

# Universidad de Alcalá

## Escuela Politécnica Superior

GRADO EN INGENIERÍA INFORMÁTICA



### Trabajo Fin de Grado

Herramienta de soporte al desarrollo de competencias básicas del  
ciudadano digital

ESCUELA POLITECNICA

**Autor:** Miguel Joaquín Hernández Hernández

**Tutor/es:** Carmen Pagés Arévalo

Septiembre 2016

UNIVERSIDAD DE ALCALÁ  
Escuela Politécnica Superior

**GRADO EN INGENIERÍA INFORMÁTICA**

Trabajo Fin de Grado  
Herramienta de soporte al desarrollo de  
competencias básicas del ciudadano digital

**Autor:** Miguel Joaquín Hernández Hernández  
**Director/es:** Carmen Pagés Arévalo

**TRIBUNAL:**

**Presidente:**

**Vocal 1º:**

**Vocal 2º:**

**FECHA:** Septiembre 2016

Dedicado a mi abuelo. Gracias, por tanto.

## Agradecimientos

Quiero agradecerse a mi familia, a mis padres que sin ellos no hubiera sido posible.

A mi padre que es mi ejemplo de esfuerzo, constancia y dedicación todos los días.

A mi madre que es mi heroína que me ha enseñado a que puedo conseguir todo lo que me preponga y es un ejemplo para mi todos los días.

A mi novia por enseñarme lo que es la perseverancia por aguantarme en esta aventura en los buenos y los malos momentos. ella es la que más sabe lo que ha sido el día a día y este es su resultado también.

A mis abuelos por haberme educado enseñándome los valores y no limitándome las metas que pueda proponerme, en especial a mi abuelo él tiene gran parte de culpa de lo que haya podido conseguir, enseñándome respeto por lo que hago.

A mis compañeros por su apoyo y por convertir momentos difíciles en divertidos.

Y me gustaría dar un agradecimiento especial a mi tutora Carmen Pagés Arévalo por su comprensión y apoyo en momentos difíciles además de sus conocimientos y su buen hacer.

## Índice

Resumen .....	6
Summary .....	6
Palabras claves .....	7
Resumen extendido .....	7
Teclado:.....	7
Ratón:.....	9
Carpetas: .....	9
Glosario de acrónimos y abreviaturas .....	10
Memoria .....	11
Introducción:.....	11
Base teórica:.....	12
Ciudadano Digital y competencias digitales.....	12
Certificación E-Citizen:.....	13
Modelo de vistas 4+1: .....	13
Ingeniería del software .....	15
Requisitos.....	15
Requisitos funcionales.....	15
Requisitos no funcionales.....	17
Descripciones de casos de uso .....	18
Casos de uso:.....	18
Diagramas .....	26
Vista de Casos de Uso.....	26
Diagramas de casos de uso:.....	26
Paquete Teclado:.....	28
Paquete Ratón:.....	29
Paquete Carpetas: .....	30
Paquete Visualización:.....	31
Diagrama de navegabilidad: .....	31
Vista de desarrollo: .....	33

Diagrama de componentes:.....	33
Trazabilidad .....	33
Estructura de carpetas: .....	34
Vista física: .....	36
Diagrama de Despliegue:.....	36
Vista de proceso.....	36
Teclado.....	37
Ratón:.....	42
Vista lógica .....	58
Diagrama de clases archivos HTML y CSS .....	58
Diagrama de clases archivos HTML y JS .....	61
Pruebas .....	68
Herramientas y tecnologías utilizadas .....	68
Herramientas .....	68
NetBeans.....	68
Enterprise Architect.....	69
Gimp.....	69
Tecnologías .....	70
JavaScript .....	70
JQuery .....	70
HTML5.....	70
CSS .....	71
Bootstrap.....	71
Manual de usuario .....	72
Página principal:.....	72
Página teoría de secciones .....	73
Secciones: .....	73
Ejercicios .....	74
Conclusión.....	76
Bibliografía .....	78

## Resumen

Herramientas Ciudadano Digital (HCD) es una aplicación web para dar soporte al desarrollo de competencias básicas del ciudadano digital. HCD está dividida en 3 secciones, categorizadas por lo que el usuario podrá aprender:

- Teclado: consiste en un conjunto de ejercicios y teoría donde el usuario podrá practicar el uso del teclado.
- Ratón: en esta sección encontraremos ejercicios y teoría donde el usuario podrá aprender el uso del ratón.
- Carpetas: conjunto de ejercicios para aprender a usar las carpetas, simulando un explorador de un sistema operativo. Además, podrá aprender que es una carpeta y sus características.

## Summary

Herramientas Ciudadano Digital (HCD) is a web application to support the development of basic competences of digital citizen. HCD is divided into three sections, categorized so the user can learn:

- Keyboard: consists of a set of exercises and theory where the user can practice using the keyboard.
- Mouse: In this section you will find exercises and theory where the user can learn to use the mouse.
- Folders: A set of exercises to learn how to use folders, simulating a browser operating system. You can also learn that it is a folder and its feature.

## Palabras claves

Aplicación web, Ciudadano digital, Competencias, Ingeniería del Software.

## Resumen extendido

El trabajo consiste en la realización de una aplicación web para dar soporte al desarrollo de competencias básicas del ciudadano digital. El ciudadano digital es cualquier persona, no importa su edad ni su raza, que utiliza la tecnología como una herramienta de forma activa. La aplicación nace con la idea de que personas con conocimientos limitados en el uso de la informática, puedan aprender tareas básicas en el uso del teclado, del ratón y de las carpetas que proporcionan los sistemas operativos y poderlas ayudar a que no se sientan excluidas por no tener los conocimientos necesarios en el uso de la informática, que tan importante es a día de hoy.

La aplicación consta de 3 secciones principales además de la página principal donde los usuarios podrán aprender teoría del teclado, del ratón y de las carpetas. Cada sección consta de una introducción teórica, donde se explican las distintas partes y las funcionalidades de los elementos nombrados anteriormente y de una serie de ejercicios y juegos en los que el usuario podrá aprender y practicar las tareas básicas de cada uno de los elementos.

A continuación, voy a nombrar y realizar una breve explicación de los ejercicios y juegos que conforman las distintas secciones, por orden en el que aparecen en la aplicación:

### Teclado:

- Escribir palabras: aparecerá una palabra y el usuario deberá escribirla correctamente.
- Escribir frase: aparecerá una frase y el usuario deberá escribirla correctamente.
- Bloqueo de mayúsculas: el usuario deberá escribir utilizando el botón de bloqueo de mayúsculas.



