

Diseño Instruccional - Curso de Algoritmia

2-2024

Cano Beltrán, Jhon Haide¹, Jurado Rosero, Javier¹.

jhon.canob@campusucc.edu.co, javier.jurador@campusucc.edu.co

1. Profesor investigador facultad de ingeniería.

Grupo de Investigación ESLINGA

Universidad Cooperativa de Colombia Campus Cali



PROYECTO DE DISEÑO INSTRUCCIONAL PARA AMBIENTES DIGITALES - Análisis

Fase 1. Análisis y delimitación del contexto de intervención

1.1. Características del contexto

**Descripción del contexto
y características
socioculturales:**

El curso de Algoritmia se impartirá en una universidad que cuenta con una comunidad diversa de estudiantes provenientes de diferentes contextos socioeconómicos y culturales. La universidad tiene un fuerte enfoque en la tecnología y la innovación, lo que permite un ambiente propicio para la enseñanza de temas complejos como la algoritmia. Los estudiantes suelen tener un alto nivel de familiaridad con

	herramientas tecnológicas, lo que facilita la implementación de un curso digital en la plataforma BrightSpace.
--	---

Especificaciones de infraestructura:	La universidad está equipada con infraestructura tecnológica avanzada que soporta la enseñanza en línea a través de BrightSpace. Los recursos disponibles incluyen: • Salones/Aulas: Aulas equipadas con proyectores y acceso a internet de alta velocidad. • Espacios Comunes: Bibliotecas y laboratorios de informática accesibles para los estudiantes. • Herramientas Tecnológicas Disponibles: Computadoras, tablets, y acceso a software especializado en algoritmia. • Redes de Conexión: Wi-Fi de alta velocidad disponible en toda la universidad. • Espacios de Estudio Individual: Cabinas de estudio en la biblioteca y otros espacios dedicados. • Flexibilidad en la Movilidad de los Muebles: Aulas con mobiliario que puede reorganizarse según las necesidades del curso.
--	--

	• Plataformas Tecnológicas: Brightspace como la principal plataforma de gestión del aprendizaje. • Acceso a Redes Sociales: Integración de redes sociales para fomentar la colaboración y el intercambio de ideas. • Velocidad Digital: Alta capacidad de banda ancha para soportar actividades en línea sin interrupciones.
--	---

Especificaciones sobre los recursos humanos:	Docentes: Profesores con experiencia en algoritmia y programación. Expertos en el Tema: Profesionales con experiencia en la industria de la tecnología y la computación. Técnicos: Personal técnico para soporte en el uso de Brightspace y otras herramientas digitales. Coordinadores: Personal encargado de la planificación y seguimiento del curso. Diseñadores Gráficos: Para la creación de materiales visuales atractivos. Programadores: Apoyo para el desarrollo de ejemplos prácticos y aplicaciones interactivas.
Presupuesto para el proyecto:	Recursos Humanos: \$6,000 USD (honorarios docentes, diseñadores gráficos, programadores). Materiales y Equipos: \$2,000 USD (software adicional, licencias). Infraestructura: \$1,000 USD (mejoras en el entorno virtual). Formación: \$1,000 USD (capacitación para el personal en el uso de Brightspace).

Perfil del experto en el tema:	Experiencia Profesional: Mínimo 5 años de experiencia en programación y diseño de algoritmos. Experiencia Académica: Grado de maestría o doctorado en Ciencias de la Computación o afines, experiencia en enseñanza universitaria. Características Profesionales y de Personalidad: Habilidades de comunicación, capacidad para explicar conceptos complejos de manera clara, enfoque en la resolución de problemas, adaptabilidad y disposición para colaborar en entornos digitales.
---------------------------------------	--

Perfil de participantes:	Número de Participantes: 30 estudiantes. Edad: Entre 20 y 25 años. Nivel de Escolaridad: Estudiantes de tercer o cuarto semestre de la carrera de Ingeniería en Computación o afines. Características Socioeconómicas: Varían desde estudiantes con acceso a recursos tecnológicos en casa hasta aquellos que dependen de los recursos de la universidad. Características Generales: Interés común en el desarrollo de software, experiencia previa básica en programación, alta motivación para aprender algoritmos complejos.
---------------------------------	---

