



Instituto Tecnológico de Costa Rica

Escuela de Computación

IC6600 - Investigacion De Operaciones GR 60

Profesor: Jean Carlos Miranda Fajardo

## **Proyecto Parte #4 - Problemas de Cambio de Equipo**

Estudiantes:

Ricardo De Jesus Soto Araya - 2020035358

Brandon Guillermo Redondo Jimenez- 2019156567

Francisco González Madrigal - 2018107608

II Semestre - Sede de Limón

31 de octubre de 2022

# Índice

<b>Manual de usuario</b>	<b>3</b>
<b>Instrucciones de compilación.</b>	<b>3</b>
<b>Instrucciones de instalación</b>	<b>3</b>
<b>Instrucciones de ejecución</b>	<b>3</b>
<b>Uso de la aplicación: Abrir módulo de Problemas de Mochila</b>	<b>4</b>
<b>Descripción del problema</b>	<b>7</b>
<b>Solución del problema</b>	<b>8</b>

# Manual de usuario

## Instrucciones de compilación.

Requisitos previos:

- Tener instalado NodeJS. Node.js (nodejs.org)
- Tener instalado SweetAlert2 SweetAlert

## Instrucciones de instalación

- 1- Se descomprime el archivo .ZIP
- 2- Se abre una consola en la dirección de la carpeta descomprimida
- 3- Se escribe npm install para instalar los módulos necesarios de React (Se instalará sweetalert si no está previamente instalado.)

## Instrucciones de ejecución

- 1- En consola desde la ubicación del proyecto -> npm start
- 2- Se procederá a mostrar la aplicación el navegador web predefinido automáticamente.

## Uso de la aplicación: Abrir módulo de Cambio de Equipos



1- Se procede a seleccionar el botón de “Reemplazo de equipos”

Reemplazo de equipos

Ingrese el costo inicial del equipo

Seleccione la vida util del proyecto

Seleccione el plazo del proyecto

	Precio de reventa	Costo del mantenimiento	Ganancia recibida por año
Año 1	0	0	0

2- Se mostrará toda la información que se puede ingresar en la aplicación.

## Reemplazo de equipos

**Ingrese el costo inicial del equipo**

500

**Seleccione la vida útil del proyecto**

3

**Seleccione el plazo del proyecto**

5



3- Se procede a llenar la información con el ejemplo base de la bicicleta visto en clase.

	Precio de reventa	Costo del mantenimiento	Ganancia recibida por año
Año 1	400	30	0
Año 2	300	40	0
Año 3	250	60	0

Generar Algoritmo

4-Se llenan los espacios con la información necesaria (Véase que en este ejemplo solo se calculan costos, por lo cual no se tomará en cuenta la ganancia percibida por el bien/plan anual)

	Precio de reventa	Costo del mantenimiento	Ganancia recibida por año
Año 1	400	30	0
Año 2	300	40	0
Año 3	250	60	0

Generar Algoritmo

5- Se procede a realizar click en el botón de generar algoritmo para mostrar la solución respectiva.

## Plan de reemplazo:

T	G(T)	Proximo
0	640	1,3,
1	510	2,4,
2	380	5,
3	260	4,
4	130	5,
5	0	



6- Se procede a mostrar el cálculo del algoritmo realizado, primero se mostrarán los planes de reemplazo y posteriormente los planes óptimos.

## Planes óptimos:

Si se desea iniciar en el tiempo 0

Plan óptimo # 1

0→1→2→5

Plan óptimo # 2

0→1→4→5

Plan óptimo # 3

0→3→4→5

Si se desea iniciar en el tiempo 1

Plan óptimo # 1

1→2→5

Plan óptimo # 2

1→4→5

7- Se muestran los posibles planes óptimos además de las múltiples posibilidades de inicios a lo largo de los años, empezando por el primer año 1/2.

<p><b>Si se desea iniciar en el tiempo 2</b></p> <p><b>Plan óptimo # 1</b></p> <p>2→5</p> <hr/>
<p><b>Si se desea iniciar en el tiempo 3</b></p> <p><b>Plan óptimo # 1</b></p> <p>3→4→5</p> <hr/>
<p><b>Si se desea iniciar en el tiempo 4</b></p> <p><b>Plan óptimo # 1</b></p> <p>4→5</p> <hr/>
<p><b>Si se desea iniciar en el tiempo 5</b></p> <p><b>Plan óptimo # 1</b></p> <p>5</p>

7- Se muestran los posibles planes óptimos además de las múltiples posibilidades de inicios a lo largo de los años, empezando por el primer año 1/2.

## Descripción del problema

Se requiere crear una función dentro de la aplicación creada anteriormente en la tercera etapa del proyecto la cual es capaz de calcular la solución más óptima de selección en respecto a cambio de equipo y/o planes de un proyecto, etc por su respectivo valor, costo, ganancia y subida/bajada de costo através de los años, por lo cual hay que programar en Javascript el código que servirá para solucionar el algoritmo anteriormente mencionado y así mostrarlo a su vez por múltiples componentes hechos en la librería de React.

## Solución del problema

La solución realizada fue la utilización de múltiples componentes de React tanto funcionales como nativos de ES6, esto de forma que los nativos de ES6 son utilizados para realizar los cálculos del algoritmo, a su vez, los componentes funcionales fueron utilizados como componentes que serán reutilizables a lo largo de futuros cambios de la aplicación para así optimizar el tiempo de desarrollo y brindar más atención a la solución de los próximos algoritmos en lugar de programar como mostrarlos.

Con respecto al código utilizado para solucionar el algoritmo, lo que se procedió fue realizar un plan de exactamente qué procesos se realizan al elaborarlo a mano, después se procedió a realizar la codificación del mismo y finalmente la implementación del mismo en la parte visual de la aplicación para así llegar a la solución deseada y poderla mostrar al usuario por medio de los componentes anteriormente mencionados.



## Análisis de resultados

<b>Requerimiento</b>	<b>Estado</b>	<b>Observaciones</b>
Ingresar Costo	Completado	Funciona correctamente en cada una de sus funcionalidades.
Ingresar vida útil	Completado	Funciona correctamente en cada una de sus funcionalidades.
Ingresar plazo del proyectos o equipos	Completado	Funciona correctamente en cada una de sus funcionalidades.
Ingresar el precio de reventa	Completado	Funciona correctamente en cada una de sus funcionalidades.
Ingresar costo de mantenimiento	Completado	Funciona correctamente en cada una de sus funcionalidades.
Ingresar ganacia percibida por el plan/equipo	Completado	Funciona correctamente en cada una de sus funcionalidades.
Mostrar plan de reemplazo	Completado	Funciona correctamente en cada una de sus funcionalidades.
Mostrar los múltiples planes óptimos	Completado	Funciona correctamente en cada una de sus funcionalidades.