

## CARRERA DE COMPUTACIÓN

### 1. Datos Informativos

- 1.1. **Módulo:**
- 1.2. **Nivel:** 7 Semestre
- 1.3. **Apellidos y Nombres:** Mendez Estiven
- 1.4. **Tema: Métricas de Rendimiento del Software**
- 1.5. **Fecha:** 26/11/2025

### 2. Objetivo

- Comprender los conceptos clave del rendimiento del software.
- Identificar y aplicar métricas de rendimiento comunes.
- Analizar resultados y tomar decisiones basadas en métricas.
- Explorar herramientas de medición del rendimiento.

### 3. Introducción

¿Qué entendemos por rendimiento del software?

- 1.- Capacidad del sistema para ejecutar tareas eficientemente.
- 2.- Velocidad y Tiempo de Respuesta sería lo rápido que el software responde a una acción del usuario.
- 3.- Escalabilidad en la facilidad para mejorar su rendimiento añadiendo más recursos (como servidores).
- 4.- Estabilidad bajo Carga su capacidad para funcionar correctamente cuando muchos usuarios lo usan a la vez.

¿Velocidad siempre significa buen rendimiento?

La velocidad es una parte fundamental del rendimiento, pero el verdadero buen rendimiento es un equilibrio entre:

- **Velocidad** (tiempo de respuesta)
- **Estabilidad** (que esa velocidad se mantenga)
- **Eficiencia** (los recursos que usa para lograrlo)
- **Escalabilidad** (cómo se comporta cuando la demanda aumenta)

## 2. Métricas clave de rendimiento

- Leer y analizar un caso (ej. una web lenta)
- Discusión en grupo: ¿Qué métricas se pueden aplicar al caso?

Diapositivas 4–9

### Posibles Causas y Relación con Métricas

Métrica	Posible Causa en la Web Lenta
Tiempo de respuesta	Código ineficiente, base de datos lenta, muchos recursos (imágenes, JS).
Throughput	Servidor sobrecargado por muchas solicitudes simultáneas.
Uso de recursos	Memoria agotada, CPU al 100%, disco lento.
Latencia	Usuarios lejos del servidor, red con congestión.
Disponibilidad	Caídas frecuentes por falta de escalabilidad.

## 3. Herramientas para medir rendimiento

- Demostración: Uso básico de [Lighthouse de Chrome] o [JMeter] -

Diapositiva 10

- Reto práctico: Medir el tiempo de carga de un sitio web y presentar resultado

Informe desde el 26 nov 2025, 8:00:53 p.m.

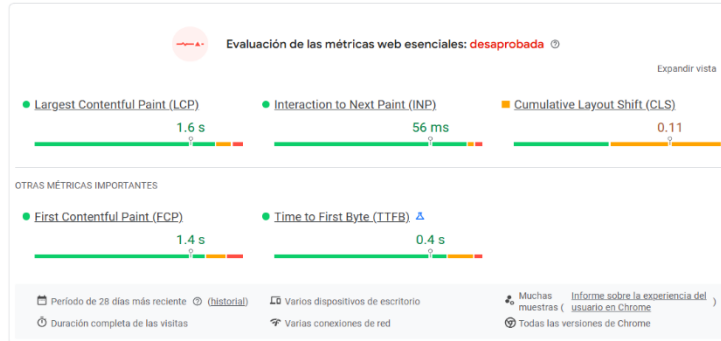
<https://aulas-virtuales.upec.edu.ec/>

Analizar

[Celulares](#) [Escritorio](#)