- Mayúsculas: aparece una palabra en mayúsculas y el usuario debe escribirla correctamente.
- Mayúsculas y minúsculas: aparece una palabra en mayúsculas y minúsculas, el usuario debe escribirla correctamente.
- Números: aparece un número y el usuario debe escribirlo correctamente.
- Borrar palabras: aparece una palabra y el usuario debe borrarla
- Cursor: aparece una frase y posición donde el usuario deberá mover el cursor.
- Seleccionar: aparece una frase y el usuario debe seleccionar un número determinado de veces.
- Mover el cursor y borrar: aparece una frase y posición donde el usuario deberá mover el cursor, una vez movido aparece la frase que debe borrar.
- Escribir palabras con punto y coma: aparece una frase con puntos y comas y el usuario debe escribirla correctamente.
- Escribir con acentos: aparece una frase con acentos y el usuario debe escribirla correctamente.
- Escritura con diéresis: aparece una palabra con diéresis y el usuario debe escribirla correctamente.
- Escritura con salto de línea: aparece una frase con saltos de línea y el usuario debe escribirla correctamente.
- Ejercicio de copiar y pegar: aparece una frase y la palabra que el usuario debe copiar y pegar correctamente.
- Ejercicio de cortar y pegar: aparece una frase y la palabra que el usuario debe cortar y pegar correctamente.
- Ejercicio de interrogaciones y exclamaciones: aparece una frase con interrogaciones y exclamaciones y el usuario debe escribirla correctamente.
- Ejercicio de números ordinales: aparece una frase con números ordinales y el usuario debe escribirla correctamente
- Ejercicio de símbolos: aparece un símbolo y el usuario debe escribirlo correctamente.
- Ejercicio de resumen: aparece una frase que contendrá lo aprendido en los ejercicios escritos anteriormente y el usuario deberá escribirla correctamente.

#### Ratón:

- Ejercicio click izquierdo: aparece una imagen y el usuario debe pulsar con el botón izquierdo del ratón unas veces determinadas.
- Ejercicio scroll vertical: el usuario deberá desplazarse verticalmente sobre la página para encontrar una imagen y volver a subir para completar el ejercicio.
- Ejercicio scroll horizontal: el usuario deberá desplazarse horizontalmente sobre la página para encontrar una imagen y volver al inicio para completar el ejercicio.
- Ejercicio doble click: aparece una imagen y el usuario debe hacer doble click en ella unas veces determinadas.
- Ejercicio desplazamiento horizontal: aparece un recorrido horizontal y el usuario deberá llevar el cursor de una posición a otra y volver.
- Ejercicio desplazamiento horizontal y vertical: aparece un recorrido que contiene tramos horizontales y verticales y el usuario deberá llevar el cursor de una posición a otra.
- Ejercicio seleccionar texto: aparece un texto y el usuario debe seleccionarlo unas veces determinadas.
- Ejercicio copiar texto: aparece una frase y la palabra que el usuario debe copiar y pegar correctamente.
- Ejercicio cortar texto: aparece una frase y la palabra que el usuario debe cortar y pegar correctamente.
- Ejercicio arrastrar elementos: aparecen unas series de figuras y el usuario deberá arrastrarlas hasta una posición determinada.

#### Carpetas:

- Ejercicio navegar por carpetas: aparecen una serie de carpetas y el usuario deberá entrar en una de ellas y volver a la ruta anterior.
- Ejercicio crear y eliminar carpetas: aparecen una serie de carpetas y el usuario deberá crear una carpeta y más tarde eliminar la carpeta que desee.
- Ejercicio cambiar nombre carpetas: aparecen una serie de carpetas y el usuario deberá cambiar el nombre a una carpeta
- Ejercicio cortar carpetas: aparecen una serie de carpetas y el usuario debe cortar y pegar una carpeta.
- Ejercicio copiar carpetas: aparecen una serie de carpetas y el usuario deberá copiar y pegar una carpeta.

Todos los ejercicios contienen una explicación de cómo se deben hacer, para que el usuario pueda aprender. Cuando el ejercicio sea correcto, la aplicación mostrará un mensaje y además cada ejercicio tendrá un botón donde el usuario podrá ver si el ejercicio es incorrecto y si desean cambiar de ejercicio.

Para la realización del trabajo se han seguido una serie de pasos, descritos brevemente a continuación:

- Investigación de la certificación E-Citizen a la cual ira destinada la aplicación.
- Construcción de la aplicación:
  - Obtención y documentación de los requisitos: a través de una reunión con mi tutora y de diversos correos.
  - Análisis y diseño: he utilizado el modelo de vistas 4+1
  - Implementación: donde he implementado primero el código HTML y CSS y más tarde la funcionalidad en JavaScript con la librería JQuery.
  - Pruebas: que incluían pruebas de la visualización de la aplicación y de la funcionalidad.

## Glosario de acrónimos y abreviaturas

ECDL	European Computer Driving License
HCD	Herramientas Ciudadano Digital
UML	Unified Modeling Language
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
JS	JavaScript

## Memoria

### Introducción:

Herramientas Ciudadano Digital es una aplicación web que nace con la idea de poder ayudar a las personas con conocimientos limitados en informática a desarrollar las competencias básicas en el uso del teclado, del ratón y de las carpetas y así ayudarlas aprender contenidos de la certificación E-Citizen.

El principal objetivo del trabajo, es que los usuarios puedan aprender el uso de estas competencias básicas a través de ejercicios y juegos que resulten intuitivos y divertidos, de forma que puedan aprender y divertirse a la vez.

El principal usuario de la aplicación serán personas mayores, por lo que otro objetivo ha sido que la aplicación sea lo más atractiva, agradable e intuitiva posible, con secciones donde los usuarios, puedan aprender teoría y realizar ejercicios en un entorno visual que resulte agradable y ameno.

La aplicación consta de 3 secciones categorizadas según lo que el usuario va a aprender:

- Teclado: en esta sección el usuario podrá aprender teoría relacionada con el teclado y realizar ejercicios que le ayuden a practicar tareas básicas en el uso del teclado.
- Ratón: en esta sección el usuario podrá aprender teoría relacionada con el ratón y realizar ejercicios en el que el usuario podrá realizar tareas básicas con este periférico y así poder practicar su uso.
- Carpetas: en esta sección el usuario podrá aprender teoría de las carpetas y podrá realizar ejercicios que le ayuden a practicar tareas básicas que se pueden realizar con las carpetas que ofrece el sistema operativo.

