuario aprueba la salida se descontarán los elementos del inventario. Cabe destacar que el administrador y gestor podrán ver todos los movimientos por aprobar pero solo podrán aprobar o rechazar los que le corresponda.

Módulo Materiales

En este módulo se registran las categorías, proveedores y nuevos materiales, también se pueden editar, inactivar o eliminar. Cabe mencionar que en caso de que seas administrador puedes editar las existencias.

Módulo Prácticas

En este módulo se podrán crear prácticas a las cuales los alumnos se puedan registrar, así mismo los usuarios podrán crear nuevas prácticas, registrar asistencia de los alumnos cuando se llegue la fecha y crear plantillas de prácticas.

Módulo Alumnos

Este apartado es igual al de usuarios, con la diferencias que todos (menos los mismos alumnos) pueden registrar nuevos alumnos, desactivarlos o activarlos, eliminarlos o editarlos.

Así mismo está la función extra por parte de los usuarios de reiniciar las prácticas y faltas que se hayan realizado. Esto está relacionado con el apartado de prácticas.

Módulo Entradas

Este módulo es similar al de salidas, con la única diferencia de que no se escoge un responsable, si no un proveedor, por lo que no tiene que ser validado por nadie. También aquí se podrán ver el historias de entradas y cargar las facturas.

Módulo Reportes

En este apartado, que es exclusivo del administrador y gestor, se podrán ver estadísticas de compras, ventas, costos por práctica, etc. Con el fin de no comprometer datos “sensibles” con todos los usuarios.

Otras Funciones

Finalmente tiene funciones típicas de todos los sistemas como login, cambiar contraseña, cambiar foto de perfil.

Base de datos

Para la base de datos se estará utilizando PhpmyAdmin.

Diagrama de base de datos

Se adjunta el diagrama de la base de datos, el cual está sujeto a posibles cambios.

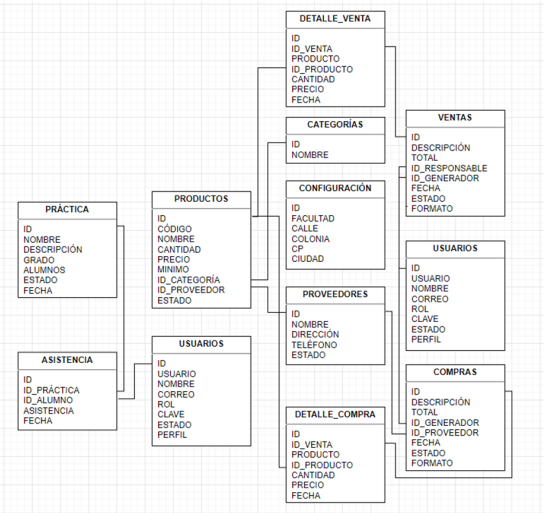

Figura 2: Base de datos tentativa.

Herramientas

GitHub

Para realizar el control de las versiones y el alojamiento del código fuente, tomando como medida de seguridad el uso del .gitignore para no subir archivos con datos importantes como posibles tokens o datos de conexión a la base de datos para evitar filtraciones de información y que en un futuro se tenga problemas con terceros queriendo tener acceso a la base de datos o la información de los usuarios.

VS Code

Para hacer el código ya que se utiliza como el editor de código el cual gracias a su versatilidad y sus extensiones para el desarrollo hace que sea más sencillo el poder desarrollar aplicaciones web como es nuestro caso brindándonos herramientas para la seguridad,un mejor performance al momento de realizar el código o facilidad de poder probar la aplicación web.

000WebHost

Para poder tener el sitio en la red necesitamos tener en donde poder tenerlo para eso usamos webhost el cual nos permite tener una aplicación web en la red de manera sencilla y a su vez brindando una gran cantidad de funciones para poder administrar de una mejor manera ésta aplicación web ya sea en el ámbito de seguridad,base de datos o incluso información acerca del tráfico del sitio.

XAMPP

Para realizar las pruebas del correcto funcionamiento de la API se utilizara postman en busca de encontrar posibles vulnerabilidades explotables para obtener información en caso de no estar autorizado y a su vez poder tener un servidor local.

WhatsApp

Para realizar la comunicación al momento de la creación del mvp para ponernos de acuerdo acerca de los cambios que se iban a realizar.

Políticas

Fallos

Se seguirá el principio de fallas de manera segura por lo que en caso de obtener errores se guardaran en logs para evitar que se muestran los fallos y prevenir que los usuarios externos intenten realizar algún tipo de ataque mediante la información que muestre el error.