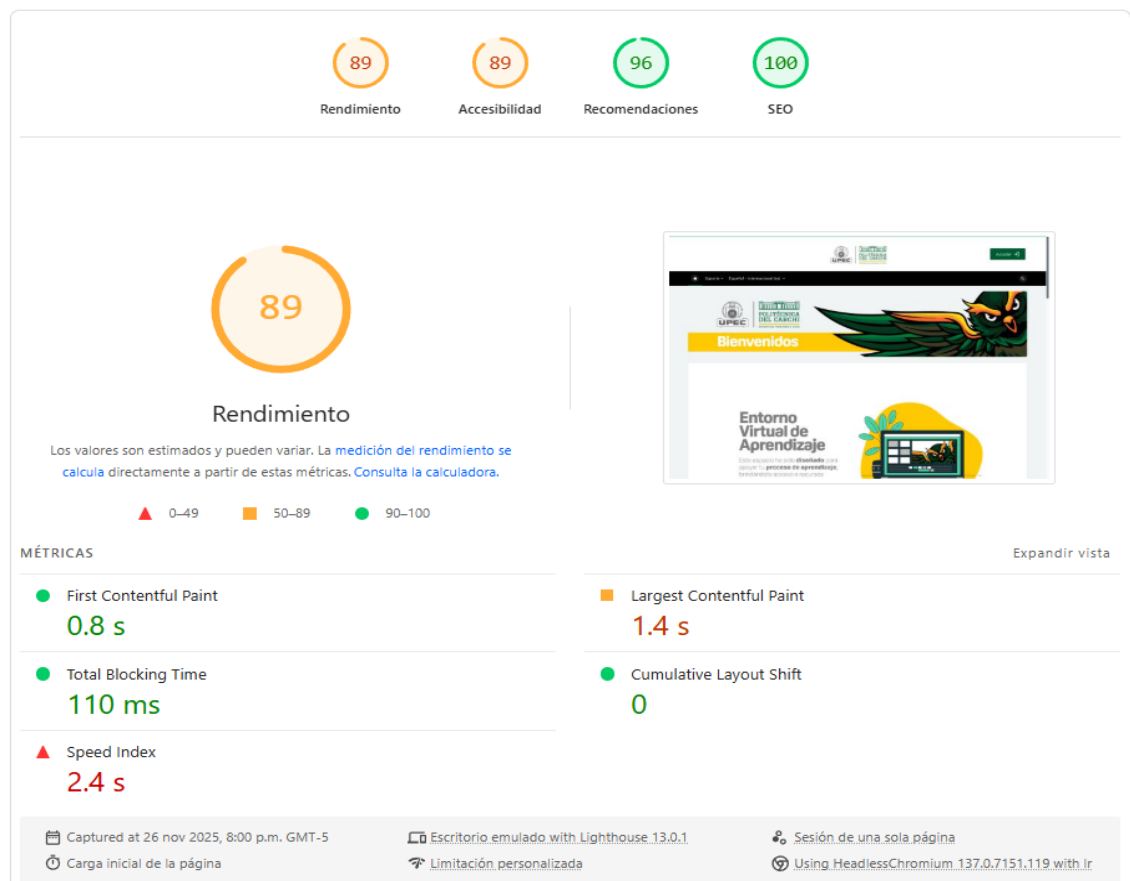
Descubre lo que experimentan tus usuarios reales

[Esta URL](#) [Origen](#)



Diagnostica problemas de rendimiento

Diagnostica problemas de rendimiento



Calle Antisana y Av. Universitaria  
Telf: (06) 2980837 - 2984435  
[info@upec.edu.ec](mailto:info@upec.edu.ec)  
[www.upec.edu.ec](http://www.upec.edu.ec)  
Tulcán - Ecuador

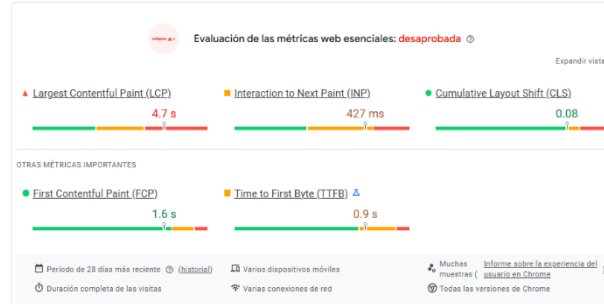
Informe desde el 26 nov 2025, 8:03:30 p.m.

