

# UNIDAD 1

PROYECTO DE EVALUACIÓN

HERNÁNDEZ SILVA RAÚL ABRAHAM  
1321124285

GUTIÉRREZ GARCÍA HAROLD GUILLERMO  
1321124281

ESCALANTE SUÁREZ ALDO MJAIL  
1320114010



UNIVERSIDAD POLITÉCNICA  
DE TECÁMAC

PROFESOR: EMMANUEL TORRES SERVIN  
CLIENTE-SERVIDOR

## índice.

<b>ÍNDICE.</b>	<b>1</b>
<b>DESCRIPCIÓN DEL PROBLEMA</b>	<b>2</b>
<b>REPORTE SOBRE LA COMUNICACIÓN DE DISPOSITIVOS DE RED Y LAS ARQUITECTURAS</b>	
<b>CLIENTE SERVIDOR</b>	<b>2</b>
<b>DIAGRAMA DE COMPONENTES DE LA ARQUITECTURA CLIENTE/SERVIDOR</b>	<b>3</b>
<b>CUADRO COMPARATIVO ENTRE LOS MODELOS IAAS, PAAS, SAAS Y</b>	
<b>CLIENTE/SERVIDOR</b>	<b>5</b>
<b>PROPUESTA TÉCNICA DE ARQUITECTURA CLIENTE/SERVIDOR CONTEMPLANDO LOS</b>	
<b>MODELOS DE CÓMPUTO EN LA NUBE</b>	<b>6</b>
<b>JUSTIFICACIÓN DE SELECCIÓN DE SISTEMA OPERATIVO MÓVIL</b>	<b>7</b>
<b>JUSTIFICACIÓN DE LENGUAJE DE PROGRAMACIÓN MÓVIL</b>	<b>8</b>

## **Descripción del problema**

El problema que enfrenta Juan es el control de su empresa de patitos de hule, que cuenta con 40 empleados y se encuentra en constante crecimiento. Debido a un presupuesto limitado, ha perdido el control de sus empleados en ciertas áreas de la empresa. Para solucionar esto, Juan decide contratar programadores para desarrollar una aplicación que le permita administrar su empresa de manera general, sin embargo, no tiene conocimiento en el mundo del desarrollo de software. Es por ello que se requiere de una propuesta técnica que contemple la comunicación de dispositivos de red y las arquitecturas cliente servidor, para ayudar a Juan a solucionar su problema y lograr un control efectivo de su empresa.

## **Reporte sobre la comunicación de dispositivos de red y las arquitecturas cliente servidor**

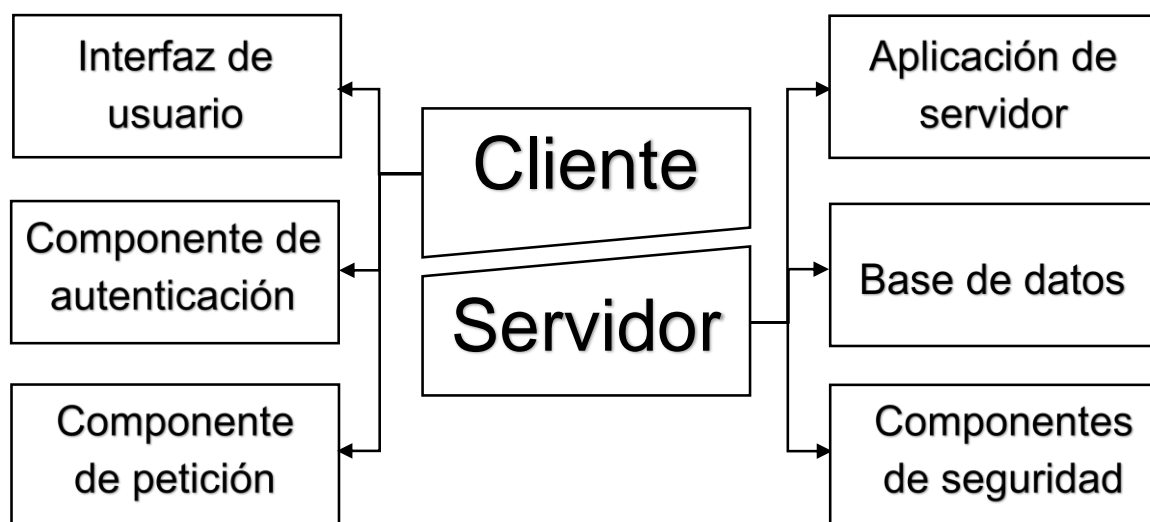
En la empresa de Juan, que produce patitos de hule, el crecimiento constante ha generado la necesidad de tener una mejor gestión de los empleados y las áreas de la empresa. Para ello, Juan ha decidido contratar a programadores para desarrollar una aplicación que permita una administración general de la empresa.