Mecanismos de seguridad

Autenticación

Los usuarios se tendrán que autenticar mediante su usuario y contraseña y validar en un capcha, además solo podrá haber una sesión abierta del mismo usuario.

Autorización

Al momento de acceder el sistema solamente les mostrará las funcionalidades que el usuario pueda usar según su tipo rol en el sistema, y en caso de que intentara acceder mediante una URL esta le mostrará un mensaje diciendo que no tiene autorización para esa página.

Validación

Todos los datos que el usuario pueda ingresar al sistema serán validados para evitar que tengan errores o código malicioso.

Árbol de Ataque

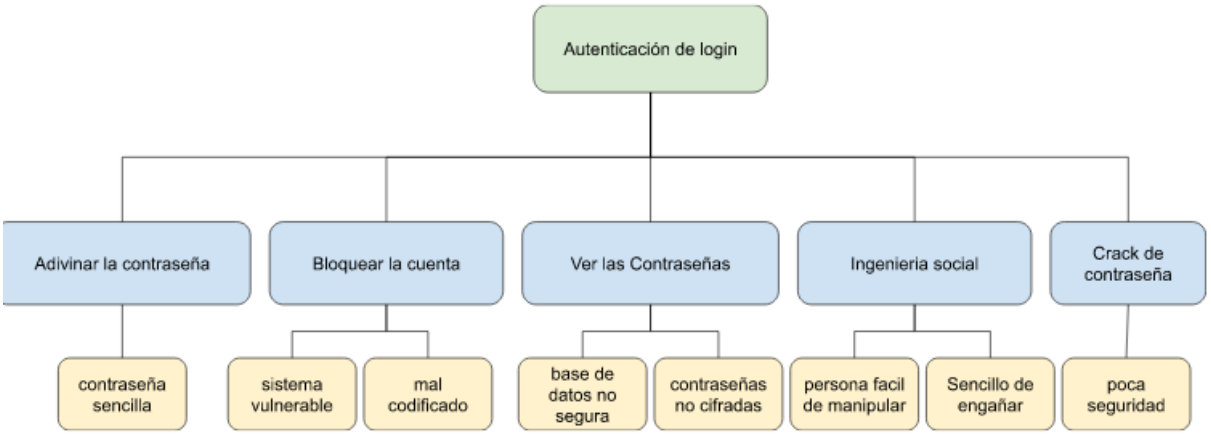

Figura 3: árbol de ataque de autenticación de login.

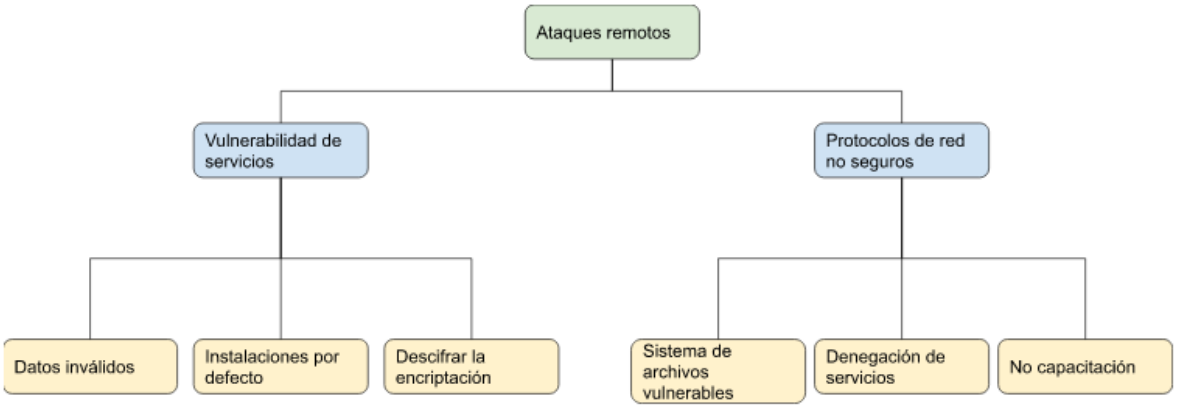

Figura 4: árbol de ataque de Ataques remotos.

