

Innovando Soluciones: Reto en Computación para Ingenieros de Sistemas

Ingeniería | Ingeniería de sistemas | Aprendizaje Basado en Retos

Descripción

Este plan de clase está diseñado para estudiantes universitarios de Ingeniería de Sistemas y tiene como propósito introducirlos en el campo de la Computación mediante un reto real que fomente la creatividad y el pensamiento crítico. Los estudiantes aprenderán a analizar problemas computacionales, diseñar soluciones innovadoras y evaluar su viabilidad técnica en un contexto aplicado. La relevancia de este enfoque radica en conectar los conceptos teóricos con desafíos reales de la ingeniería, preparando a los futuros profesionales para enfrentar situaciones complejas en su desarrollo profesional.

Durante la sesión, los estudiantes trabajarán activamente en equipos para proponer soluciones a un problema concreto relacionado con la optimización de recursos computacionales en una empresa ficticia, lo que los motivará a aplicar sus conocimientos previos en programación, algoritmos y sistemas. Este aprendizaje basado en retos facilita la adquisición de competencias clave como la colaboración, la comunicación efectiva y la resolución de problemas, fundamentales para su vida académica y laboral.

Objetivos de Aprendizaje

- Analizar un problema real de optimización computacional en el contexto de Ingeniería de Sistemas.
- Diseñar una propuesta de solución innovadora utilizando conceptos fundamentales de computación.
- Evaluar la factibilidad técnica y práctica de la solución propuesta en equipo.
- Comunicar de manera clara y efectiva los resultados y justificaciones del proyecto.

Recursos Necesarios

- Computadoras con acceso a internet (1 por cada 2 estudiantes).
- Software básico de edición de texto y presentaciones (Microsoft Word, PowerPoint o equivalente).
- Material impreso con enunciado del reto y pautas de trabajo (1 por grupo).
- Pizarra y marcadores para anotaciones grupales.
- Proyector para presentación inicial y cierre.
- Acceso a repositorios de documentación técnica y lenguajes de programación comunes (Python, Java).

Requisitos Previos

- Conocimientos básicos de programación y estructuras de datos.

- Familiaridad con conceptos de algoritmos y complejidad computacional.
- Experiencia previa en trabajo colaborativo y presentación de resultados.
- Comprensión de sistemas de información y su aplicación en empresas.

Actividades

Fase de Inicio

Tiempo estimado: 10 minutos

Propósito de la sesión

Docente: Explica que la sesión se centrará en aplicar conocimientos de computación para resolver un reto práctico que refleja situaciones reales en empresas de tecnología, destacando la importancia de la innovación y el trabajo en equipo.

Estudiantes: Escuchan atentamente para comprender el objetivo general y la relevancia del reto.

Activación de conocimientos previos

Docente: Plantea la siguiente pregunta detonadora a toda la clase: “¿Qué estrategias conocen para optimizar el rendimiento de sistemas computacionales en una empresa? Mencionen dos ejemplos concretos.”

Estudiantes: Responden en voz alta o escriben brevemente sus ideas, fomentando un intercambio rápido de conocimientos previos.

Motivación y enganche

Docente: Muestra un caso real breve (2 minutos) sobre cómo una empresa mejoró su eficiencia computacional ahorrando costos y tiempo mediante soluciones innovadoras. Luego, plantea el reto del día: “Ustedes deben diseñar una propuesta para optimizar los recursos computacionales de una empresa ficticia que enfrenta problemas de lentitud y altos costos.”

Estudiantes: Se motivan al comprender la conexión con situaciones reales y el impacto potencial de su trabajo.

Contextualización

Docente: Relaciona el reto con la vida cotidiana profesional de los estudiantes, resaltando que enfrentarán problemas similares en su futuro laboral y la importancia de desarrollar habilidades para solucionarlos creativamente.

Estudiantes: Reflexionan sobre la aplicabilidad del aprendizaje y se preparan para iniciar el trabajo en equipo.

Fase de Desarrollo

Tiempo estimado: 40 minutos

Presentación del contenido

Docente: Expone brevemente (5 minutos) los conceptos clave de optimización computacional, resaltando técnicas comunes y herramientas digitales disponibles, evitando una exposición magistral extensa para mantener el enfoque en el reto.

Estudiantes: Toman notas y plantean dudas rápidas para aclarar conceptos esenciales.

Actividad 1: Análisis del problema

- **Objetivo específico:** Analizar el problema real de optimización computacional.
- **Instrucciones:**
 - **Docente:** Divide la clase en grupos de 3-4 estudiantes y entrega el enunciado detallado del reto.
 - Indica que deben identificar y listar los principales problemas que enfrenta la empresa ficticia.
 - Solicita que cada grupo discuta y anote causas y consecuencias en un esquema simple.
- **Organización:** Grupos de 3-4 estudiantes.
- **Producto:** Lista de problemas y esquema de causas-consecuencias.
- **Tiempo estimado:** 15 minutos.
- **Rol del docente:** Circula entre grupos, hace preguntas guía como “¿Qué factores impactan más en la lentitud?”, “¿Cómo afectan estos problemas al negocio?”, y apoya a clarificar dudas.