La comunicación de dispositivos de red y la arquitectura cliente-servidor son cruciales en este contexto. La arquitectura cliente-servidor es un modelo en el que un dispositivo cliente solicita servicios o recursos a un dispositivo servidor. El servidor proporciona los recursos o servicios solicitados y el cliente los utiliza.

En este caso, la aplicación desarrollada para la empresa de Juan funcionará como un servidor, proporcionando recursos y servicios a los dispositivos clientes, que serían los computadores y/o dispositivos móviles de los empleados. La comunicación entre el servidor y los clientes se realiza a través de la red de la empresa, permitiendo la gestión y control de las áreas y empleados desde cualquier lugar y en cualquier momento.

Además, existen diferentes modelos de cómputo en la nube, como IAAS (Infrastructure as a Service), PAAS (Platform as a Service) y SAAS (Software as a Service), que permiten la implementación de arquitecturas cliente-servidor en la nube. Estos modelos ofrecen una mayor flexibilidad y escalabilidad a la arquitectura, lo que es ideal para una empresa en constante crecimiento como la de Juan.

### Diagrama de componentes de la arquitectura Cliente/Servidor



Cliente:

- **Interfaz de usuario:** Una interfaz amigable y fácil de usar para los empleados de la empresa que permita acceder a los datos y realizar tareas específicas. Esto puede ser una aplicación web o de escritorio según la preferencia del usuario.
- **Componente de autenticación:** Este componente se encarga de verificar la identidad del usuario y permitir el acceso a la aplicación solo a los usuarios autorizados.

- Componente de petición: Este componente se encarga de enviar solicitudes al servidor y recibir respuestas.

Servidor:

- Aplicación de servidor: Este componente es el corazón de la aplicación y es el encargado de realizar todas las tareas y mantener los datos de la empresa.
- Base de datos: Este componente almacena todos los datos y información de la empresa y permite su acceso y modificación a través de la aplicación de servidor.
- Componente de seguridad: Este componente se encarga de proteger los datos y la información de la empresa y garantizar que solo los usuarios autorizados tengan acceso a ella.

Esta arquitectura permite a Juan administrar de manera eficiente su empresa a través de una interfaz amigable y fácil de usar. Además, la separación de responsabilidades entre el cliente y el servidor asegura una mayor seguridad y escalabilidad de la aplicación.

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

Característica	IAAS (Infrastructure as a Service)	PAAS (Platform as a Service)	SAAS (Software as a Service)	Cliente/Servidor
<b>Definición</b>	Provee infraestructura informática en la nube	Provee una plataforma para desarrollar y ejecutar aplicaciones	Provee software a través de internet	Modelo tradicional de aplicaciones en donde el cliente accede a los recursos del servidor
<b>Ejemplos</b>	AWS, Microsoft Azure, Google Cloud	Heroku, Force.com, Google App Engine	Salesforce, Microsoft Office 365, Gmail	Aplicaciones de escritorio como Microsoft Word
<b>Control</b>	Alta flexibilidad y control sobre la infraestructura	Limitado control sobre la plataforma, pero mayor velocidad y eficiencia en el desarrollo	Mínimo control, pero mayor comodidad y accesibilidad	Alta flexibilidad y control sobre la aplicación
<b>Costo</b>	Costos más altos, pero mayor escalabilidad y personalización	Costos intermedios, pero mayor eficiencia en el desarrollo y menor administración	Costos más bajos, pero menor control y personalización	Costos intermedios, pero mayor flexibilidad y control

En resumen, cada modelo tiene sus fortalezas y debilidades en términos de control, costo y eficiencia. Es importante evaluar cuidadosamente las necesidades de la empresa y los recursos disponibles antes de tomar una decisión sobre cuál modelo es el adecuado. En el caso de Juan, dado su presupuesto limitado y su falta de conocimiento en el mundo del desarrollo de software, podría ser una buena opción considerar un modelo SAAS para obtener una solución de bajo costo y fácil de usar para administrar su empresa.