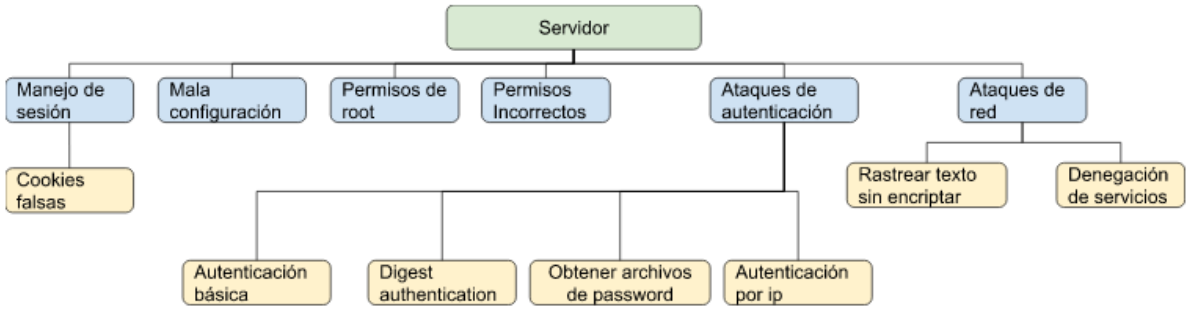

Figura 5: árbol de ataque de Servidor.

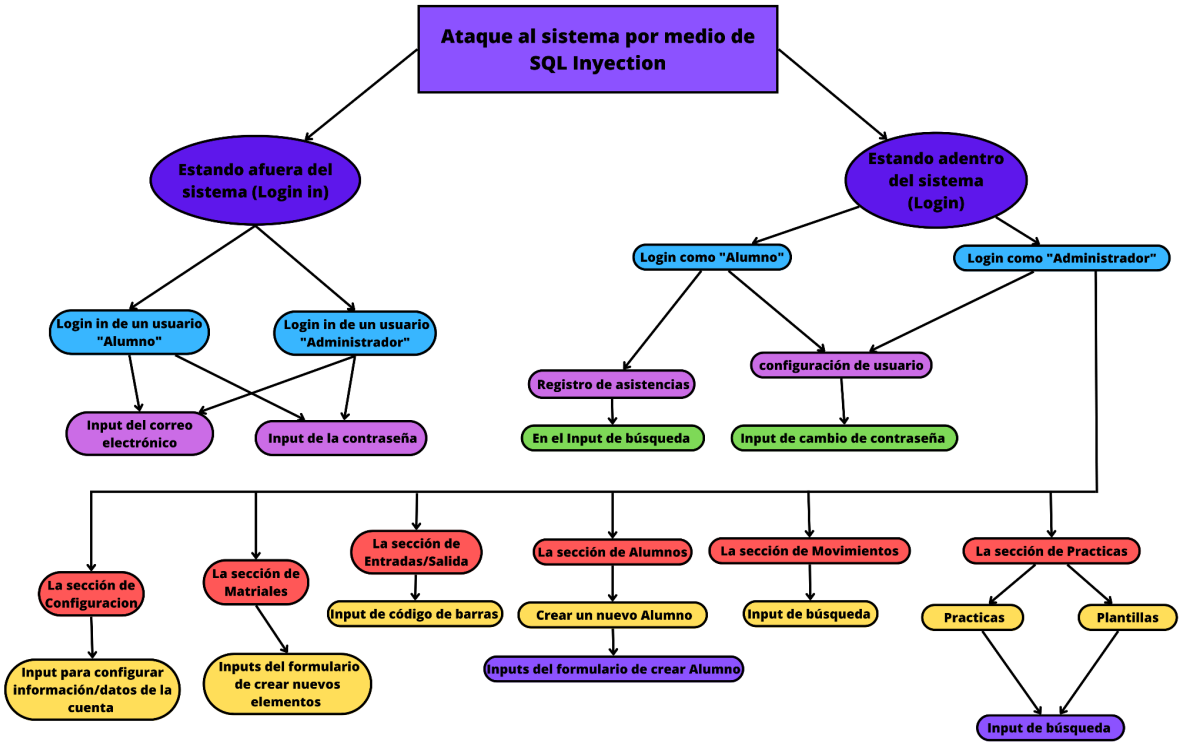

Figura 6: árbol de ataque de SQL Inyection.