## Base teórica:

### Ciudadano Digital y competencias digitales

Se denomina ciudadano digital a cualquier persona no importa su edad, su sexo o su etnia que utiliza la tecnología como herramienta y así mejorar su participación en la sociedad.

El ciudadano digital debe tener una serie de competencias para poder utilizar la tecnología, a estas competencias se las denomina competencias digitales.

Las competencias digitales son un conjunto de conocimientos y capacidades para la utilización efectiva de las tecnologías de la

información para alcanzar los objetivos relacionados con el trabajo, el aprendizaje, el uso del tiempo libre, la inclusión y participación en la sociedad. Estas competencias requieren de conocimientos relacionados con el lenguaje específico básico y de los derechos y libertades que asisten a las personas en el mundo digital.

También precisa del desarrollo de conocimientos en el acceso, el procesamiento y el uso de la información y su comunicación.

Además de todo esto la adquisición de las competencias requiere de aptitudes y valores que permitan a los usuarios adaptarse a las nuevas necesidades tecnológicas muy cambiantes en estos tiempos.



## Certificación E-Citizen:

La certificación E-Citizen (ciudadano digital) pretende que las personas con un conocimiento informático limitado aprendan y saquen el máximo provecho de los ordenadores y de Internet. De esta forma obtendrán las habilidades esenciales para acceder y participar de forma activa en la sociedad de la información.

El destinatario de la certificación E-Citizen, es todo ciudadano que posee un conocimiento sobre ordenadores e Internet limitado y se sienten excluidos por no tener los conocimientos necesarios.

El programa de certificación E-Citizen está diseñado para ser lo más simple y directo posible. Los alumnos trabajan con un instructor que les guía por los distintos contenidos del programa. El programa E-Citizen está estructurado de forma muy clara y lógica, lo que permite que la progresión sea suave y continua, además de ser flexible lo que permite a las personas aprender a su propio ritmo. Por lo general la formación tiene una duración de 30 horas.

La certificación está patrocinada por la Fundación ECDL, una organización internacional que está presente en toda Europa y en todo el mundo bajo las siglas ICDL. La fundación se dedica a elevar los niveles de competencias digitales de la sociedad.

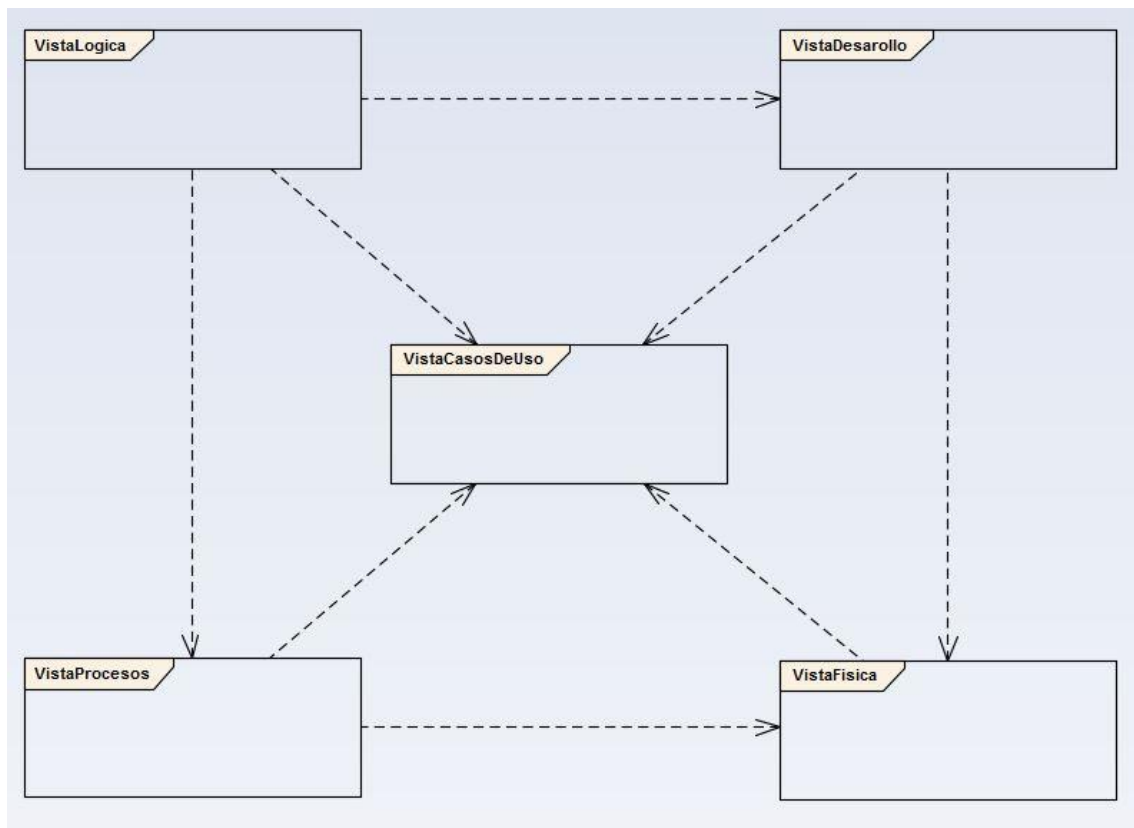


Cualquier persona, con independencia de su nivel académico y de su profesión, pueden acreditar sus conocimientos informáticos a través de la fundación. A día de hoy más de 14 millones de personas en el mundo ya se han unido a ECDL.

## Modelo de vistas 4+1:

Para la realización de la ingeniería del software he utilizado el modelo de vistas de arquitectura 4+1. El modelo 4+1 fue diseñado por el profesor Philippe Kruchten, su objetivo es el de describir la arquitectura de un sistema software con el uso de múltiples vistas. Las vistas describen el

sistema desde distintos puntos de vista para diferentes interesados como usuarios finales, desarrolladores o el ingeniero de sistemas.



Las vistas que componen el modelo son:

- Vista lógica: representa la funcionalidad que el sistema proporciona a los usuarios finales. Se representa con un diagrama de clases, de comunicación o de secuencia UML.
- Vista de desarrollo: representa el sistema desde la perspectiva del programador, se muestra como está dividido el sistema software en componentes y las dependencias entre ellos. Se representa con diagramas de componentes UML.
- Vista de proceso: explica los procesos del sistema y como se comunican, se enfoca en el comportamiento del sistema en tiempo de ejecución. Se representa con diagramas de actividad.

