



LISTA DE COTEJO

PARTE 1

PROGRAMACION CLIENTE/SERVIDOR

UNIVERSIDAD POLITECNICA DE TECAMAC

MAESTRO: EMMANUEL TORRES SERVIN

MATRICULA: 1320114042

GRUPO: 2522IS

ESTUDIANTE ERIK SAMUEL HERNANDEZ
GARCIA

MATERIA: PROGRAMACIÓN CLIENTE SERVIDOR

UNIDAD DE APRENDIZAJE: 1

CUATRIMESTRE: 5 GRUPO: 2522IS

PROFESOR: EMMANUEL TORRES SERVIN



GOBIERNO DEL
ESTADO DE MÉXICO



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: ERIK SAMUEL HERNANDEZ GARCIA

	ASPECTO PARA EVALUAR	VALOR DEL ITEM	VALOR OBTENIDO
1	COMPRENSIÓN 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%	

OBSERVACIONES GENERALES: _____

NOMBRE Y FIRMA DEL EVALUADOR

NOMBRE Y FIRMA DEL ALUMNO

INDICE

Reconoce la comunicación de dispositivos de red y las arquitecturas 3

Realiza el Diagrama de componentes de la arquitectura Cliente/Servidor..... 0

Cuadro comparativo entre los modelos IAAS, PAAS, SAAS y Cliente/Servidor 1

Presenta una propuesta técnica de arquitectura Cliente/Servidor contemplando los modelos de cómputo en la nube.
..... 0

Reconoce la comunicación de dispositivos de red y las arquitecturas

Arquitecturas de redes

La conexión entre equipos informáticos es posible gracias a los protocolos de comunicaciones.

Un protocolo de comunicaciones es un conjunto de reglas perfectamente organizadas y convenidas de mutuo acuerdo entre los participantes en una comunicación, cuya misión es permitir el intercambio de información entre los dos dispositivos, detectando los posibles errores que se produzcan.

El conjunto de protocolos que facilitan la comunicación entre dispositivos se le denomina arquitectura de la red.

CARACTERÍSTICAS DE LAS ARQUITECTURAS DE RED

El software de red hace la misma función con el hardware de red que el sistema operativo de una máquina hace con el hardware de un ordenador.

La arquitectura de una red viene definida por tres características que definen la tecnología que se emplea en la red:

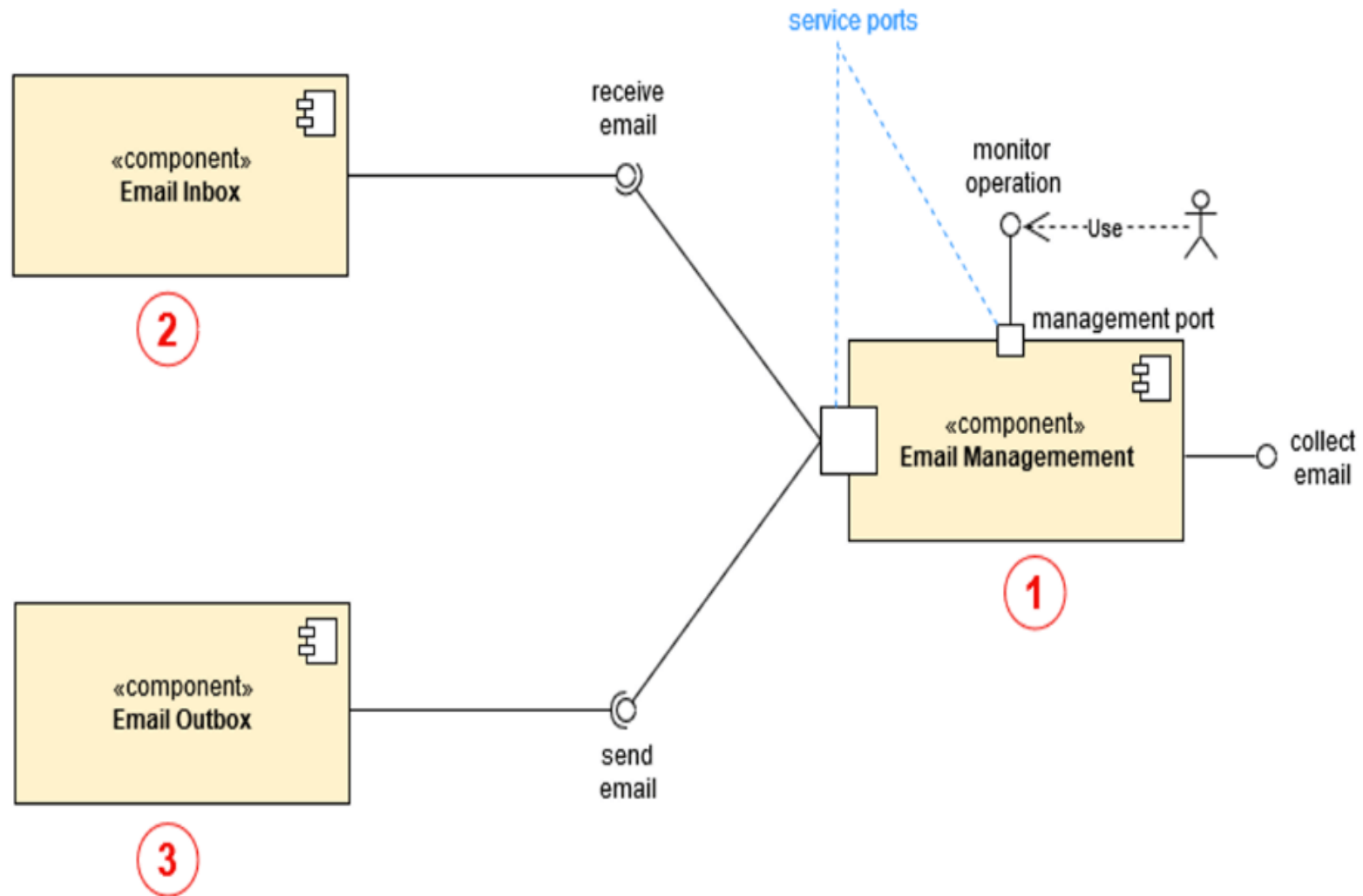
- **Topología:** la topología es la organización de su cableado. Esto define la interconexión de las estaciones y el camino de transmisión de datos sobre el medio de comunicación.
- **Método de acceso a la red:** Una característica de una red es que todos los elementos comparten el medio de transmisión de la información. El método de acceso define la forma y protocolo mediante el cual cada elemento de la red accede al medio.
- **Protocolo de comunicaciones:** El protocolo está constituido por las reglas y procedimientos utilizados en la red para realizar la comunicación. Estas reglas tienen en cuenta el método para corregir errores, establecer la comunicación, etc.

