



Universidad Carlos III

Desarrollo de Software 2021-22

**Práctica Final: Identificación de casos de prueba**

Curso 2021-22

Fecha: 23/05/22

GRUPO: 05 A

Diego Alonso Herreruela (100432074@alumnos.uc3m.es)

# Índice

<b>REQUISITO FUNCIONAL 1</b>	<b>3</b>
DIAGRAMA DE FLUJO DE CONTROL	5
PRUEBAS	6
<b>REQUISITO FUNCIONAL 2</b>	<b>6</b>
DOMINIOS	6
ÁRBOL	7
DIAGRAMA DE FLUJO	8
PRUEBAS	9

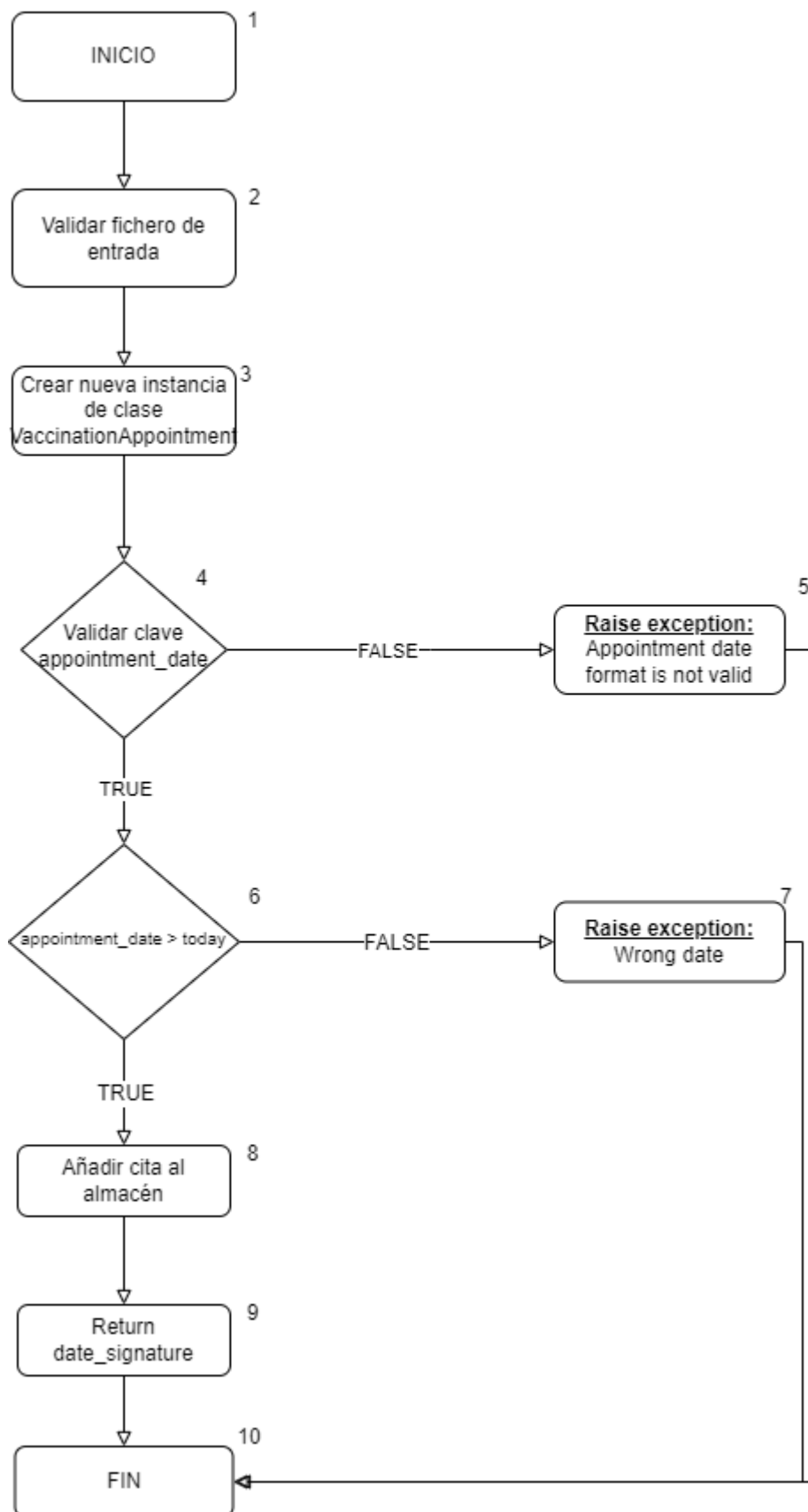
## **REQUISITO FUNCIONAL 1**

Para este requisito funcional no será necesario crear ninguna nueva función, sino que se deberá modificar una ya existente. El cambio consiste en que la función *get\_vaccine\_date*, recibirá como entrada, además del fichero con el paciente para el que se crea la cita, un string que representa la fecha de la cita que se quiere crear. La función entonces deberá comprobar si la fecha introducida está en el formato correcto, y de ser así, si la fecha es mayor al día actual.

Para obtener los casos de prueba de esta nueva parte de la función, se puede observar que se deben llevar a cabo varias comprobaciones. Por lo tanto, se ha decidido que la mejor manera de obtener los casos de prueba será mediante un diagrama de flujo.



## DIAGRAMA DE FLUJO DE CONTROL



## PRUEBAS

Las pruebas para esta modificación del método serán muy simples, ya que solo habrá que comprobar los tres posibles caminos que se pueden tomar en el diagrama. De hecho, el camino válido ya estará expuesto en el test válido previo, y solo hará falta crear los de los caminos inválidos.