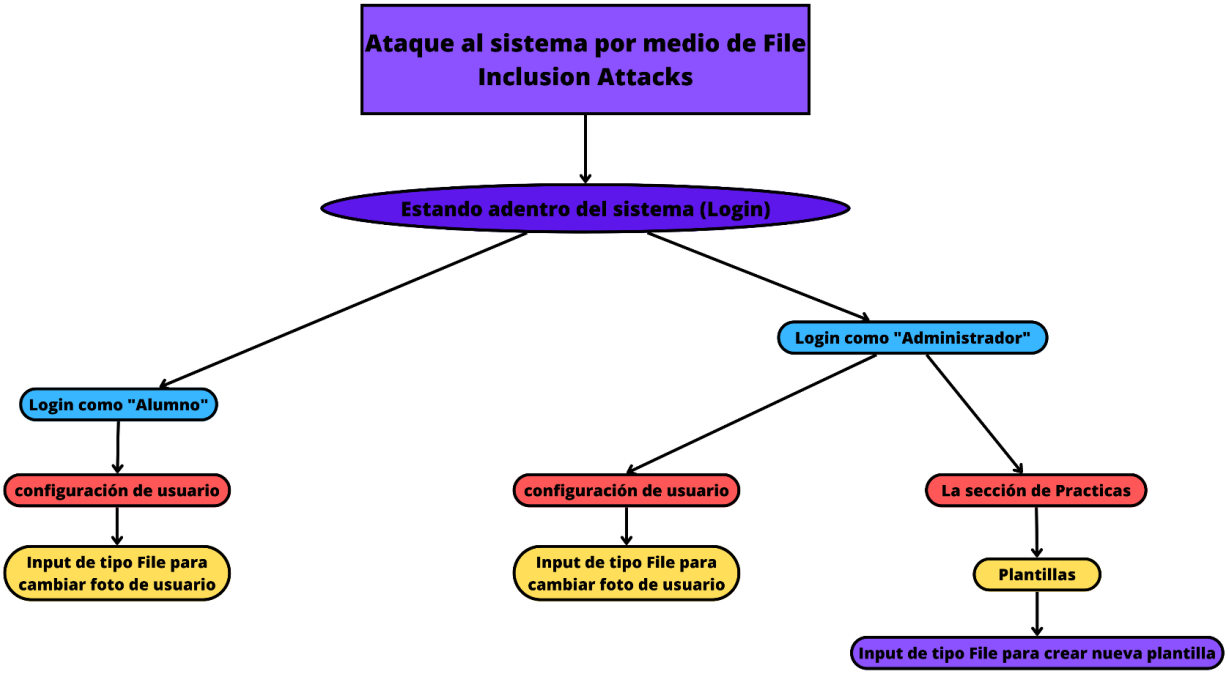

Figura 7: árbol de ataque de File Inclusión Attacks.