- Vista física: representa el sistema desde la perspectiva de un ingeniero de sistemas, describe las conexiones físicas entre los componentes. Se representa con diagramas de despliegue.
- Vista de casos de uso: además de las vistas descritas anteriormente existe la vista de casos de uso que tiene la función de unir y relacionar las otras 4 vistas y describir la arquitectura, de ahí viene el nombre de 4+1. Para describirla se utilizan diagramas de casos de uso.

## Ingeniería del software

En esta sección voy a explicar cuál ha sido el proceso para la realización de la ingeniería del software que he desarrollado para realizar la aplicación.

### Requisitos

En primer lugar, realice una de las fases más importantes del proyecto, la obtención de requisitos. A través de una reunión y diversos emails con mi tutora, obtuve una lista de requisitos funcionales y no funcionales de lo que se quería conseguir de la aplicación. A partir de ahí he realizado las tablas que mostrare a continuación especificando los requisitos funcionales y no funcionales.

Las tablas tienen dos columnas:

- Código que nos servirá más adelante cuando realicemos la especificación de casos de uso poder tener una trazabilidad de los requisitos que satisfacemos con cada caso de uso 7.
- Descripción: donde se detalla el requisito.

### Requisitos funcionales

Código	Descripción
RF-1	La aplicación debe tener un menú principal en el que accederemos a los distintas partes del programa



RF-2	La aplicación debe permitir practicar el uso del teclado
RF-2.1	La aplicación debe permitir realizar ejercicios de escritura
RF-2.1.1	La aplicación debe permitir realizar ejercicios de escritura tanto con letras mayúsculas y minúsculas
RF-2.1.2	La aplicación debe permitir realizar ejercicios de escritura con caracteres numéricos
RF-2.1.3	La aplicación debe permitir realizar ejercicios de borrado de caracteres
RF-2.1.4	La aplicación debe permitir realizar ejercicios de escritura con acentos y diéresis
RF-2.1.5	La aplicación debe permitir realizar ejercicios de escritura con comas y puntos
RF-2.1.6	La aplicación debe permitir realizar ejercicios de escritura con símbolos del teclado - Estos símbolos son: \ # \$ % & ( ) * + - / : ; < = > _
RF-2.2	La aplicación debe permitir realizar ejercicios de copiar y pegar texto a través de los controles del teclado
RF-2.3	La aplicación debe permitir realizar ejercicios de cortar y pegar texto a través de los controles del teclado
RF-2.4	La aplicación debe permitir realizar ejercicios para aprender a controlar el cursor con el teclado
RF-2.5	La aplicación debe permitir realizar ejercicios para aprender a seleccionar palabras a través del teclado
RF-3	La aplicación debe permitir practicar el uso del ratón
RF-3.1	La aplicación debe permitir realizar ejercicios para practicar el uso de las barras de desplazamientos horizontal y vertical
RF-3.2	La aplicación debe permitir realizar ejercicios de hacer click con los botones izquierdo y derecho del ratón
RF-3.3	La aplicación debe permitir realizar ejercicios de hacer doble click con el botón izquierdo del ratón
RF-3.4	La aplicación debe permitir realizar ejercicios de selección de texto con el ratón
RF-3.5	La aplicación deber permitir realizar ejercicios de copiar texto a través del ratón

RF-3.6	La aplicación deber permitir realizar ejercicios de cortar texto a través del ratón
RF-3.7	La aplicación debe permitir realizar ejercicios de desplazamiento del puntero por la pantalla con el ratón
RF-3.8	La aplicación debe permitir realizar ejercicios de arrastrar elementos con el click izquierdo
RF-4	La aplicación debe permitir practicar el uso de carpetas.
RF-4.1	La aplicación debe permitir realizar ejercicios de creación y borrado de carpetas
RF-4.2	La aplicación debe permitir realizar ejercicios de cambio de nombre de carpetas
RF-4.3	La aplicación debe permitir realizar ejercicios de navegación entre carpetas
RF-4.4	La aplicación deber permitir realizar ejercicios de copiar carpetas
RF-4.5	La aplicación deber permitir realizar ejercicios de cortar carpetas
RF-5	La aplicación deberá permitir moverse entre ejercicios
RF-6	La aplicación deberá dar una breve explicación de cada uno de los ejercicios, para facilitar el uso al usuario
RF-7	La aplicación deberá realizar una introducción de cada una de las partes del programa definiendo así el teclado, el ratón y las carpetas
RF-8	La aplicación deberá mostrar mensajes
RF-8.1	La aplicación deberá mostrar un mensaje cuando el usuario finalice un ejercicio correctamente
RF-8.2	La aplicación deberá mostrar un mensaje cuando el usuario desee cambiar de ejercicio y no haya terminado

#### Requisitos no funcionales

Código	Descripción
RNF-1	El lenguaje de programación elegido para desarrollar el sistema será JavaScript con la librería JQuery.
RNF-2	El estilo debe estar definido con HTML y CSS.

RNF-3	La interfaz gráfica deberá ser intuitiva y atractiva.
RNF-4	El sistema debe de poder permitir implementar nuevas funcionalidades en el futuro y poder modificar las ya creadas.

### Descripciones de casos de uso

A partir de la especificación de los requisitos obtenidos elabore la especificación de requisito donde se muestra los siguientes campos:

Descripción: se detalla el caso de uso.

Requisitos cubiertos: requisitos que se satisfacen con ese caso de uso.

Prioridad (Alta, Media o Baja): prioridad para elaborar el caso de uso, siendo los casos de uso con prioridad alta los que primero se han realizado. Para elegir la prioridad he tenido en cuenta el nivel de dificultad que me suponía o creía que iba a tener el caso de uso, siendo los casos de uso más difíciles los que tienen la prioridad más alta.

### Casos de uso:

#### 1. Realizar ejercicio de escritura

- **Descripción**

El usuario podrá realizar ejercicios de escritura con el teclado.

