



GOBIERNO DEL
ESTADO DE MÉXICO

MATERIA: PROGRAMACIÓN CLIENTE SERVIDOR

UNIDAD DE APRENDIZAJE: 1

CUATRIMESTRE: 5 GRUPO: 1522 IS

PROFESOR: EMMANUEL TORRES SERVIN

PERIODO: ENERO-ABRIL 2022

INGENIERÍA EN SOFTWARE



LISTA DE COTEJO: CASOS DE ESTUDIO ASIGNADO POR EL PROFESOR

COMPETENCIA DE LA ASIGNATURA:

Diseñar software mediante el diagnóstico de los requerimientos, en la teoría de los diseños de software, metodologías de modelado, estructuración de datos, interfaces para su desarrollo y validación, considerando estándares y normatividad aplicable para determinar la representación técnica del software cumpliendo con las necesidades del cliente.

RESULTADO DE APRENDIZAJE:

Identificación y aplicación de los procesos básicos del pensamiento, utilización correcta de los procesos básicos del pensamiento.

NOMBRE DEL ALUMNO: Brandon Duran Rocha

	ASPECTO A EVALUAR	VALOR DEL ÍTEM	VALOR OBTENIDO
1	COMPREENSIÓN DE LA INFORMACIÓN, PROBLEMÁTICA Y COMPONENTES DEL CASO, EL ALUMNO: (ED)		
1	Reconoce la comunicación de dispositivos de red y las arquitecturas	10	
2	Realiza el Diagrama de componentes de la arquitectura Cliente/Servidor	10	
3	Cuadro comparativo entre los modelos IAAS, PAAS, SAAS y Cliente/Servidor	10	
4	Presenta una propuesta técnica de arquitectura Cliente/Servidor contemplando los modelos de cómputo en la nube.	10	
2	EL ALUMNO ELABORA UN PORTAFOLIO DE EVIDENCIAS A PARTIR DE UN CASO DE ESTUDIO SOBRE ARQUITECTURAS WEB: (ED, EP)		
1	Crea un Repositorio	5	
2	El código del programa web se encuentra realizado de manera correcta.	10	
3	Se consideran estándares y buenas prácticas.	5	
4	Cumple con los requerimientos solicitados.	10	
5	El programa es funcional.	10	
3	LA ESTRUCTURA Y PRESENTACIÓN DEL CASO RESUELTO CUMPLE CON: (EP, ED).		
1	Carátula.	2.5	
2	Índice de contenido.	2.5	
3	Respeto por el formato sugerido para la resolución del caso.	2.5	
4	Redacción técnico-profesional.	2.5	
5	Ortografía correcta.	2.5	
6	Redacción clara.	2.5	
7	Presentación profesional impresa y/o electrónica según lo estipulado por el profesor	5	
	PUNTUACIÓN OBTENIDA	100%	

COMPETENCIA ALCANZADA: _____ **CALIFICACIÓN:** _____

OBSERVACIONES GENERALES: _____

NOMBRE Y FIRMA DEL EVALUADOR

NOMBRE Y FIRMA DEL ALUMNO

LISTA DE COTEJO

Programación cliente servidor

DOCENTE TORRES SERVIN EMMANUEL

ESTUDIANTE BRANDON DURAN ROCHA

Contenido

Reconoce la comunicación de dispositivos de red y las arquitecturas	3
Realiza el Diagrama de componentes de la arquitectura Cliente/Servidor	3
Cuadro comparativo entre los modelos IAAS, PAAS, SAAS y Cliente/Servidor	4
Presenta una propuesta técnica de arquitectura Cliente/Servidor contemplando los modelos de cómputo en la nube.	6

Reconoce la comunicación de dispositivos de red y las arquitecturas

Los protocolos para la transmisión de datos en internet más importantes son TCP (Protocolo de Control de Transmisión) e IP (Protocolo de Internet). De manera conjunta (TCP/IP) podemos enlazar los dispositivos que acceden a la red, algunos otros protocolos de comunicación asociados a internet son POP, SMTP y HTTP.

Estos los utilizamos prácticamente todos los días, aunque la mayoría de los usuarios no lo sepan ni conozcan su funcionamiento. Estos protocolos permiten la transmisión de datos desde nuestros dispositivos para navegar a través de los sitios, enviar correos electrónicos, escuchar música online, etc.

Hay gran cantidad de protocolos que han aportado soluciones diferentes a los problemas de red: Netbeui, AppelTalk, TCP/IP, etc. Entre ellos destaca hoy en día TCP/IP que se ha impuesto como estándar de facto en todo tipo de redes. Hoy en día incluso los protocolos propietarios se implementan como interfaces de TCP/IP.