1.2. Ideas generales para el proceso de enseñanza – aprendizaje

Ideación de contenido	Conocimientos Declarativos: Conceptos fundamentales de algoritmia, estructuras de datos, análisis de complejidad, paradigmas de diseño de algoritmos. Conocimientos Procedimentales: Implementación de algoritmos en lenguajes de programación como Python o C++, resolución de problemas mediante algoritmos, optimización de código.
-----------------------	--

	Conocimientos Actitudinales: Fomento del pensamiento crítico, trabajo en equipo, persistencia en la resolución de problemas.
--	--

Especificación del tipo de acción formativa que realizarás y el tiempo de duración.	Tipo de Acción Formativa: • Cursos: Curso principal de Algoritmia con una duración de 16 semanas. • Talleres: Talleres complementarios de 2 semanas para la práctica intensiva de algoritmos específicos. • Seminarios: Seminarios de expertos invitados sobre aplicaciones avanzadas de algoritmos. • Videos Tutoriales: Series de videos explicativos sobre conceptos clave, disponibles durante todo el curso. Duración Total del Proyecto: 16 semanas.
---	---

Alcances de la acción formativa:	Memorizar: Los principios fundamentales de algoritmia y estructuras de datos. Aplicar: Diseñar e implementar algoritmos eficientes para resolver problemas complejos. Comprender: Analizar y optimizar algoritmos existentes, así como desarrollar soluciones innovadoras en programación.
--	--

PROYECTO DE DISEÑO INSTRUCCIONAL PARA AMBIENTES DIGITALES – Diseño

Descripción general	<i>El proyecto de diseño instruccional para el curso de Algoritmia está enfocado en estudiantes de tercer o cuarto semestre de la carrera de Ingeniería en Computación o afines, en una universidad que utiliza la plataforma Brightspace. El curso tiene como objetivo principal enseñar los conceptos y técnicas fundamentales en algoritmia, proporcionando a los estudiantes las habilidades necesarias para diseñar, implementar y analizar algoritmos eficientes. Se llevará a cabo en un entorno digital, con una combinación de clases en línea, talleres prácticos y actividades interactivas.</i>
Modelo	<i>El modelo adoptado para este curso es el Modelo Centrado en el Alumno con un enfoque constructivista, el cual permite a los estudiantes ser el centro del proceso de aprendizaje, promoviendo la autogestión y el aprendizaje activo. Este modelo es adecuado para ambientes digitales, ya que facilita la personalización del aprendizaje y la integración de múltiples recursos tecnológicos, fomentando la interacción y la colaboración en línea.</i>
Justificación	<i>La elección del modelo centrado en el alumno se justifica por la naturaleza interactiva y flexible de los ambientes digitales. BrightSpace, como plataforma de gestión del aprendizaje, ofrece herramientas que soportan este modelo, permitiendo a los estudiantes acceder a recursos, participar en foros de discusión, realizar autoevaluaciones y recibir retroalimentación inmediata. La modalidad educativa en línea es complementada con aprendizaje basado en retos, lo que alinea el curso con las demandas actuales del mercado laboral y las expectativas de los estudiantes en cuanto a la adquisición de competencias prácticas.</i>

Objetivos, lecciones y temas	Memorizar los principios fundamentales de algoritmia y estructuras de datos. Aplicar conceptos de algoritmia en la resolución de problemas complejos mediante la implementación de algoritmos en Python. Comprender el análisis de complejidad de algoritmos y optimización de soluciones. • Lección 1: Introducción a la Algoritmia ○ Tema 1.1: Conceptos Básicos de Algoritmia ○ Tema 1.2: Estructuras de Datos Fundamentales • Lección 2: Complejidad Computacional ○ Tema 2.1: Análisis de Algoritmos ○ Tema 2.2: Notación Big O • Lección 3: Diseño de Algoritmos ○ Tema 3.1: Algoritmos de Divide y Vencerás ○ Tema 3.2: Programación Dinámica • Lección 4: Algoritmos de Búsqueda y Ordenación ○ Tema 4.1: Búsqueda Binaria ○ Tema 4.2: Ordenamiento por Mergesort y Quicksort • Lección 5: Grafos y Algoritmos Relacionados ○ Tema 5.1: Representación de Grafos ○ Tema 5.2: Algoritmo de Dijkstra
Estrategias y técnicas didácticas	Aprendizaje Activo: Los estudiantes participan activamente en el proceso de aprendizaje a través de actividades prácticas y debates en línea. Aprendizaje Basado en Retos: Se presentan desafíos reales que los estudiantes deben resolver aplicando los conocimientos adquiridos. Aprendizaje Flexible: Se ofrece la posibilidad de acceder a los contenidos en cualquier momento y desde cualquier lugar, adaptándose a los ritmos individuales de aprendizaje. Gráficos y Mapas Conceptuales: Para visualizar la estructura de datos y los pasos de los algoritmos.