Los ejercicios serán los siguientes:

- Ejercicio de escritura tanto con letras mayúsculas y minúsculas.
- Ejercicio de escritura con caracteres numéricos
- Ejercicio de borrado de caracteres
- Ejercicio de escritura con acentos y diéresis
- Ejercicio de escritura con comas y puntos
- Ejercicio de escritura con símbolos del teclado

Estos símbolos son: \ # \$ % & ( ) \* + - / : ; < = > \_

- **Requisitos cubiertos**  
RF-2.1.1, RF-2.1.2, RF-2.1.3, RF-2.1.4, RF-2.1.5 y RF-2.1.6
- **Prioridad**  
Alta

2. Realizar ejercicio copiar texto con teclado

- **Descripción**  
El usuario podrá realizar ejercicios de copiar y pegar texto a través de los controles del teclado
- **Requisitos cubiertos**  
RF-2.2
- **Prioridad**  
Alta

3. Realizar ejercicio de cortar texto con teclado

- **Descripción**  
El usuario podrá realizar ejercicios de cortar y pegar texto a través de los controles del teclado
- **Requisitos cubiertos**  
RF-2.3
- **Prioridad**  
Alta

4. Realizar ejercicio mover cursor con teclado

- **Descripción**  
El usuario podrá practicar ejercicios de para aprender a controlar el cursor con los controles del teclado
- **Requisitos cubiertos**  
RF-2.4

- **Prioridad**  
Alta
5. Realizar ejercicio de seleccionar texto con teclado
- **Descripción**  
El usuario podrá practicar ejercicios para aprender a seleccionar palabras a través del teclado
  - **Requisitos cubiertos**  
RF-2.5
  - **Prioridad**  
Media
6. Realizar ejercicio desplazamiento con ratón
- **Descripción**  
El usuario podrá practicar ejercicios para aprender a moverse por una página web a través de las barras de desplazamientos verticales y horizontales
  - **Requisitos cubiertos**  
RF-3.1
  - **Prioridad**  
Alta
7. Realizar ejercicio click
- **Descripción**  
El usuario podrá realizar ejercicios de pulsar con los botones del ratón, en lo que tendrán que pulsar sobre imágenes con el botón izquierdo o derecho
  - **Requisitos cubiertos**  
RF-3.2
  - **Prioridad**  
Alta

8. Realizar ejercicio doble-click

- **Descripción**

El usuario podrá realizar ejercicios de hacer doble click.

- **Requisitos cubiertos**

RF-3.3

- **Prioridad**

Media

9. Realizar ejercicio selección de textos con ratón

- **Descripción**

El usuario podrá realizar ejercicios donde deberán seleccionar textos con el ratón. Aparecerán textos y se pedirá que seleccione una serie de palabras.

- **Requisitos cubiertos**

RF-3.4

- **Prioridad**

Media

10. Realizar ejercicio copiar texto con ratón

- **Descripción**

El usuario podrá realizar ejercicios donde deberán seleccionar textos y copiarlos en otra parte de la aplicación.

- **Requisitos cubiertos**

RF-3.5

- **Prioridad**

Alta

#### 11.Realizar ejercicio cortar texto con ratón

- **Descripción**

El usuario podrá realizar ejercicios donde deberán seleccionar textos y cortarlos en otra parte de la aplicación.

- **Requisitos cubiertos**

RF-3.6

- **Prioridad**

Media

#### 12.Realizar ejercicio desplazamiento cursor con ratón

- **Descripción**

El usuario podrá practicar ejercicios de desplazamientos del puntero, donde deberán llevar el puntero desde un punto origen hacia otro destino intentando no salirse del camino que se les marcará.

- **Requisitos cubiertos**

RF-3.7

- **Prioridad**

Alta

#### 13.Realizar ejercicio de arrastrar

- **Descripción**

El usuario podrá practicar ejercicios de arrastrar elementos con el ratón, en los que tendrán que coger una figura y arrastrarla hasta una posición determinada.

- **Requisitos cubiertos**

RF-3.8

- **Prioridad**

Alta

#### 14. Realizar ejercicio creación y borrado de carpetas

- **Descripción**

El usuario podrá realizar ejercicios donde aprenderán a crear carpetas y borrar carpetas. Se les solicitará que creen una carpeta en la ruta que deseen y luego borren una carpeta que deseen.

- **Requisitos cubiertos**

RF-4.1

- **Prioridad**

Alta

#### 15. Realizar ejercicio cambiar nombre de carpeta

- **Descripción**

El usuario podrá realizar ejercicios donde aprenderán a cambiar el nombre de carpetas. Aparecerán varias carpetas y se pedirá a los usuarios que a una carpeta determinada se le asigne el nombre que deseen.

- **Requisitos cubiertos**

RF-4.2

- **Prioridad**

Alta

#### 16. Realizar ejercicio navegación de carpetas

- **Descripción**

El usuario podrá realizar ejercicios donde aprenderán a moverse entre distintas carpetas.

- **Requisitos cubiertos**

RF-4.3

- **Prioridad**

Alta



#### 17. Realizar ejercicios copiar carpeta

- **Descripción**

El usuario podrá realizar ejercicios donde aprenderán a copiar una carpeta o archivo. Se les pedirá que copien el archivo o carpeta que deseen en la ubicación que escojan.

- **Requisitos cubiertos**

RF-4.4

- **Prioridad**

Alta

#### 18. Realizar ejercicios cortar carpeta

- **Descripción**

El usuario podrá realizar ejercicios donde aprenderán a cortar una carpeta o archivo. Se les pedirá que copien el archivo o carpeta que deseen en la ubicación que escojan.

- **Requisitos cubiertos**

RF-4.5

- **Prioridad**

Alta

#### 19. Ver resultado

- **Descripción**

Cuando el usuario realice el ejercicio correctamente el sistema deberá mostrar un mensaje automático confirmando que se ha realizado correctamente. También aparecerá un mensaje cuando el usuario pulse el botón de comprobar resultado y no sea correcto.

- **Requisitos cubiertos**

RF-8.1 y RF-8.2

- **Prioridad**

Baja

## 20.Ver sección

- **Descripción**

El usuario podrá ver las distintas secciones de la aplicación donde aparecerán los ejercicios que están disponibles en cada una de las secciones.

Las secciones de la aplicación serán:

- Pantalla principal
- Teclado
- Ratón
- Carpetas

Cada una de las partes contendrá una breve explicación sobre lo que se va a realizar y curiosidades del ratón, del teclado y de las carpetas.

- **Requisitos cubiertos**

RF-1, RF-5 y RF-7

- **Prioridad**

Baja

## 21.Cambiar ejercicio

- **Descripción**

El usuario podrá cambiar de ejercicio cuando desee ya sea desde el menú principal de la sección donde se encuentre o desde el propio ejercicio a través de un botón que se habilitará

- **Requisitos cubiertos**

RF-5

- **Prioridad**

Baja

## Diagramas

Para describir la arquitectura del sistema voy a utilizar el modelo de vistas 4+1 (ver apartado bases teóricas), siendo cada apartado una de las vistas que conformar este modelo.

