

UBICACIÓN: Unidad 1. Selección de arquitecturas y herramientas de programación

OBJETIVOS:

Aprender a diseñar un sistema de reservas en línea utilizando conceptos de modelos cliente/servidor.

Saber Integrar tecnologías del lado del cliente y del servidor para una aplicación web funcional.

Aplicar conocimientos sobre bases de datos y seguridad en el desarrollo de la aplicación.

TIEMPO ESTIMADO DE REALIZACIÓN: 90 minutos.

ENUNCIADO:

Tu tarea es desarrollar un plan conceptual para un sistema de reservas en línea, como para un hotel o restaurante. Este sistema permitirá a los usuarios ver la disponibilidad de habitaciones o mesas y realizar reservas. Deberás considerar la integración de tecnologías del lado del cliente y del servidor, la gestión de bases de datos y la implementación de medidas de seguridad.

Desarrollo de la actividad:

Diseño del Sistema:

Define la arquitectura del sistema (dos niveles, tres niveles, n-capas) y justifica tu elección.

Arquitectura de tres niveles. Así separamos las funciones del sistema (frontend, backend y lógica del negocio) esto facilita el mantenimiento, seguridad y escalabilidad.

Tendríamos separados la interfaz, base de datos y las consultas y verificación de reservas.

Describe cómo se manejarán las solicitudes del cliente y las respuestas del servidor.

Manejo de solicitudes y respuestas:

El cliente accede al sistema de reservas a través de un navegador web donde podrá consultar la disponibilidad, hacer reservas o consultar las reservas mediante formularios.

El servidor recibe las solicitudes del cliente y responde usando para ello la lógica del negocio que podría ser consultar las bases de datos existentes.

Usando JavaScript por ejemplo, sin recargar la página se toma la respuesta en formato por ejemplo JSON y se muestra en la interfaz.

Tecnologías del Lado del Cliente:

Elige y justifica las tecnologías de frontend para la interfaz de usuario del sistema de reservas.

JavaScript: Ya que de forma dinámica maneja la interacción cliente/servidor, evitando que haya que recargar la página al hacer actualizaciones en tiempo real.

Se puede usar por ejemplo Vue.js para facilitar aún más la creación de interfaces dinámicas.

Tecnologías del Lado del Servidor:

Selecciona un lenguaje de programación del lado del servidor (PHP, Python, Java, etc.) y explica cómo se utilizará para procesar las reservas.

Usaría PHP en este caso, al ser un sistema de reservas ya que PHP cuenta con un gran abanico de soporte a bases de datos que es algo fundamental en este caso.

Además si usamos un framework como Laravel añadimos herramientas avanzadas para manejar la seguridad y las propias bases de datos.

Describe cómo se gestionarán las sesiones de usuario y la seguridad en el servidor.

Las sesiones de usuario se manejarán con cookies o identificadores únicos. Se implementarán medidas de seguridad para proteger la información del usuario.

Base de Datos:

Elige entre una base de datos SQL o NoSQL para gestionar las reservas y justifica tu elección.

SQL: En un sistema de reservas, las relaciones entre entidades (usuarios, reservas, habitaciones o mesas) son clave, por lo que un sistema relacional es la mejor opción.

Esboza un modelo de datos simple para tu sistema.

Usuarios: ID, nombre, correo.

Habitaciones/Mesas: ID, número, tipo, estado.

Reservas: ID, usuario, habitación/mesa, fechas.

Seguridad:

Propón medidas para proteger el sistema contra inyecciones SQL, XSS y otros ataques comunes.

Describe cómo se implementará la encriptación y la autenticación segura.

1. Escapar todo dato ingresado, así se evita ataques como Cross-Site (XSS).
2. Usar consultas preparadas para prevenir inyecciones SQL.
3. Usar la autenticación segura, algunos métodos serían mediante uso de QR, biometría o doble factor de autenticación.
4. Cifrar datos mediante protocolo HTTP evitando así que puedan tomarse datos entre la interacción cliente/servidor.

CAPACIDAD Y CRITERIOS DE EVALUACIÓN:

CE

PUNTUACIÓN

0%-20%

20%-40%

40%-60%

60%-80%

80%-100%

Claridad y justificación en la elección de la arquitectura del sistema y las tecnologías utilizadas

El alumno no proporciona una explicación clara o justificación de las elecciones realizadas. Las tecnologías y la arquitectura elegidas no son adecuadas para el proyecto, o no hay evidencia de una elección consciente.

La justificación de las elecciones es mínima o poco clara. Las tecnologías y la arquitectura seleccionadas son parcialmente adecuadas, pero hay errores significativos o incoherencias en la elección.

Las elecciones de tecnología y arquitectura son generalmente adecuadas, con una justificación básica proporcionada. Sin embargo, hay áreas donde la elección podría haber sido más adecuada o mejor explicada.

Buena elección de tecnologías y arquitectura, con justificaciones claras y bien fundamentadas. Puede haber pequeñas áreas de mejora en la selección o en la explicación de las elecciones.

Excelente elección de tecnologías y arquitectura, perfectamente alineadas con los requisitos del proyecto. Las justificaciones son claras, detalladas y bien razonadas, demostrando una comprensión profunda de cómo estas elecciones impactan en el proyecto.

Comprensión de la integración entre el cliente y el servidor, y la gestión de bases de datos

Falta de comprensión básica de la integración cliente-servidor y la gestión de bases de datos. Las explicaciones son incorrectas o faltan por completo.

Comprensión limitada de la integración cliente-servidor y la gestión de bases de datos. Hay errores significativos en la explicación o en la aplicación de estos conceptos.

Comprensión adecuada de la integración cliente-servidor y la gestión de bases de datos. La explicación es correcta en su mayoría, pero falta detalle o hay pequeños errores.

Buena comprensión de la integración cliente-servidor y la gestión de bases de datos. La explicación es clara y mayormente completa, con solo detalles menores faltantes o levemente incorrectos.

Excelente comprensión de la integración cliente-servidor y la gestión de bases de datos. La explicación es detallada, precisa y muestra una comprensión profunda de cómo estos elementos trabajan juntos en el desarrollo web.

Propuestas efectivas para la seguridad del sistema

No se proponen medidas de seguridad, o las propuestas son inadecuadas y no protegerían efectivamente el sistema.

Las propuestas de seguridad son básicas o genéricas, con una aplicación limitada o incorrecta en el contexto del proyecto.

Propuestas de seguridad adecuadas, pero faltan detalles importantes o hay una comprensión incompleta de cómo implementar estas medidas efectivamente.

Buenas propuestas de seguridad, bien integradas en el diseño del sistema. Puede haber pequeñas áreas de mejora en la profundidad o la aplicación de las medidas de seguridad.

Excelentes propuestas de seguridad, perfectamente integradas en el diseño del sistema. Las medidas son completas, bien explicadas y demuestran una comprensión avanzada de la seguridad en el desarrollo web.

SE VALORARÁ POSITIVAMENTE:

Creatividad en el diseño del sistema.

Enfoque práctico y realista en la implementación.

Soluciones innovadoras para la seguridad y la experiencia del usuario.

¿QUÉ HAY QUE ENTREGAR Y CÓMO HACERLO?

Los estudiantes deberán entregar un informe en formato de documento de texto que incluya el diseño del sistema, la selección y justificación de todas las tecnologías y estrategias propuestas. El documento debe ser subido a través del enlace proporcionado por el equipo docente.