Nombre: Jorge Manuel Oyoqui Aguilera
Matrícula: A01711783
Fecha: 28/04/2025
Materia: Construcción de Software y Toma de Decisiones
Grupo: 501
Profesores: Enrique Alfonso Calderón Balderas,
Denisse L. Maldonado Flores, Alejandro Fernández Vilchis

Tecnológico de Monterrey

Preguntas Laboratorio 17

- **¿Qué ventajas tiene escribir el código SQL únicamente en la capa del modelo?**
  - Se pueden **separar las responsabilidades**, de modo que cada capa de la aplicación tenga una única responsabilidad, mejorando así el mantenimiento y la evolución de la aplicación.
  - Se **mejora el proceso del mantenimiento de la página**, de modo que, si se debe modificar algo en la estructura de la base de datos o en cómo hacer las consultas, sólo se debe modificar el modelo, sin tocar cosas como los controladores o las vistas.
  - Se puede **reutilizar código** al reutilizar funciones, o consultas desde diferentes partes de la aplicación, en lugar de repetir el mismo código SQL en varios lugares.
  - Al evitar que el SQL esté disperso en la aplicación, se **mejora la seguridad** de la página. Por ejemplo, ayuda a reducir la posibilidad de inyecciones SQL. También permite aplicar validaciones y sanitizaciones de datos en un solo lugar.
  - Facilita el testing, de modo que puedes probar la lógica del modelo de forma aislada sin depender de alguna otra parte de la aplicación. Esto también mejora la calidad del software, teniendo de esta forma pruebas unitarias más simples.
  - Este modelo es compatible con estándares en desarrollo de software como lo sería el MVC (Modelo-Vista-Controlador) o DAO (Data Access Object).

Geekforgeeks (2025). “MVC Design Pattern”. Recuperado de: https://www.geeksforgeeks.org/mvc-design-pattern/

ORACLE (S.F.). “Objeto de Acceso de Datos”. Recuperado de: https://www.oracle.com/java/technologies/data-access-object.html

Microsoft (2023). “ASP.NET MVC Overview”. Recuperado de: https://learn.microsoft.com/en-us/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview

- **¿Qué es SQL injection y cómo se puede prevenir?**

  Consiste en una inserción de código malicioso usando los campos de entrada de la página, de modo que parte de ese código es ejecutado por la base de datos (si es que no se usa buena seguridad), permitiendo así que los hackers accedan a información privada o sensible o incluso puedan darse altos privilegios dentro de la base de datos.

  Se pueden clasificar en tres tipos:

  - **En banda**: los datos se extraen utilizando el mismo canal que se utiliza para inyectar el código SQL. Este es el ataque más directo, en el que los datos recuperados se presentan directamente en la página web de la aplicación. (OWASP, S.F.)
  - **Fuera de banda:** los datos se recuperan utilizando un canal diferente (por ejemplo, se genera un correo electrónico con los resultados de la consulta y se envía al evaluador). (OWASP, S.F.)
  - **Inferencial o Ciego:** No hay transferencia real de datos. Aun así, el evaluador puede reconstruir la información enviando solicitudes específicas y observando el comportamiento resultante del servidor de BD. (OWASP, S.F.)

  Una forma de prevenir SQL Injection es realizar pruebas controladas de inyección SQL, conocidas como *SQL Injection Testing*. Este proceso, realizado por los mismos desarrolladores o por equipos de seguridad, permite identificar posibles puntos vulnerables de la página y medir la gravedad de los ataques (OWASP, S.F.).

  Basándose en los resultados de estas pruebas, se pueden aplicar dstintas defensas:

  - **Uso de sentencias preparadas y consultas parametrizadas:** Separar el código SQL de los datos que recibe la consulta. Esta técnica obliga al servidor de bases de datos a tratar la entrada como datos y no como código ejecutable.
  - **Uso correcto de procedimientos almacenados:** Definir rutinas en la base de datos que gestionen consultas seguras, evitando construir SQL dinámico con entradas no validadas.
  - **Validación de entrada mediante listas blancas:** Aceptar únicamente entradas que cumplan criterios específicos esperados (por ejemplo, solo números para un campo de edad), rechazando todo lo que no esté explícitamente permitido.
  - **Escape de caracteres especiales:** Aunque no es el método más recomendable como única defensa, en algunos casos puede ayudar a evitar la ejecución de comandos maliciosos si se combinan con otras prácticas seguras.

Jack Leonard (2015). “SQL Injection Attacks, Visually Explained.”. Recuperado de: https://medium.com/visually-explained/sql-injection-attacks-visually-explained-c71b5f9e1af2

OWASP (S.F.). “Testing for SQL Injection”. Recuperado de: https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/07-Input_Validation_Testing/05-Testing_for_SQL_Injection

OWASP (S.F.). “SQL Injection Prevention Cheat Sheet”. Recuperado de: https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html