### Vista de Casos de Uso

La vista de casos de uso es la vista utilizada para describir la secuencia e interacción entre objetos y sirve para identificar el diseño de la arquitectura. Para describir esta vista he utilizado los siguientes diagramas:

- Diagramas de casos de uso
- Diagrama de navegabilidad

#### Diagramas de casos de uso:

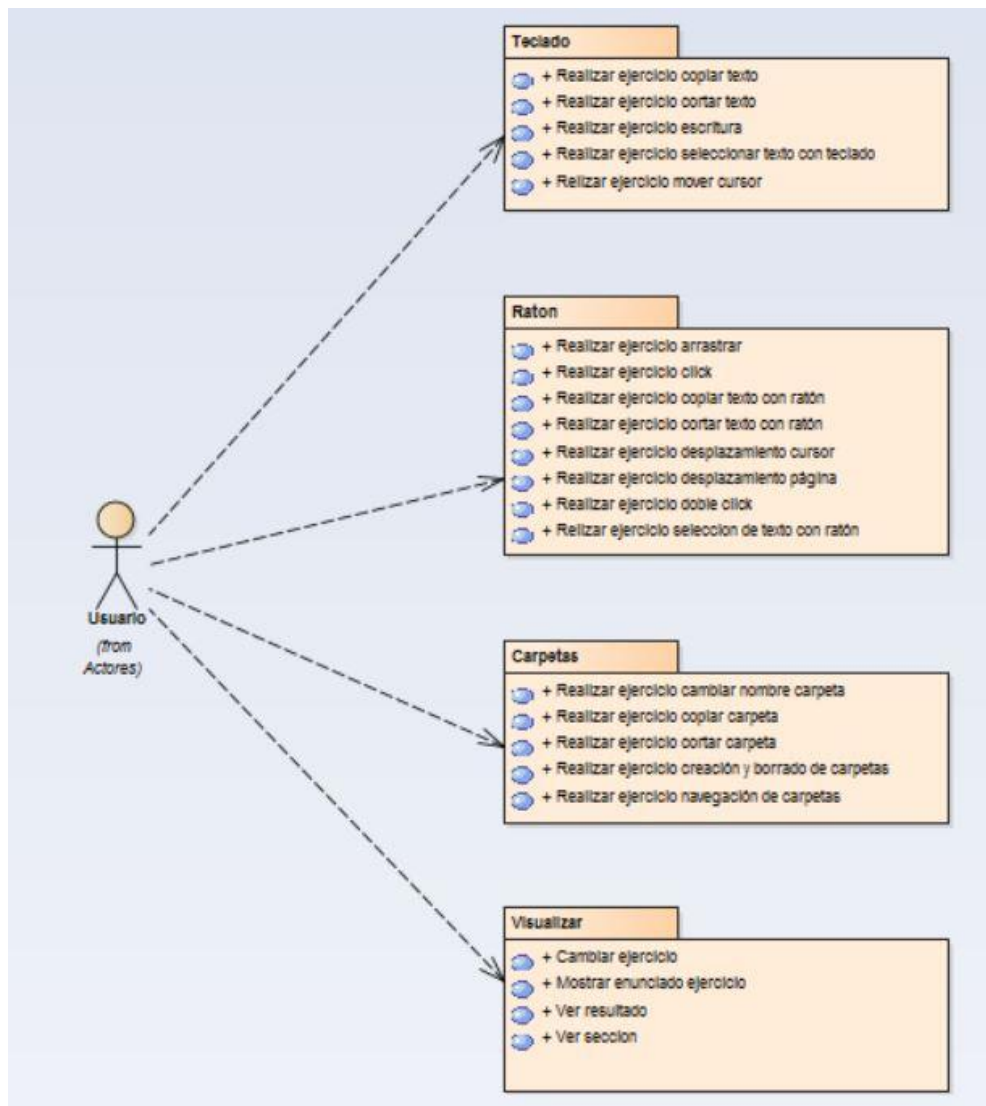
El primero paso fue identificar a los actores de nuestro sistema.  
Para Herramientas Ciudadano Digital identificamos al actor: Usuario



El actor usuario será la persona que utiliza la aplicación.  
Ahora vamos a ver de qué casos de uso depende, para ordenar los casos de uso los he paquetizado por la funcionalidad que tienen, obteniendo así cuatro paquetes:

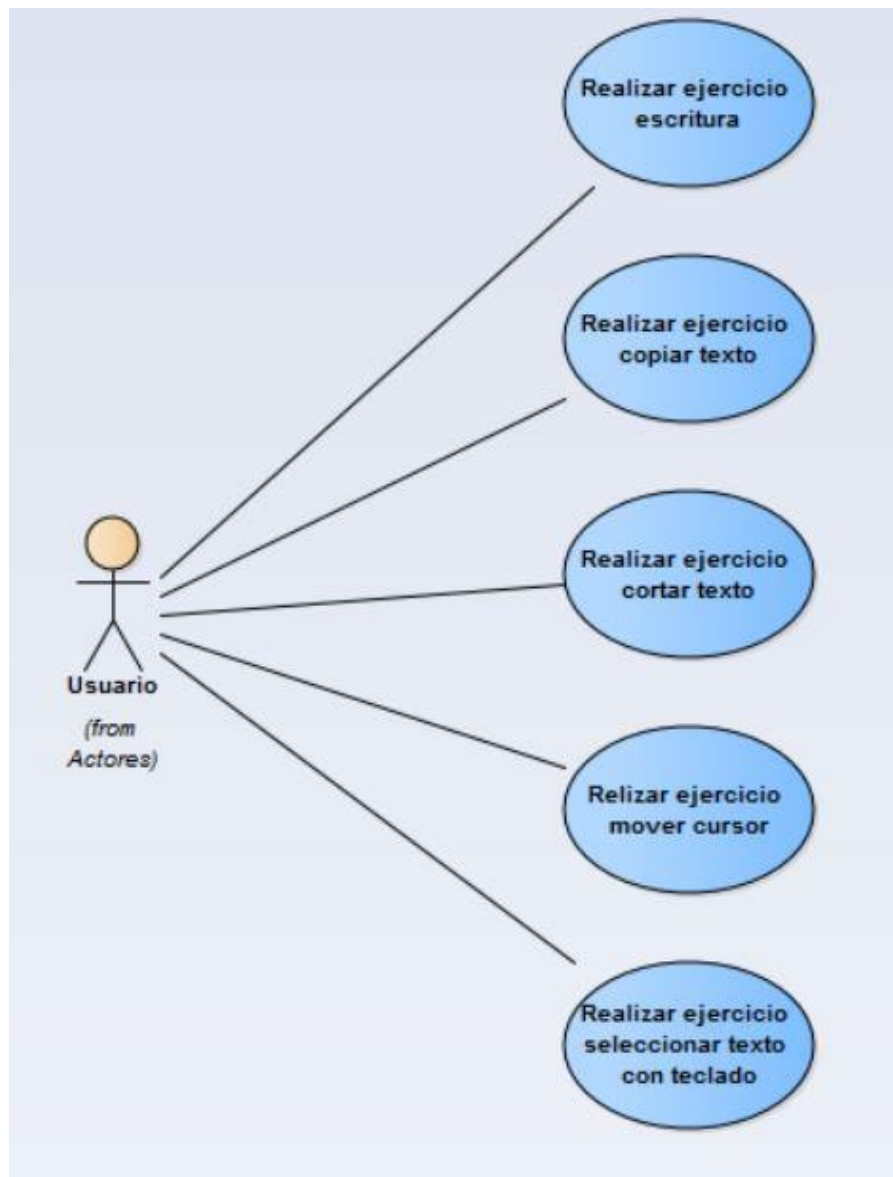
- Teclado: donde están todos los casos de uso que implementan la funcionalidad de los ejercicios del teclado.

