



**Universidad<sup>®</sup>  
de Medellín**  
Ciencia y Libertad

## **HISTORIA CLÍNICA**

**Santiago Arango**

**Diego Murillo**

**Profesor: Edison Gutierréz**

**Lenguajes de programación y código limpio**

**Antioquia/Medellín**

**18/04/2023**

## **Descripción del problema**

Se desea implementar una herramienta digital que integre la historia clínica de todos los colombianos, con el fin de facilitar el control y seguimiento que realizan las EPS, las IPS y todos los organismos de control.

Usted ha sido seleccionado para desarrollar la aplicación que permita cumplir con este objetivo. Para esto, por cada una de las personas pertenecientes al sistema de salud se solicitará la siguiente información:

- Identificación
- Nombres
- Apellidos
- Fecha de nacimiento
- Tipo de régimen: Contributivo o subsidiado
- Semanas cotizadas en el sistema de salud
- Fecha de ingreso al sistema de salud
- Fecha de ingreso a la actual EPS
- EPS (Sura, Nueva EPS, Salud Total, Sanitas, Savia)
- Historia clínica
- Cantidad de enfermedades que padece
- Enfermedad más relevante
- Tipo de afiliación: Cotizante o beneficiario
- Costos en tratamientos

Una de las funcionalidades que debe tener la aplicación es permitir que una persona cambie de EPS, para eso se ingresará el número de identificación y antes de proceder con el cambio se debe validar que la persona lleve más de 3 meses en la EPS que va a cambiar.

A partir de estos datos se desea conocer:

- Porcentaje de costos por EPS
- Total de costos por EPS
- Porcentaje de pacientes sin enfermedades
- Identificación y nombre completo del paciente con mayor costo en los tratamientos
- Porcentaje de pacientes por rango edad (Niños: 0 – 12 Adolescente: 12 – 18 Joven: 18 – 30 Adulto: 30 – 55 Adulto Mayor: 55 – 75 Anciano: Mayor de 75)
- Porcentaje de pacientes por régimen
- Porcentaje de paciente por tipo de afiliación
- Total de pacientes que la enfermedad más relevante sea el cáncer.

## Requerimientos Funcionales App Historia Clínica

<b>Nombre</b>	R01 - Generar paciente
<b>Resumen</b>	La aplicación va a tener una función que le permita al usuario generar un paciente
<b>Entrada</b>	identificación, nombre, apellidos, fecha de nacimiento
<b>Resultado</b>	1. El sistema imprimirá cada uno de estos datos indicando cual dato debe ingresar el usuario, para que puedan ser ingresados, y los guardará en una lista de pacientes.

<b>Nombre</b>	R02 - Ingresar historia clínica
<b>Resumen</b>	La aplicación va a tener una función que le permita al usuario observar su saldo disponible.
<b>Entrada</b>	Identificación, tipo de régimen, eps, semanas cotizadas, fechas de ingreso al sistema de salud, cantidad de enfermedades, enfermedad más relevante, tipo de afiliación y costos en tratamiento.
<b>Resultado</b>	1. Luego de la previa validación de la identificación, se pedirá al usuario que ingrese esos respectivos datos, para que así se guarden en la lista del respectivo propietario.

<b>Nombre</b>	R03 - Cambiar de eps
<b>Resumen</b>	La aplicación va a tener una función que le permita al usuario cambiar de eps, siempre y cuando lleve más de 3 meses.
<b>Entrada</b>	identificación
<b>Resultado</b>	1. El sistema validará que el usuario lleva más de 3 meses en su eps actual, para así poder hacer el respectivo cambio.

<b>Nombre</b>	R04 - Calcular Porcentaje
<b>Resumen</b>	La aplicación tendrá una función que permitirá realizar los cálculos de un porcentaje.
<b>Entrada</b>	Lista de pacientes totales, lista de pacientes específicos.
<b>Resultado</b>	El sistema recibirá estas dos listas y lo que hará es dividir la lista de pacientes específicos sobre la lista de pacientes totales y lo multiplica por 100 y así retornará el porcentaje.

<b>Nombre</b>	R05 - Mostrar porcentaje de pacientes sin enfermedades
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario el porcentaje de pacientes sin enfermedades
<b>Entrada</b>	Lista de pacientes totales
<b>Resultado</b>	Dicha función lo que hará es que a través de una función lambda obtendrá los pacientes el cual su cantidad de enfermedades sea cero, y con ayuda de la función calcular porcentaje retornará dicho porcentaje.

<b>Nombre</b>	R06 - Mostrar porcentaje de pacientes por rango de edad
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario el porcentaje de pacientes por rango de edad
<b>Entrada</b>	Lista de pacientes totales
<b>Resultado</b>	Dicha función lo que hará es que a través de una función lambda obtendrá los pacientes el cual su edad la organizará de acuerdo a cada rango, y con ayuda de la función calcular porcentaje retornará dicho porcentaje.