<https://www.instagram.com/>

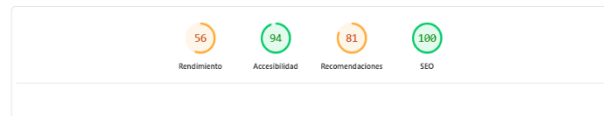
Analizar

[Celulares](#) [Escritorio](#)

Descubre lo que experimentan tus usuarios reales

[Esta URL](#) [Origen](#)

Diagnostica problemas de rendimiento

[Celulares](#)[Escritorio](#)

56

Rendimiento

Los valores son estimados y pueden variar. La [medición del rendimiento se calcula](#) directamente a partir de estas métricas. [Consulta la calculadora.](#)

▲ 0-49 ■ 50-89 ● 90-100

MÉTRICAS

■ First Contentful Paint

2.8 s

■ Total Blocking Time

530 ms

● Speed Index

3.3 s

▲ Largest Contentful Paint

32.4 s

● Cumulative Layout Shift

0.059

Expandir vista

Captured at 26 nov 2025, 8:03 p.m. GMT-5

⌚ Carga inicial de la página

📱 Moto G Power, emulado with Lighthouse 13.0.1

🌐 Limitación de la red: 4G lenta

🔗 Sesión de una sola página

🔗 Using HeadlessChromium 137.0.7151.119 with Jr

[Ver diagrama de árbol](#)Mostrar las auditorías relevantes para la métrica: [All](#) [FCP](#) [LCP](#) [INP](#) [CLS](#)

ESTADÍSTICAS

▲ Mejora la entrega de imágenes — Ahorro estimado de 2,548 KiB

■ JavaScript heredado — Ahorro estimado de 21 KiB

○ Causantes del cambio de diseño

○ Desglose de LCP

#### 4. Buenas prácticas y análisis

- Actividad: Clasifica prácticas como “preventivas” o “reactivas” -

Diapositiva 11

- Breve explicación

##### **Preventivas (P):**

- Pruebas antes del lanzamiento
- Implementar caché
- Revisar código
- Establecer límites de recursos

##### **Reactivas (R):**

- Monitorización con alertas
- Optimizar consultas lentas
- Escalar por alta demanda
- Optimizar imágenes tras reporte

##### **Diferencia clave:**

- **Preventivas** = Antes del problema (evitan)
- **Reactivas** = Después del problema (solucionan)

#### 5. Práctico:

- En grupos, elegir una página o app conocida
- Estimar su rendimiento usando las métricas vistas
- Documentar conclusiones y proponer mejoras

#### **"Recetas Online"**

##### **Problemas principales:**

- Videos muy lentos (6-8 segundos en cargar)
- Alto consumo de datos (150MB por receta)
- App gasta mucha batería
- Caídas frecuentes en sección de videos

### **3 mejoras urgentes:**

1. **Streaming adaptable** - Videos que se ajusten a la conexión
2. **CDN para videos** - Usar red de distribución especializada
3. **Carga bajo demanda** - Videos solo cuando son visibles

### **Resultado esperado:**

- Videos de 8s → 2s
- Consumo de 150MB → 40MB
- App más estable y menos pesada

### **4. Conclusiones**

- Las métricas son esenciales para evaluar objetivamente el rendimiento y tomar decisiones basadas en datos, no en suposiciones.
- El equilibrio es clave: un buen rendimiento balancea velocidad, estabilidad, uso de recursos y escalabilidad.
- La perspectiva del usuario es fundamental: lo que importa es cómo el usuario final experimenta la rapidez y confiabilidad del software.

### **¿Qué métrica podrían utilizar en sus proyectos?**

1. Tiempo de carga - Velocidad que ve el usuario
2. Errores por minuto - Estabilidad del sistema
3. Uso de CPU/Memoria - Eficiencia de recursos
4. Tiempo respuesta APIs - Rendimiento del backend
5. Usuarios simultáneos - Capacidad de escalar

### **¿Cómo se define si un sistema es 'rápido' o eficiente?**

- Responde antes de 3 segundos (web) o 2 segundos (app)
- El usuario no nota esperas

Es "eficiente" si:

- Usa pocos recursos (CPU, memoria, batería)
- Mantiene su velocidad con muchos usuarios