- Ratón: donde se encuentran los casos de uso que implementan la funcionalidad de los ejercicios del ratón.
- Carpetas: casos de uso que implementan la funcionalidad de los ejercicios de carpetas
- Visualización: en este paquete se encuentran los casos de uso en los que su funcionalidad tiene algo que ver con visualizar algo por pantalla y no pertenece a ninguno de los paquetes anteriores

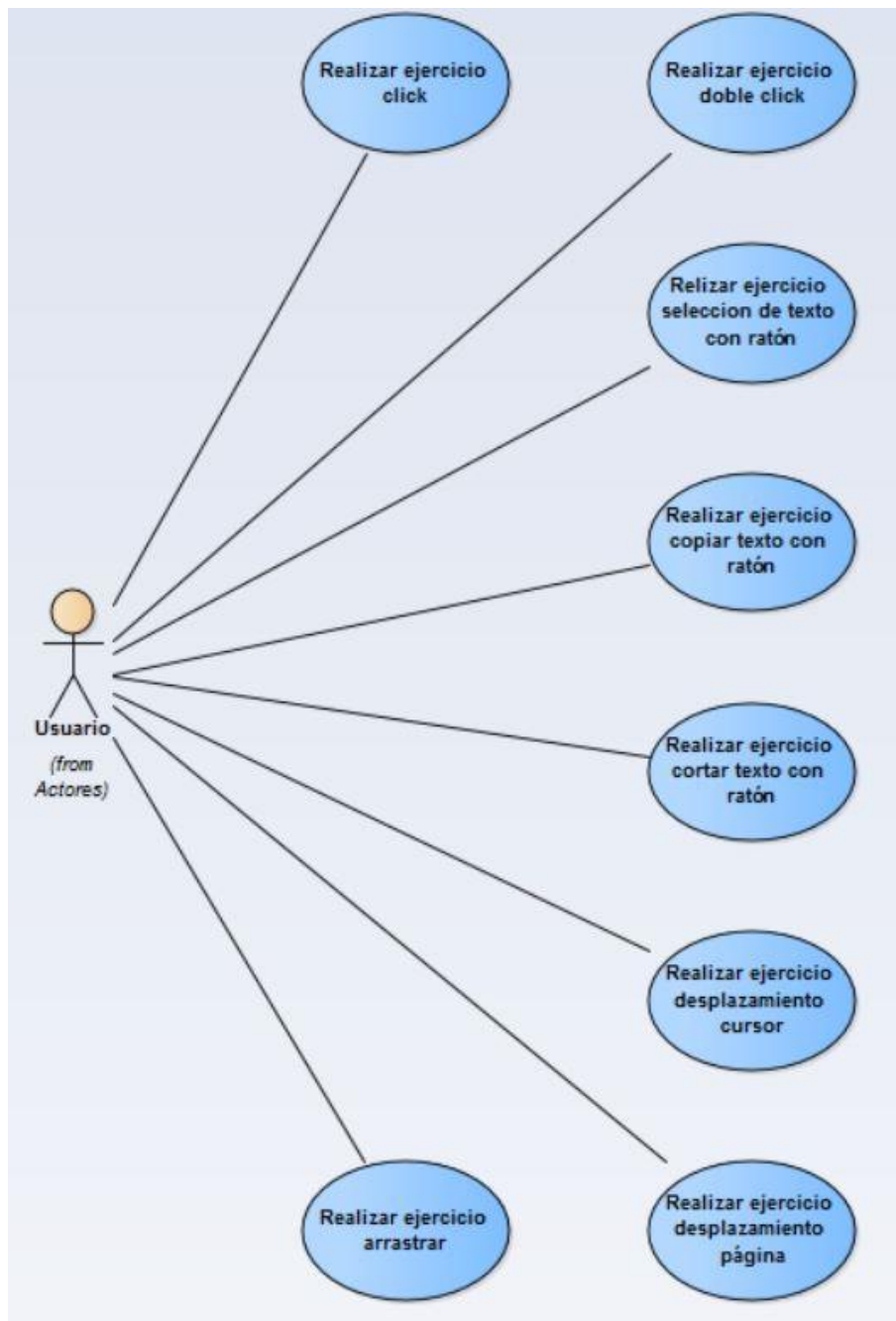


A continuación, se muestran los casos de uso de cada paquete relacionados con el actor Usuario.