## REQUISITO FUNCIONAL 2

El segundo requisito funcional sí que requerirá la creación de una nueva función, llamada *cancel\_vaccine\_date*, que tendrá la capacidad de anular las citas que se hayan creado. Para esto, deberá recibir un fichero JSON con tres campos: el string con 64 caracteres hexadecimales que representan a la cita (*date\_signature*), un tipo de cancelación formado por un string que solo puede tener dos valores; “Temporal” y “Final”. y un string de 2 a 100 caracteres que exprese el motivo de la cancelación. Dependiendo del valor del tipo de cancelación, la función deberá ofrecer o no la posibilidad de poder reactivar las citas canceladas.

Para generar las pruebas de comprobación del correcto funcionamiento de la función, se ha optado por la utilización de un árbol de derivación, ya que la estructura de los parámetros de entrada pueden considerarse como una gramática.

Como además de validar la entrada, la función tiene que comprobar si la cita seleccionada se encuentra en el almacén, que la fecha de cancelación de la cita es menor que la fecha establecida para la cita, o que que la cita no esté cancelada ya, se incluirán estos tests mediante otro diagrama de flujo. El caso válido de este método se comprobará de esta manera también ya que es el más completo.

## DOMINIOS

El fichero de entrada tiene la siguiente estructura:

```
{ "date_signature": "<String de 64 caracteres hexadecimales>",  
  "cancelation_type": "Temporal | Final"  
  "reason": "<String de 2 a 100 caracteres>"
```