	Debates en Foros: Para discutir enfoques y optimizaciones de algoritmos. Autoevaluaciones: Para que los estudiantes puedan medir su comprensión de los temas antes de pasar a la siguiente lección.
Actividades	Prácticas de Programación: Ejercicios semanales donde los estudiantes implementan algoritmos en Python. Proyectos Colaborativos: Grupos de estudiantes trabajan en resolver un problema complejo, aplicando múltiples algoritmos y presentando sus resultados. Debates en Línea: Discusiones sobre los beneficios y desventajas de diferentes enfoques algorítmicos. Exámenes en Línea: Pruebas que evalúan la comprensión de conceptos clave y la capacidad de aplicarlos en la práctica.
Recursos	Textos: Artículos y capítulos de libros sobre algoritmia. Enlaces: Acceso a sitios web y bases de datos académicas con recursos adicionales. Videos: Tutoriales en video que explican paso a paso la implementación de algoritmos. Simuladores en Línea: Herramientas que permiten visualizar el funcionamiento de algoritmos. Foros de Discusión: Espacios en Brightspace para interactuar con otros estudiantes y profesores.
Calendarización	Publicación de las Lecciones: Se publicarán semanalmente, cada lunes. Períodos para la Realización de Actividades: • Prácticas de Programación: Del martes al domingo de cada semana. • Proyectos Colaborativos: Dos semanas de duración, comenzando en la semana 7. • Debates en Línea: Disponibles durante toda la semana correspondiente a cada lección. Plazos de Evaluación: • Recepción de Tareas: Cada domingo hasta las 23:59.

	• Períodos de Evaluación y Retroalimentación: Los lunes y martes de cada semana.
Estrategia de evaluación	El aprendizaje se valorará mediante: • Evaluaciones Formativas: Pruebas en línea al final de cada lección para medir la comprensión de conceptos. • Evaluaciones Sumativas: Examen final que cubre todo el contenido del curso, enfocándose en la aplicación práctica de los conocimientos. • Evaluación de Proyectos: Los proyectos colaborativos serán evaluados en función de la creatividad, eficiencia y justificación de los algoritmos implementados. • Retroalimentación Continua: Los estudiantes recibirán retroalimentación inmediata sobre sus autoevaluaciones y tareas para mejorar continuamente.

PROYECTO DE DISEÑO INSTRUCCIONAL PARA AMBIENTES DIGITALES – Desarrollo

Identificación de elementos y herramientas tecnológicas de acuerdo al modelo pedagógico	Modelo Centrado en el Alumno: El curso de Algoritmia emplea un modelo centrado en el alumno con un enfoque constructivista, utilizando la plataforma Brightspace como el entorno principal de aprendizaje. Este modelo promueve la

	interacción activa, la colaboración y el aprendizaje autónomo, apoyado por diversas herramientas tecnológicas.
Multimedia	Para facilitar la comprensión y el aprendizaje de los estudiantes, se utilizarán los siguientes elementos multimedia: • Videos: Explicaciones detalladas de los conceptos de algoritmia y ejemplos de implementación en Python y C++. • Imágenes Fijas y Animadas: Diagramas de flujo, gráficos y animaciones que muestran la ejecución paso a paso de algoritmos. • Infografías: Resúmenes visuales que destacan los puntos clave de cada lección. • Sonidos: Breves efectos sonoros para mantener la atención y guiar a los estudiantes a través de actividades interactivas. • Texto Interactivo: Contenidos que permiten la interacción directa del alumno, como la manipulación de variables en algoritmos.
Software	Para el desarrollo de la actividad formativa, se utilizarán las siguientes herramientas de software: • Correo Electrónico: Para la comunicación formal entre estudiantes y profesores. • Sistemas de Chat: Herramientas como Microsoft Teams o Slack para la colaboración en tiempo real. • Sistemas de Foro: Foros de discusión dentro de Brightspace para debates asincrónicos sobre temas específicos de algoritmia.
Hardware	El hardware requerido incluye: • Equipo para Videoconferencias: Incluye líneas telefónicas, internet de alta velocidad, y diademas con micrófono y audífonos para sesiones en vivo.