## **Propuesta técnica de arquitectura Cliente/Servidor contemplando los modelos de cómputo en la nube**

Para ayudar a Juan en su empresa de patitos de hule, se puede proponer una arquitectura de cliente-servidor que incluya elementos de los modelos de cómputo en la nube (IaaS, PaaS, SaaS). Esta propuesta técnica podría contemplar lo siguiente:

Software como servicio (SaaS): Se podría utilizar una solución SaaS para algunas funcionalidades específicas, como el seguimiento de la nómina y los horarios de los empleados, o el seguimiento de inventario y ventas. Esto permitiría ahorrar costos y tiempo en el desarrollo de estas funcionalidades, y brindar una solución probada y eficiente.

En cuanto a la arquitectura cliente-servidor, la aplicación se podría desarrollar utilizando una arquitectura de 3 capas, con una interfaz de usuario en el lado del cliente, una capa de aplicación en el lado del servidor que se encargaría de la lógica de negocios y la integración con los servicios en la nube, y una capa de datos en el lado del servidor que almacenaría la información de la empresa.

En resumen, esta propuesta técnica combinaría elementos de los modelos de cómputo en la nube con una arquitectura cliente-servidor para brindar una solución eficiente, escalable y asequible para la empresa de patitos de hule de Juan.

## Justificación de selección de sistema operativo móvil

En este caso, la selección de un sistema operativo móvil para la aplicación de administración es crítica. Juan debe elegir un sistema operativo que ofrezca una amplia gama de aplicaciones y herramientas para su uso en su empresa. Además, es importante que el sistema operativo sea fácil de usar para sus empleados y que proporcione una interfaz intuitiva para que puedan acceder a la información y las funciones que necesitan de manera rápida y eficiente.

Uno de los sistemas operativos móviles más populares es Android, que ofrece una amplia gama de aplicaciones y herramientas para el desarrollo de aplicaciones móviles. Android es un sistema operativo abierto, lo que significa que los desarrolladores pueden acceder a su código fuente y modificarlo según sus necesidades. Además, Android cuenta con una amplia comunidad de desarrolladores que pueden ayudar con el desarrollo y la solución de problemas.

La justificación de selección de Android como sistema operativo móvil para el tema de Juan es el siguiente:

**Amplia disponibilidad:** Android es el sistema operativo móvil más utilizado en todo el mundo, lo que significa que hay una amplia gama de dispositivos compatibles disponibles en el mercado. Esto significa que los empleados de Juan pueden elegir un dispositivo que se ajuste a sus necesidades y presupuesto.

- **Personalización:** Android ofrece una amplia gama de opciones de personalización, lo que significa que se puede adaptar fácilmente a las necesidades específicas de Juan y su empresa. Por ejemplo, se puede personalizar la interfaz de usuario y las configuraciones de seguridad para garantizar que se cumplan las necesidades de privacidad y seguridad de la empresa.
- **Gran cantidad de aplicaciones disponibles:** La tienda de aplicaciones de Android, Google Play Store, ofrece una amplia gama de aplicaciones que pueden ser útiles para el negocio de Juan. Por ejemplo, puede encontrar



aplicaciones de gestión de tareas, seguimiento de horas de trabajo, gestión de inventario, etc.

- Integración con Google Workspace: Juan podría estar usando Google Workspace para la gestión de correos electrónicos, calendarios y documentos. Android se integra perfectamente con estas herramientas, lo que significa que sus empleados pueden acceder a ellas de manera fácil y eficiente en sus dispositivos móviles.

## **Justificación de lenguaje de programación móvil**

Java es una excelente opción para desarrollar aplicaciones móviles en el sistema operativo Android. Algunas de las razones por las que Java es una buena elección incluyen:

- Amplia disponibilidad de bibliotecas: Java cuenta con una amplia gama de bibliotecas y herramientas disponibles para el desarrollo de aplicaciones móviles, lo que permite a los desarrolladores implementar funcionalidades avanzadas con mayor rapidez y eficiencia.
- Multiplataforma: Java es un lenguaje de programación multiplataforma, lo que significa que los desarrolladores pueden escribir código una sola vez y ejecutarlo en diferentes plataformas, incluidas las aplicaciones móviles de Android.
- Gran comunidad de desarrolladores: Java cuenta con una gran comunidad de desarrolladores que pueden ofrecer asistencia en caso de que surjan problemas. Además, existe una amplia cantidad de recursos y tutoriales disponibles en línea para ayudar a los desarrolladores a aprender y mejorar sus habilidades.
- Seguridad: Java es conocido por ser un lenguaje de programación seguro y fiable, lo que es esencial para las aplicaciones móviles de negocios.