<b>Nombre</b>	R07 - Mostrar porcentaje de pacientes por tipo de régimen
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario el porcentaje de pacientes por tipo de régimen
<b>Entrada</b>	Lista de pacientes totales
<b>Resultado</b>	Dicha función lo que hará es que a través de una función lambda obtendrá los pacientes de acuerdo a su tipo de régimen, y con ayuda de la función calcular porcentaje retornará dicho porcentaje.

<b>Nombre</b>	R08 - Mostrar porcentaje de pacientes por tipo de afiliación
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario el porcentaje de pacientes por tipo de afiliación
<b>Entrada</b>	Lista de pacientes totales
<b>Resultado</b>	Dicha función lo que hará es que a través de una función lambda obtendrá los pacientes de acuerdo a su tipo de afiliación, y con ayuda de la función calcular porcentaje retornará dicho porcentaje.

<b>Nombre</b>	R09 - Mostrar total de pacientes con cáncer
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario el total de pacientes el cual su enfermedad más relevante sea el cáncer
<b>Entrada</b>	Lista de pacientes totales
<b>Resultado</b>	Dicha función lo que hará es que a través de una función lambda obtendrá los pacientes que su enfermedad más relevante sea el cáncer, y retornará la cantidad de esa lista.

<b>Nombre</b>	R10 - Mostrar paciente con mayor costo de tratamiento
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario el nombre y apellido del paciente que tiene mayor costo de tratamiento.
<b>Entrada</b>	Lista de pacientes totales
<b>Resultado</b>	Dicha función lo que hará es que a través de una función lambda obtendrá el paciente que su costo de tratamiento sea el mayor, y retornará dicho valor, con el nombre y apellido de su respectivo propietario.

<b>Nombre</b>	R11 - Mostrar datos de un paciente
<b>Resumen</b>	La aplicación va a tener una función que le mostrará al usuario los datos de un paciente
<b>Entrada</b>	identificación
<b>Resultado</b>	Dicha función lo que hará es que se le ingresará la identificación del paciente que se quiere ver su información, luego verificará que exista dicho paciente, en caso de existir le retornará la respectiva información, en caso contraria se arrojará un mensaje de usuario no encontrado.

<b>Nombre</b>	R12 - Actualizar Eps
<b>Resumen</b>	La aplicación va a tener una función que le permitirá al usuario actualizar su eps
<b>Entrada</b>	identificación
<b>Resultado</b>	Dicha función lo que hará es que se le ingresará la identificación del paciente que se quiere actualizar su información, luego verificará que exista dicho paciente, en caso de existir, verificará que el paciente lleve más de 3 meses en su eps, en caso de llevar más de 3 meses, le permitirá hacer el cambio, en caso contrario arrojará un mensaje de error.

<b>Nombre</b>	R13 - Actualizar Tipo de Régimen
<b>Resumen</b>	La aplicación va a tener una función que le permitirá al usuario actualizar su tipo de régimen
<b>Entrada</b>	identificación
<b>Resultado</b>	Dicha función lo que hará es que se le ingresará la identificación del paciente que se quiere actualizar su información, luego verificará que exista dicho paciente, en caso de existir, le permitirá al usuario elegir su tipo de régimen, y se guardarán los respectivos cambios, en caso de no existir se arrojará un mensaje de error.

<b>Nombre</b>	R14 - Actualizar Historia Clínica
<b>Resumen</b>	La aplicación va a tener una función que le permitirá al usuario actualizar su historia clínica
<b>Entrada</b>	identificación
<b>Resultado</b>	Dicha función lo que hará es que se le ingresará la identificación del paciente que se quiere actualizar su información, luego verificará que exista dicho paciente, en caso de existir, le permitirá al usuario ingresar su nueva historia clínica, y se guardarán los respectivos cambios, en caso de no existir se arrojará un mensaje de error.

<b>Nombre</b>	R15 - Actualizar Costo de tratamientos
<b>Resumen</b>	La aplicación va a tener una función que le permitirá al usuario actualizar su costo de tratamiento
<b>Entrada</b>	identificación
<b>Resultado</b>	Dicha función lo que hará es que se le ingresará la identificación del paciente que se quiere actualizar su información, luego verificará que exista dicho paciente, en caso de existir, le permitirá al usuario ingresar su nuevo costo de tratamiento, y se guardarán los respectivos cambios, en caso de no existir se arrojará un mensaje de error.

**Link del repositorio.**

<https://github.com/FloreZ1025/ProyectoAulaVisualSantiagoDiego.git>