	• Servidores: Soporte de la plataforma Brightspace y almacenamiento de datos de los estudiantes. • Equipo de Telecomunicaciones: Para asegurar una conectividad estable durante las sesiones en vivo y actividades en línea.
Recursos humanos (técnica y operativa)	Los recursos humanos necesarios para el desarrollo y la impartición del curso son: • Tutores: Profesores con experiencia en algoritmia que guíen a los estudiantes a lo largo del curso. • Facilitadores: Ayudantes que asisten en la organización y resolución de dudas técnicas y administrativas. • Administradores de Plataforma: Personal encargado del mantenimiento y soporte técnico de Brightspace. • Equipo de Soporte Técnico: Especialistas que resuelven problemas técnicos relacionados con la plataforma y el hardware.
Determinación de uso de tendencias educativas	Uso de Aprendizaje Basado en Retos (ABR): Se empleará el uso del Aprendizaje Basado en Retos (ABR) para el desarrollo de proyectos colaborativos que involucren la resolución de problemas complejos utilizando algoritmos. Con esta tendencia, se pretende fomentar el pensamiento crítico, la creatividad y la aplicación práctica de los conceptos teóricos aprendidos en el curso.
¿En dónde se aplicará el uso de la tendencia educativa elegida?	Aplicación del ABR en el Curso: El ABR se implementará como una estrategia central en los proyectos colaborativos de la Lección 4, donde los estudiantes deberán trabajar en equipos para resolver problemas reales utilizando algoritmos avanzados. Este enfoque permitirá a los estudiantes aplicar directamente lo que han aprendido en un contexto práctico, desarrollando así habilidades de resolución de problemas y trabajo en equipo.

Describe el desarrollo de la aplicación de dicha tendencia.	Actividad 1: Introducción y Revisión de Conceptos Básicos • Descripción: Los estudiantes revisarán los conceptos básicos de algoritmia y estructuras de datos. Actividad 2: Elaboración de Mapa Conceptual • Descripción: Creación de un mapa conceptual que ilustre las relaciones entre diferentes estructuras de datos y algoritmos. Actividad 3: Implementación de Algoritmos en Lenguajes de Programación • Descripción: Los estudiantes implementarán varios algoritmos en Python o C++ y analizarán su eficiencia. Actividad 4: Construcción de una Historia Utilizando Storytelling • Descripción: Los estudiantes desarrollarán una narrativa que explique la aplicación de un algoritmo en un problema del mundo real. Actividad 5: Resolución de Problemas Basados en Retos • Descripción: Trabajo en equipo para resolver problemas complejos que requieren el diseño e implementación de algoritmos avanzados. Actividad 6: Discusión e Intercambio de Ideas en Foro • Descripción: Participación en foros para discutir enfoques y soluciones a problemas algorítmicos presentados en clase. Actividad 7: Evaluación Final • Descripción: Los estudiantes deberán completar un examen que abarca todos los conceptos y habilidades aprendidos durante el curso.
	Actividad 1: Introducción y Revisión de Conceptos Básicos