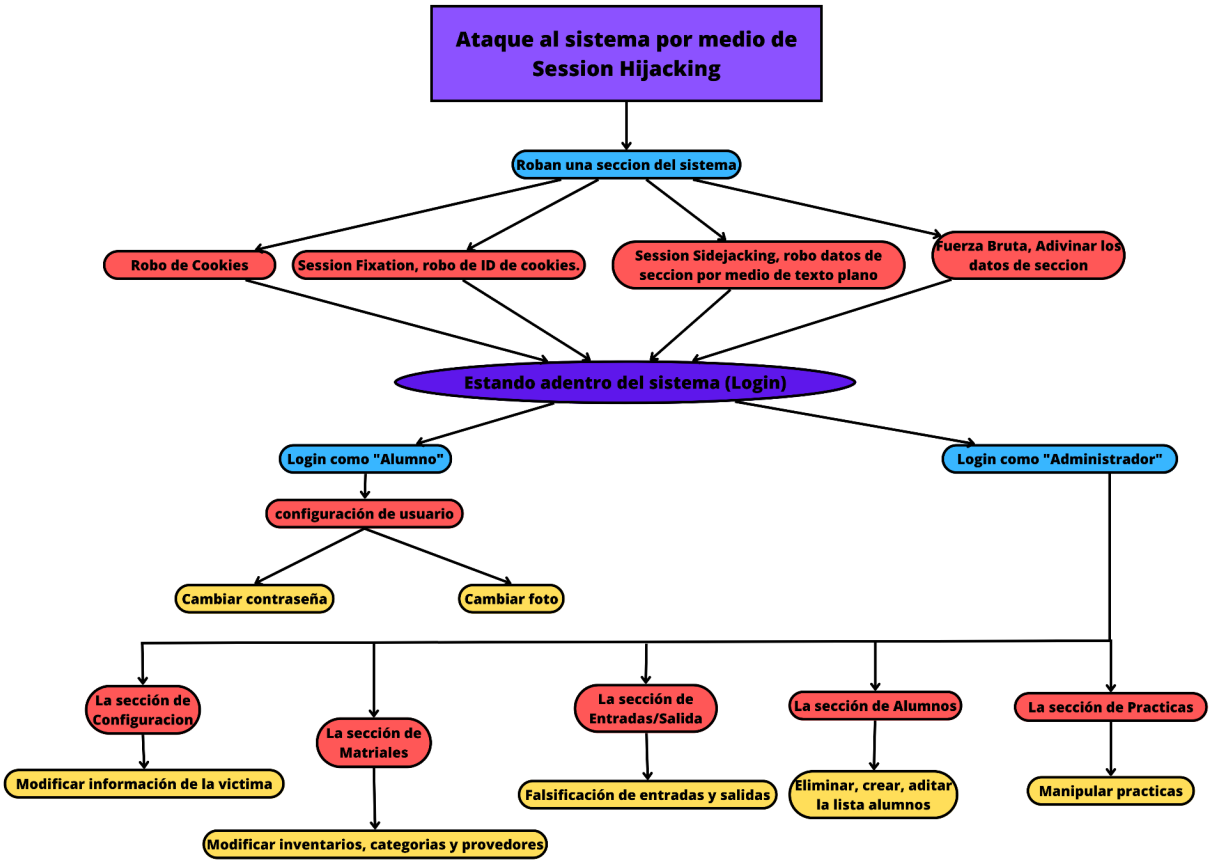

Figura 8: árbol de ataque de Session Hijacking.

SecDevOps

Planificación y Diseño:

Jira: Herramienta de gestión de proyectos para planificar y seguir el progreso del desarrollo seguro.
Threat Modeling Tools: Ayudan a identificar y evaluar posibles amenazas y vulnerabilidades en el diseño del sistema.

Desarrollo:

Static Application Security Testing (SAST) Tools: Analizan el código fuente en busca de posibles vulnerabilidades durante la fase de desarrollo. Ejemplos incluyen SonarQube, Checkmarx, Veracode.
Dependency Management Tools: Detectan y manejan las dependencias de software para asegurar que no se utilicen componentes vulnerables. Ejemplos incluyen OWASP Dependency-Check, Snyk.
Versión Control Systems: Como Git, permiten rastrear los cambios en el código y facilitan la colaboración segura en el desarrollo.

Pruebas:

Dynamic Application Security Testing (DAST) Tools: Simulan ataques en tiempo real para evaluar la seguridad de las aplicaciones en ejecución. Ejemplos incluyen OWASP ZAP, Burp Suite, Nessus.
Penetration Testing Tools: Permiten realizar pruebas de penetración en el sistema para identificar brechas de seguridad. Ejemplos incluyen Metasploit, Nmap, Wireshark.
Security Test Automation Frameworks: Simplifican y automatizan las pruebas de seguridad. Ejemplos incluyen OWASP Web Testing Environment (WTW), OWASP Security Shepherd.

Implementación:

Infrastructure as Code (IaC) Tools: Permiten definir y gestionar la infraestructura como código para asegurar su configuración y despliegue seguro. Ejemplos incluyen Terraform, AWS CloudFormation, Ansible.
Container Security Tools: Escanean y monitorean imágenes de contenedores en busca de vulnerabilidades y configuraciones incorrectas. Ejemplos incluyen Anchore, Clair, Twistlock.

Operaciones:

Security Information and Event Management (SIEM) Tools: Recolectan y analizan registros y eventos de seguridad para detectar y responder a incidentes. Ejemplos incluyen Splunk, LogRhythm, ELK Stack.
Continuous Security Monitoring Tools: Monitorean y analizan continuamente la infraestructura y las aplicaciones en busca de amenazas y comportamientos anómalos. Ejemplos incluyen Security Onion, OSSEC, Wazuh.

Proyecto Funcional

La aplicación web esta alojado en la siguente URL:
http://laboratorioenfermeria.epizy.com/

Los usuarios con lo que se pueden ingresar son los siguientes:

ADMINISTRADORES

- **USUARIO:** admin@ucol.mx **CONTRASEÑA:** 20230511 **ROL:** Administrador
- **USUARIO:** gestor@ucol.mx **CONTRASEÑA:** 20230512 **ROL:** Gestor
- **USUARIO:** vend@ucol.mx **CONTRASEÑA:** 20230513 **ROL:** Vendedor
- **USUARIO:** resp@ucol.mx **CONTRASEÑA:** 20230514 **ROL:** Responsable

ALUMNOS

- **USUARIO:** alum1@ucol.mx **CONTRASEÑA:** 20230511 **SEM:** 1
- **USUARIO:** alum2@ucol.mx **CONTRASEÑA:** 20230512 **SEM:** 1
- **USUARIO:** alum3@ucol.mx **CONTRASEÑA:** 20230513 **SEM:** 2
- **USUARIO:** alum4@ucol.mx **CONTRASEÑA:** 20230514 **SEM:** 2