

# Descriptivo del Proyecto

14/05/2024

## Linear Leap

### 1. Descripción General del Proyecto

"Linear Leap" es una aplicación web educativa diseñada para enseñar y evaluar conceptos de matemáticas lineales a través de una serie de módulos interactivos. La aplicación consta de ocho pantallas principales: una de bienvenida, cinco módulos de desarrollo de contenido teórico con ejercicios prácticos, un módulo evaluativo, y un módulo final de curiosidades teóricas acerca del tema.

### 2. Tecnologías Utilizadas

- Svelte:** Utilizado en cada fase del desarrollo para crear interfaces de usuario reactivas. Svelte es un compilador que genera código JavaScript eficiente para actualizar el DOM de manera reactiva.
- Node.js y npm:** Se usaron para gestionar las dependencias del proyecto y ejecutar Svelte.
- GitHub:** Almacenamiento del código fuente y control de versiones.
- Vercel:** Plataforma de despliegue que permite publicar la aplicación web de manera accesible públicamente.

### 3. Desarrollo de la Aplicación

#### Estructura de la Aplicación:

# Página De Bienvenida:

Introduce a los usuarios al curso y les permite comenzar.



Bienvenido a Linear Leap

Salta al futuro de las matemáticas profundadas

Comenzar

Ver video

Teoría

Este resolución implica deducir una expresión cerrada que pueda describir la secuencia completa, permitiendo calcular cualquier término directamente sin necesidad de calcular todos los anteriores, al ser deducida se debe comprobar con el método de inducción para saber si en efecto cumple la propiedad para todo el universo de esta recurrencia con el valor inicial específico.

Ejercicios de Recurrencias parte 1

RELACIONES DE RECURRENCIA

EJERCICIOS RESUELTOS

Ver en YouTube

Este video nos ilustra gráficamente la idea de una resolución analítica y su comprobación por el método de la inducción completa

Página Siguiente

# Módulos de contenido teórico:

Cada uno de los módulos desarrolla una sección del tema para abordar temáticamente todo el contenido, cuentan con micro-evaluaciones con un mínimo requerido de puntos para avanzar entre módulos

# Modulo Evaluativo:

Contiene ejercicios prácticos e interactivos, el usuario debe conseguir la puntuación mínima requerida para pasar

Ejercicio 1

veras 5 recurrencias en un orden aleatorio, la idea es que las ordenes en la pila de contenidos abajo, en orden ascendente, de manera que la recurrencia que se acerque en el menor número de iteraciones al número 100 este de primera y luego las que le siguen

$a_n = -1.5 * a_{(n-1)} + 2$

$a_n = 0.5 * a_{(n-1)}$

$a_n = 5 * a_{(n-1)} + 3$

$a_n = 2 * a_{(n-1)}$

$a_n = 4 * a_{(n-1)} - 5$

Usa esta guía de acceso rápido en caso que quieras avanzar entre módulos para buscar información específica

Modulo 1. Introducción

Modulo 2. Homogeneidad

Modulo 3. Solución c.a e inducción

Anterior

$$F_0 = 0 \text{ y } F_1 = 1$$

Cada término de la secuencia es la suma de los dos términos anteriores. Esta simple regla produce una secuencia de números que aparece sorprendentemente en numerosos contextos dentro y fuera de las matemáticas.

Incluimos un resumen oficial de BBC mundo muy interesante que nos hace un recorrido histórico



# Modulo Curiosidades:

ofrece información adicional y datos interesantes sobre las recurrencias lineales

## Implementación con Svelte:

- Componentes:** Cada pantalla y sección significativa de la aplicación está construida como un componente Svelte, lo que facilita la reusabilidad y mantenimiento del código.
- Reactividad:** Se aprovechó el sistema reactivo de Svelte para actualizar la interfaz de usuario en respuesta a las interacciones del mismo, como completar ejercicios o navegar entre módulos.
- Transiciones y Animaciones:** Se utilizaron las capacidades de Svelte para agregar animaciones que mejoran la experiencia del usuario, como transiciones suaves entre pantallas y animaciones.

## 4. Gestión de dependencias y configuración:

### • Requisitos Previos

- Node.js:** Svelte requiere Node.js para ejecutar el entorno de desarrollo y construir la aplicación. Asegúrate de tener Node.js instalado en tu máquina. Se puede descargar e instalar desde [nodejs.org](https://nodejs.org).
- npm (Node Package Manager):** Viene incluido con Node.js y se usa para administrar las bibliotecas y herramientas necesarias para tus proyectos en Node.js.
- Una vez instalados Node.js y npm, se procede con la creación de una nueva aplicación Svelte.
- Se instalaron librerías y herramientas a través de npm, incluyendo el propio Svelte. Se minimizó la descarga a la mínima requerida, que era la propia configuración svelte, esto permitiría usar librerías base de los componentes nativos de svelte, minimizando los requerimientos de desarrollo.
- Se configuraron scripts npm para facilitar el desarrollo local y la construcción de la versión de producción de la aplicación. Estos estarán incluidos en los archivos +page.svelte, que permite incluir <script> para todo lo que tiene que ver con la gestión de funciones e información “backend” del servicio, <style> que maneja el diseño de la página, y <body><div> que es el manejo del contenido de la página “frontend”.

## Principales Librerías y Herramientas Instaladas

## 1. Svelte:

- **svelte**: El paquete principal de Svelte utilizado para construir interfaces de usuario reactivas (instrucciones incluidas en el README del repositorio).

## 2. Transiciones y Animaciones:

- **svelte/transition**: Módulo de Svelte para implementar transiciones y animaciones dentro de los componentes Svelte. No requiere instalación adicional ya que viene incluido con Svelte.

## 5. Despliegue:

- **GitHub**: Todo el código fuente se gestionó utilizando repositorios en GitHub, lo que también facilitó la colaboración y el seguimiento de cambios.
- **Vercel**: La integración continua desde GitHub a Vercel permitió despliegues automáticos tras cada cambio significativo en el repositorio, asegurando que la versión más reciente siempre esté disponible en línea.

**URL del Proyecto:** La aplicación está disponible públicamente en <https://mate-dis-proyecto-oficial-7ktg.vercel.app>.

## 6. Modelo Didáctico Utilizado, Retos y Aprendizajes

El modelo didáctico empleado se baso en estrategias de aprendizaje continuo, donde el usuario se viera en la obligación de avanzar con el contenido, con test individuales en el recorrido para comprobar el desarrollo de la curva de aprendizaje. Primero se dio una introducción del contenido, definiciones y operaciones básicas, luego se desarrolló los tipos de recurrencias lineales de primer orden existentes, a su vez, se abarcó el desarrollo gráfico de fórmulas que abarcaban la recurrencia, comprobadas por el método de inducción, también se desarrolló la formulación analítica con diferentes técnicas y métodos para modelar recurrencias no homogéneas. Se garantiza que el estudiante comprenda con el paso de cada modulo, la comprensión óptima de los conocimientos ofrecidos.

Uno de los desafíos presentados fue lograr la integración de la interacción de los datos ingresados por el usuario con las funciones procesadoras y la impresión de la salida procesada en pantalla, para ello se debió manejar teoría avanzada de manejo de información desde la interfaz de usuario a la base de datos. El recorrido de un desarrollo común de manera completa.

Lecciones aprendidas sobre desarrollo de software, gestión de proyectos tecnológicos y aprendizaje sobre las recurrencias lineales de primer orden.

# Linear Leap

Salta al futuro de las matemáticas lineales

Julio Cesar Vasquez Garcia

27.777.893

Universidad de los Andes