Definición y uso de herramientas tecnológicas en cada una de las actividades planeadas.	• Descripción: Los estudiantes revisarán los conceptos básicos de algoritmia y estructuras de datos. • Herramientas Tecnológicas: ◦ Videos Interactivos (Brightspace): Videos explicativos con preguntas de control integradas para asegurar la comprensión. ◦ Quizzes en Línea (Brightspace): Cuestionarios automáticos para evaluar la comprensión inicial. Actividad 2: Elaboración de Mapa Conceptual • Descripción: Creación de un mapa conceptual que ilustre las relaciones entre diferentes estructuras de datos y algoritmos. • Herramientas Tecnológicas: ◦ MindMeister/CmapTools/Gliffy: Herramientas para la creación de mapas conceptuales interactivos. Actividad 3: Implementación de Algoritmos en Lenguajes de Programación • Descripción: Los estudiantes implementarán varios algoritmos en Python o C++ y analizarán su eficiencia. • Herramientas Tecnológicas: ◦ Jupyter Notebooks (Python) / IDEs (C++): Entornos de desarrollo interactivos para la programación y ejecución de algoritmos. ◦ GitHub/Classroom (para trabajo colaborativo): Repositorios donde los estudiantes pueden compartir su código y colaborar en proyectos. Actividad 4: Construcción de una Historia Utilizando Storytelling
---	--

- **Descripción:** Los estudiantes desarrollarán una narrativa que explique la aplicación de un algoritmo en un problema del mundo real.
- **Herramientas Tecnológicas:**
 - **PowToon/Storybird/Storify:** Herramientas para la creación de videos o presentaciones animadas con enfoque en storytelling.

Actividad 5: Resolución de Problemas Basados en Retos

- **Descripción:** Trabajo en equipo para resolver problemas complejos que requieren el diseño e implementación de algoritmos avanzados.
- **Herramientas Tecnológicas:**
 - **Microsoft Teams/Slack (para comunicación):** Plataformas de chat y colaboración para la coordinación del equipo.
 - **Google Docs/Overleaf (para documentación):** Herramientas para la co-escritura de la solución y documentación técnica del proyecto.
 - **Brightspace (para entrega y seguimiento):** Entrega de proyectos y seguimiento de progreso a través de la plataforma LMS.

Actividad 6: Discusión e Intercambio de Ideas en Foro

- **Descripción:** Participación en foros para discutir enfoques y soluciones a problemas algorítmicos presentados en clase.
- **Herramientas Tecnológicas:**
 - **Foros Asincrónicos en Brightspace:** Espacios donde los estudiantes pueden publicar y discutir sus ideas.
 - **Zoom/Google Hangouts (para debates sincrónicos):** Herramientas

	para reuniones en tiempo real donde se discuten las soluciones y estrategias adoptadas. Actividad 7: Evaluación Final • Descripción: Los estudiantes deberán completar un examen que abarca todos los conceptos y habilidades aprendidos durante el curso. • Herramientas Tecnológicas: ◦ Exámenes en Línea (Brightspace): Examen con preguntas de opción múltiple, de desarrollo y ejercicios de programación que se evaluarán automáticamente y manualmente.
Identificación de las actividades en las cuales se aplicará la incorporación de herramientas tecnológicas para su desarrollo.	Actividad 1: Elaboración de Mapa Conceptual • Herramientas: MindMeister, CmapTools, Gliffy Actividad 2: Construcción de una Historia Utilizando Storytelling • Herramientas: PowToon, Storybird, Storify Actividad 3: Discusión e Intercambio de Ideas en Foro • Herramientas: Foros asincrónicos en Brightspace, foros sincrónicos en ZOOM o Google Hangouts.
Selección y definición de la herramienta tecnológica adecuada para cada actividad.	Actividad 1: Elaboración de Mapa Conceptual • Herramientas: Los estudiantes podrán utilizar MindMeister para crear mapas conceptuales de estructuras de datos y algoritmos, lo que facilitará la visualización de conexiones entre conceptos clave. Actividad 2: Construcción de una Historia Utilizando Storytelling

- | | |
|--|--|
| | <ul style="list-style-type: none">• Herramientas: Se utilizará PowToon para que los estudiantes creen historias animadas que expliquen la aplicación de algoritmos en la vida real, ayudando a desarrollar habilidades narrativas y comprensión profunda de los conceptos. |
|--|--|

Actividad 3: Discusión e Intercambio de Ideas en Foro

- | | |
|--|---|
| | <ul style="list-style-type: none">• Herramientas: Los foros de Brightspace permitirán a los estudiantes participar en discusiones asincrónicas, mientras que ZOOM o Google Hangouts se utilizarán para debates en tiempo real, facilitando el intercambio de ideas y opiniones. |
|--|---|