Sabiendo el formato que debe tener la entrada de la función, podemos generar los dominios para construir el árbol

```
Fichero ::= Inicio_objeto Datos Fin_objeto  
Inicio_objeto ::= (  
Fin_objeto ::= )
```

```
Datos ::= Campo1 Separador Campo2 Separador Campo3  
Campo1 ::= Etiqueta_dato1 Igualdad Valor_dato1
```

```

Campo2 ::= Etiqueta_dato2 Igualdad Valor_dato2
Campo3 ::= Etiqueta_dato3 Igualdad Valor_dato3
Separador ::= ,
Igualdad ::= :

Etiqueta_dato1 ::= Comillas Valor_etiqueta1 Comillas
Valor_etiqueta1 ::= date_signature

Valor_dato1 ::= Comillas Valor1 Comillas
Valor1 ::= a|b|c|d|e|f|0|1|2|3|4|5|6|7|8|9| {64}

Etiqueta_dato2 ::= Comillas Valor_etiqueta2 Comillas
Valor_etiqueta2 ::= cancelation_type

Valor_dato2 ::= Comillas Valor2 Comillas
Valor2 ::= Temporal | Final

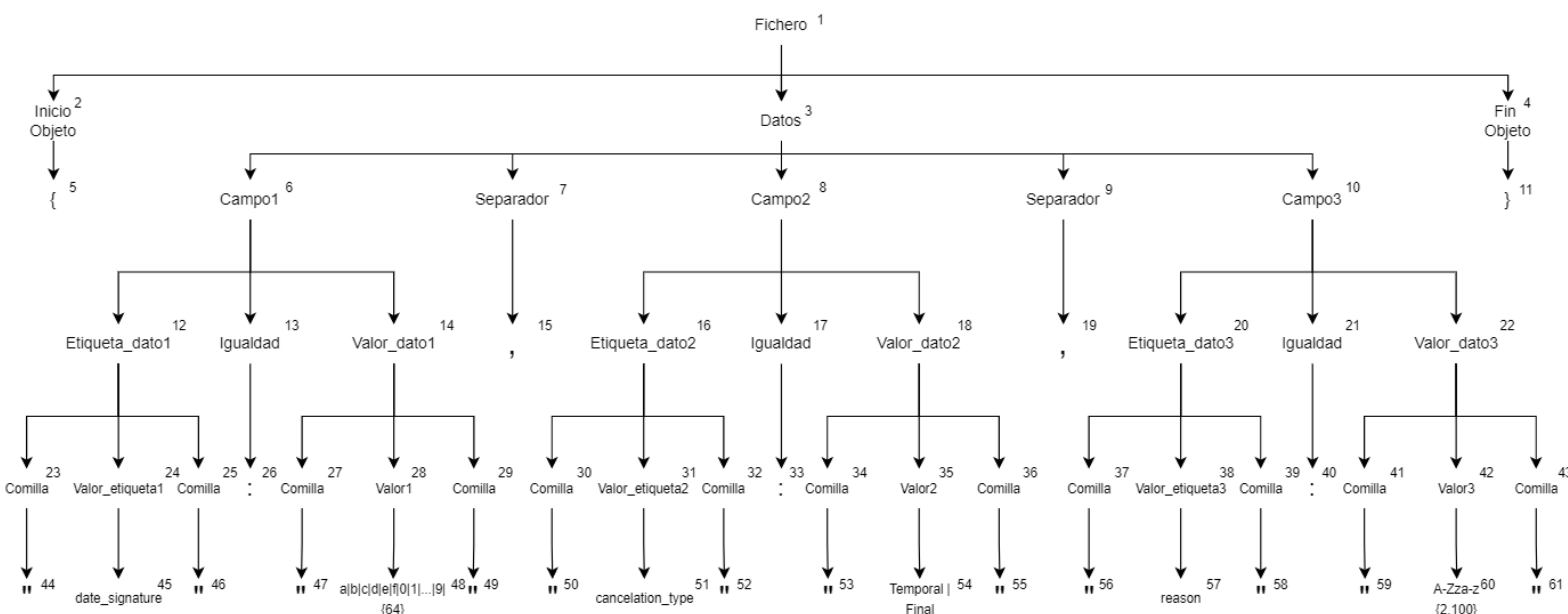