PRINCIPALES PROBLEMAS EN EL DISEÑO DE UNA ARQUITECTURA DE RED

Al diseñar un sistema de comunicación en una red se abordan problemas de gran complejidad. Los problemas más importantes en el diseño son los siguientes:

- **Encaminamiento:** En una red es necesario elegir el camino de un mensaje para llegar a destino. usualmente el camino más corto o el que tenga menor tráfico.
- **Direccionamiento:** Permite identificar el ordenador al que hay que enviar un mensaje. Un ordenador suele tener asignadas varias direcciones diferentes, relacionadas con diferentes niveles o capas de su arquitectura de red.
- **Acceso al medio:** Como se comparte el medio de transmisión entre los elementos de una red es necesario evitar colisiones ya que si no fuera así los mensajes se mezclarían y no sería posible interpretarlos.
- **Saturación del receptor:** Un emisor rápido y un receptor lento saturarían al receptor generando pérdida de datos.
- **Mantenimiento del orden:** Es necesario mantener el orden de los fragmentos de un mensaje sino resultaría imposible su interpretación.
- **Control de errores:** El protocolo debe revisar si se ha recibido correctamente la información enviada ya que los medios de transmisión son imperfectos.
- **Multiplexación:** En ciertos sitios de una red y por razones económicas, se debe compartir un medio entre comunicaciones que no tienen que ver entre sí. Se debe asegurar la integridad de cada mensaje.

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
Ventajas	La empresa ya no necesita adquirir hardware para implementar infraestructura informática. Permite optimizar los gastos corrientes.	permite niveles más altos de programación con una complejidad extremadamente reducida; el desarrollo general de la aplicación puede ser más eficaz, ya que se tiene una infraestructura built-in; y el mantenimiento y mejora de la aplicación es más sencillo	<ul style="list-style-type: none"> ➤ Se reducen los costos ➤ Los usuarios pueden acceder desde cualquier lugar a los datos e información ➤ No se necesita la compra de una licencia para el uso de una aplicación 	<ul style="list-style-type: none"> ➤ Centralización de los recursos ➤ Mejora de la seguridad. ➤ Escalabilidad de la instalación.
Desventajas	El proveedor es el responsable de la disponibilidad y seguridad de los servicios.	es posible que los desarrolladores no pueden utilizar todas las herramientas convencionales (bases de datos relacionales, con joins irrestrictos	<ul style="list-style-type: none"> ➤ El usuario no puede hacer modificaciones a las aplicaciones 	<ul style="list-style-type: none"> ➤ El servidor puede ser propenso a ataques de denegación de servicio. ➤ Los paquetes de datos se modificarán o falsificarán durante la transmisión.
Características	<ul style="list-style-type: none"> ➤ Escalabilidad y flexibilidad. ➤ 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"> ➤ Paas debe asegurar la escalabilidad del sistema al desarrollador. ➤ Servicios de Desarrollo, testeo, implantación, hosting, y mantenimiento de aplicaciones. 	<ul style="list-style-type: none"> ➤ Actualización centralizada: no es necesario que descargues parches. ➤ Mayor integración con una mayor red de software de comunicación. 	<ul style="list-style-type: none"> ➤ Espera y recibe las respuestas del servidor. ➤ Por lo general, puede conectarse a varios servidores a la vez.

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

<https://blog-personal1.herokuapp.com/>