Realiza el Diagrama de componentes de la arquitectura Cliente/Servidor

Cuadro comparativo entre los modelos IAAS, PAAS, SAAS y Cliente/Servidor

	IAAS	PAAS	SAAS	CLIENTE/SERVIDOR
Propósito	Infraestructuras como servicio. Las empresas contratan la infraestructura de hardware a un tercero a cambio de una cuota o alquiler. La contratación de este hardware permite elegir la capacidad de proceso (procesadores), la memoria a utilizar (memoria RAM) y el espacio de almacenamiento (disco duro).	Ofrece plataformas como servicios. En estas plataformas se pueden lanzar aplicaciones como bases de datos, middleware, herramientas de desarrollo, servicios de inteligencia empresarial, etc.	Aloja el software de la empresa, así como sus datos, en servidores externos a la misma, y paga una cuota por su utilización.	Es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta
Características	<ul style="list-style-type: none"> • Escalabilidad y flexibilidad: Los recursos están disponibles a medida que los necesitas. • Pago por uso: Es un modelo que actúa bajo demanda, por lo que solo pagarás por lo que utilices. • Reducción de costes: Elimina el coste de implementación y mantenimiento de hardware físico. 	<ul style="list-style-type: none"> • Incluye bases de datos y middleware • Incluye servicios para desarrollar, probar, implementar, alojar y mantener aplicaciones dentro del mismo entorno del IDE • Provisión de servicios que permite que múltiples usuarios utilicen simultáneamente el mismo IDE. 	<ul style="list-style-type: none"> • Acceso y administración por una red. • Permite gestionar las actividades desde las sedes centrales. • Actualización centralizada: no es necesario que descargues parches. 	<ul style="list-style-type: none"> • Es quien inicia solicitudes o peticiones, tiene por tanto un papel activo en la comunicación (dispositivo maestro o amo). • Espera y recibe las respuestas del servidor. • Por lo general, puede conectarse a varios servidores a la vez.
Ventajas	<ul style="list-style-type: none"> • Las organizaciones eligen IaaS porque a menudo es más fácil, rápido y rentable operar 	<ul style="list-style-type: none"> • No necesitan invertir en 	<ul style="list-style-type: none"> • Se reducen los costos de infraestructura, 	<ul style="list-style-type: none"> • Facilita la integración entre diferentes sistemas y comparte información

	<p>una carga de trabajo sin tener que comprar, administrar y respaldar la infraestructura subyacente.</p> <ul style="list-style-type: none"> • Es un modelo de servicio en la nube eficaz para cargas de trabajo que son temporales, experimentales o que cambian inesperadamente. • En general, los clientes de IaaS pagan por usuario, generalmente por hora, semana o mes. 	<p>infraestructura física:</p> <ul style="list-style-type: none"> • Flexibilidad: Los clientes pueden disfrutar de un control total sobre las herramientas que se instalan en sus plataformas, y crear una plataforma perfectamente adaptada a sus necesidades concretas. • Seguridad: Se ofrecen diversos mecanismos de seguridad, que incluyen la protección de los datos y la realización y recuperación de copias de seguridad. 	<p>soporte y otros ya que el encargado de realizar estas es el proveedor de servicios.</p> <ul style="list-style-type: none"> • Los usuarios pueden acceder desde cualquier lugar a los datos e información de las aplicaciones siempre que cuenten con acceso a Internet. • No se necesita la compra de una licencia para el uso de una aplicación, sino que se paga el uso de esta por un tiempo determinado. 	<p>permitiendo por ejemplo que las máquinas ya existentes puedan ser utilizadas mediante una interfaz más amigable para el usuario. De esta manera podemos integrar varias PCs con sistemas medianos y grandes sin necesidad de que todos tengan que utilizar el mismo sistema operativo.</p> <ul style="list-style-type: none"> • Al favorecer el uso de las interfaces de gráficas interactivas, los sistemas construidos bajo este esquema tienen una mayor interacción con el usuario. • La estructura modular facilita de más la integración de nuevas tecnologías y el crecimiento de la infraestructura computacional favoreciendo así la estabilidad de las soluciones.
Ejemplos	<ul style="list-style-type: none"> • Amazon Web Services • Microsoft Azure 	<ul style="list-style-type: none"> • Google App Engine • Bungee Connect 	<ul style="list-style-type: none"> • Microsoft Office 365 • WordPress 	<ul style="list-style-type: none"> • Servidor de correo • Sistema DNS

Presenta una propuesta técnica de arquitectura Cliente/Servidor contemplando los modelos de cómputo en la nube.

La idea propuesta sería la de un servidor de archivos donde estén almacenados en forma digital los libros de la biblioteca, esta podrá acceder dentro de la universidad, esto será a través de la red de la universidad. Otro requisito para poder acceder será contar con una matrícula vigente y los archivos expirarán después de cierto tiempo.