Etiqueta_dato3 ::= Comillas Valor_etiqueta3 Comillas
Valor_etiqueta3 ::= reason

Valor_dato3 ::= Comillas Valor3 Comillas
Valor3 ::= A-Za-z {2,100}

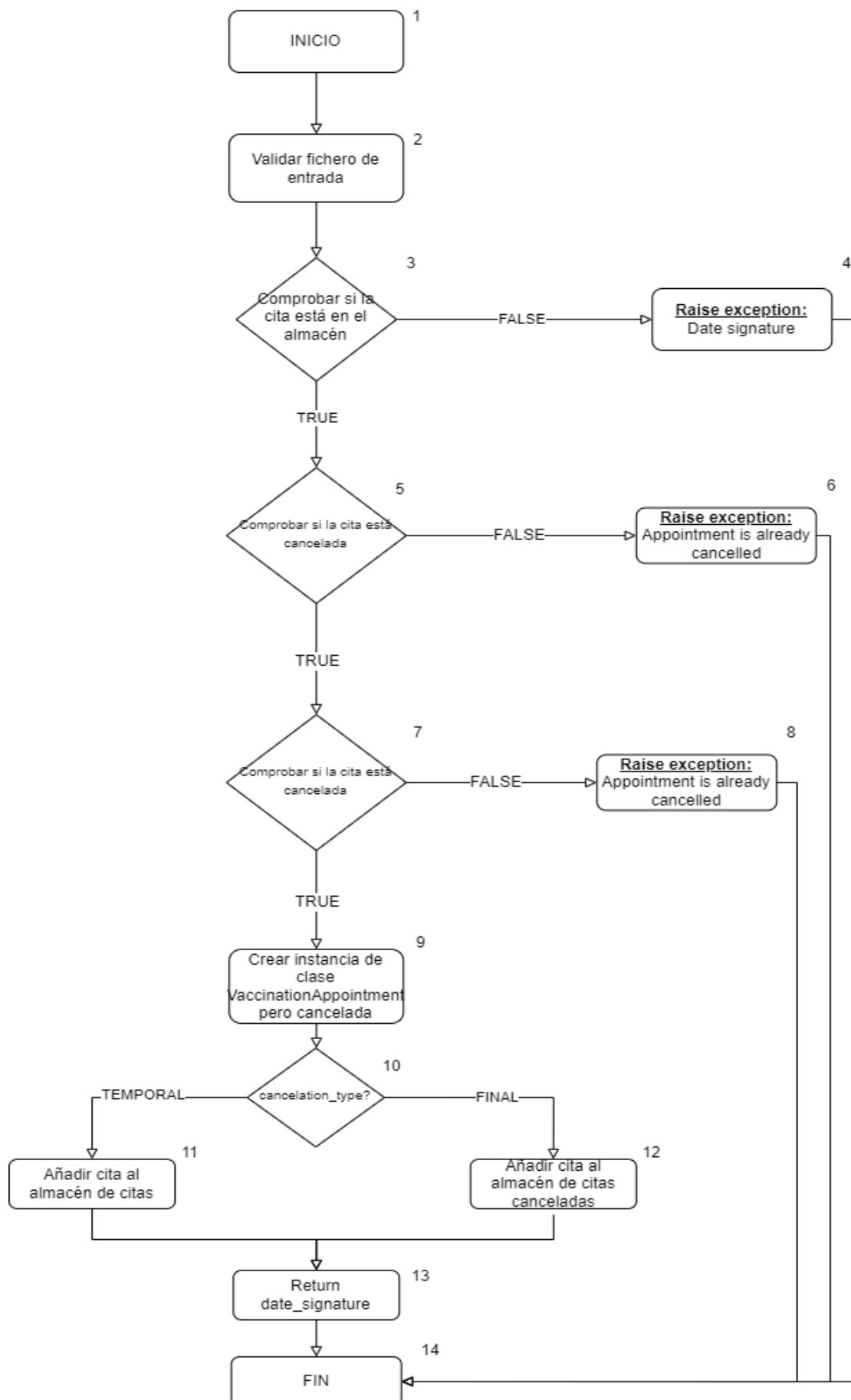
```

El siguiente paso será, una vez establecidos los dominios, construir el árbol. Quedaría de la siguiente manera

## ÁRBOL



## DIAGRAMA DE FLUJO





## PRUEBAS

Después de generar el árbol, el siguiente paso será generar los tests de prueba. Para los nodos no terminales, se deberán probar los casos en los que estos estén borrados o duplicados. En el caso de los nodos terminales, se deberá probar su eliminación, duplicación, y codificación.

Al ser la longitud del campo de entrada que representa la razón de cancelación de la cita un string entre 2 y 100 caracteres, se utilizarán valores límite para representar los casos de prueba no válidos, ya que los válidos ya están recogidos.