Actividad 2: Diseño de propuesta de solución

- **Objetivo específico:** Diseñar una propuesta innovadora para optimizar recursos.
- **Instrucciones:**
 - **Docente:** Indica que los grupos deben idear una solución específica basada en técnicas de computación, describiendo brevemente cómo funcionaría y qué beneficios aportaría.
 - Recomienda que consideren aspectos técnicos y prácticos para la implementación.
- **Organización:** Grupos de 3-4 estudiantes.
- **Producto:** Breve propuesta escrita o esquema visual con la descripción de la solución.
- **Tiempo estimado:** 15 minutos.
- **Rol del docente:** Observa la creatividad de las propuestas, formula preguntas como “¿Qué tecnología usarían?”, “¿Cómo medirían el éxito?”, y sugiere mejoras o consideraciones adicionales.

Actividad 3: Evaluación y preparación de presentación

- **Objetivo específico:** Evaluar la factibilidad técnica y preparar la comunicación de resultados.
- **Instrucciones:**
 - **Docente:** Solicita que los grupos identifiquen posibles limitaciones o desafíos de su propuesta y preparen una presentación breve (3 minutos) para explicar su solución y justificación.
 - Indica que deben organizar sus ideas y asignar roles para la exposición.

- **Organización:** Grupos de 3-4 estudiantes.
- **Producto:** Presentación oral breve o exposición ante la clase.
- **Tiempo estimado:** 10 minutos.
- **Rol del docente:** Apoya en la estructuración de ideas, sugiere enfoques claros y prepara a los estudiantes para la retroalimentación.

Diferenciación

- Para estudiantes que terminan antes: Se les invita a investigar tecnologías emergentes relacionadas con optimización computacional y a preparar un breve comentario para enriquecer la discusión final.
- Para estudiantes que requieren apoyo: El docente facilita guías de preguntas específicas para orientar el análisis y diseño, y fomenta el trabajo colaborativo para complementar habilidades.

Transiciones

Docente: Conecta cada actividad recordando que el análisis es la base para diseñar soluciones, y que evaluar fortalezas y limitaciones es clave para comunicar con éxito.

Estudiantes: Vinculan cada paso para construir su aprendizaje progresivamente.

Fase de Cierre

Tiempo estimado: 10 minutos

Síntesis

Docente: Solicita que cada grupo comparta su propuesta en 2 minutos, resaltando el problema identificado, la solución y la evaluación realizada.

Estudiantes: Presentan su trabajo y escuchan a otros grupos, completando un organizador gráfico en la pizarra con las ideas principales.

Reflexión metacognitiva

Docente: Plantea las siguientes preguntas para que los estudiantes respondan individualmente por escrito:

- ¿Qué aprendí sobre el análisis y diseño de soluciones computacionales en este reto?
- ¿Cómo contribuyó mi trabajo en equipo a la solución propuesta?
- ¿Qué aspectos puedo mejorar para futuros retos similares?

Retroalimentación

Docente: Proporciona comentarios inmediatos destacando fortalezas y aspectos a mejorar en las propuestas y presentaciones, motivando la autoevaluación y la colaboración.

Transferencia

Docente: Conecta lo aprendido con posibles proyectos futuros en la carrera, y la importancia de aplicar estas habilidades en contextos profesionales reales.

Tarea o reto

Docente: Propone un reto opcional para profundizar: “Investiga un caso real de optimización computacional en una empresa y prepara un breve informe para discutir en la próxima sesión.”

Evaluación

Tipo de evaluación: Diagnóstica durante la activación de conocimientos en la fase de inicio; formativa durante las actividades de desarrollo mediante observación y retroalimentación continua; sumativa en el cierre con la presentación y reflexión individual.

Criterios de evaluación:

- Análisis adecuado y completo del problema (vinculado al objetivo 1).
- Creatividad y pertinencia en el diseño de la solución (vinculado al objetivo 2).
- Evaluación realista de la factibilidad técnica (vinculado al objetivo 3).
- Claridad y eficacia en la comunicación oral y escrita (vinculado al objetivo 4).

Instrumentos sugeridos: Rúbrica de evaluación para la presentación grupal, lista de cotejo para la participación en análisis y diseño, y autoevaluación escrita para reflexión metacognitiva.

Evidencias de aprendizaje: Producto del esquema de análisis, propuesta de solución escrita, presentación oral y respuestas a preguntas de reflexión.