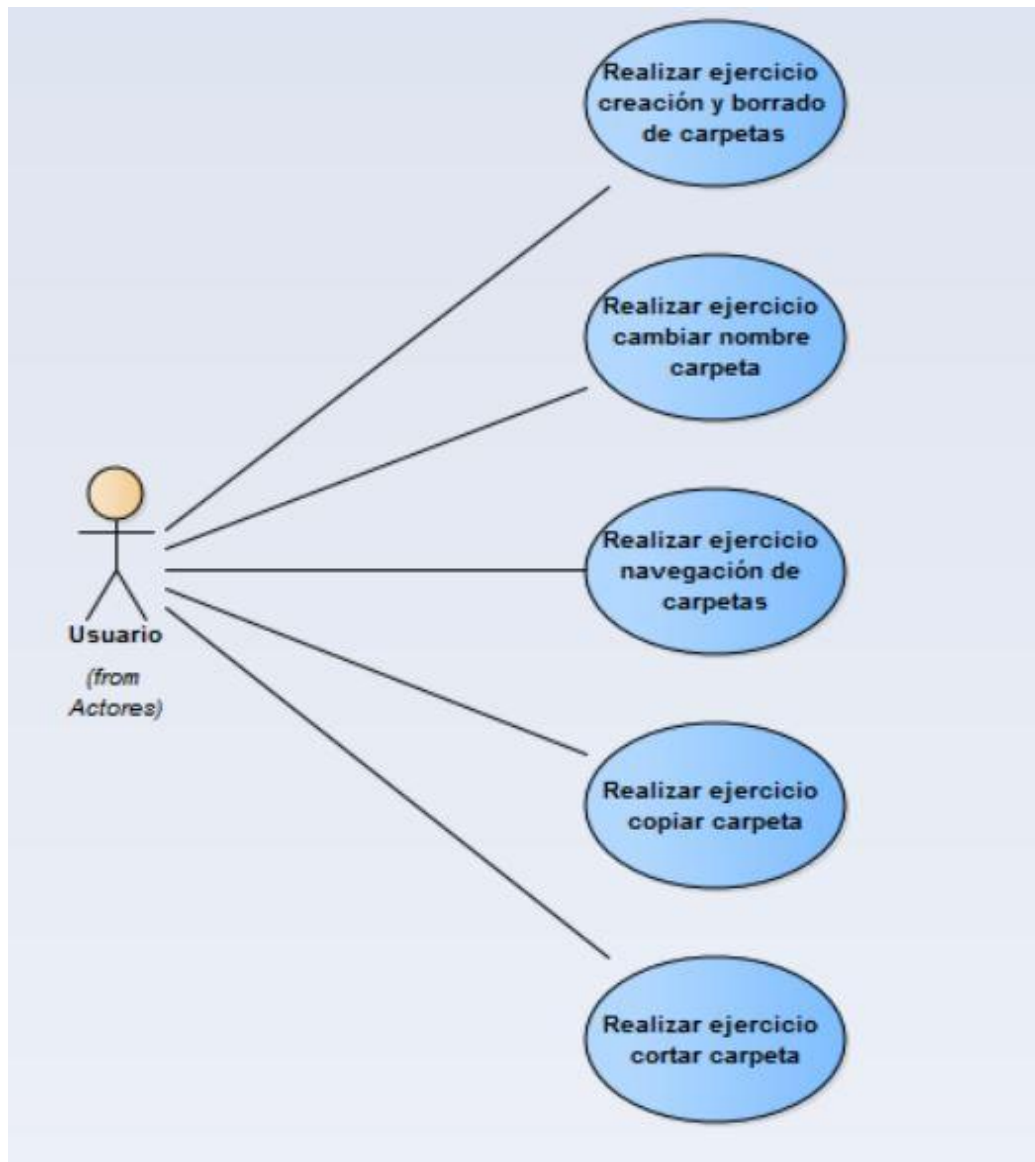
Paquete Teclado:



Paquete Ratón:



Paquete Carpetas:



Paquete Visualización:

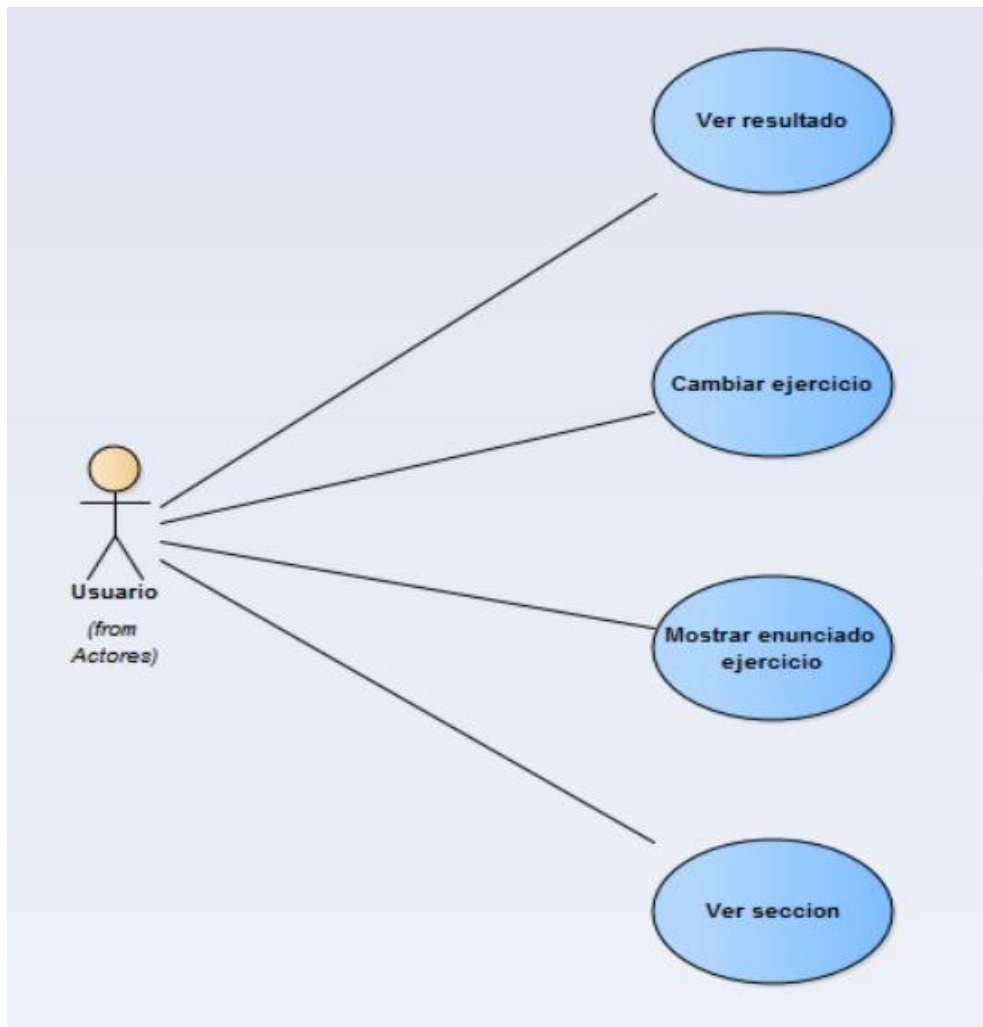


Diagrama de navegabilidad:

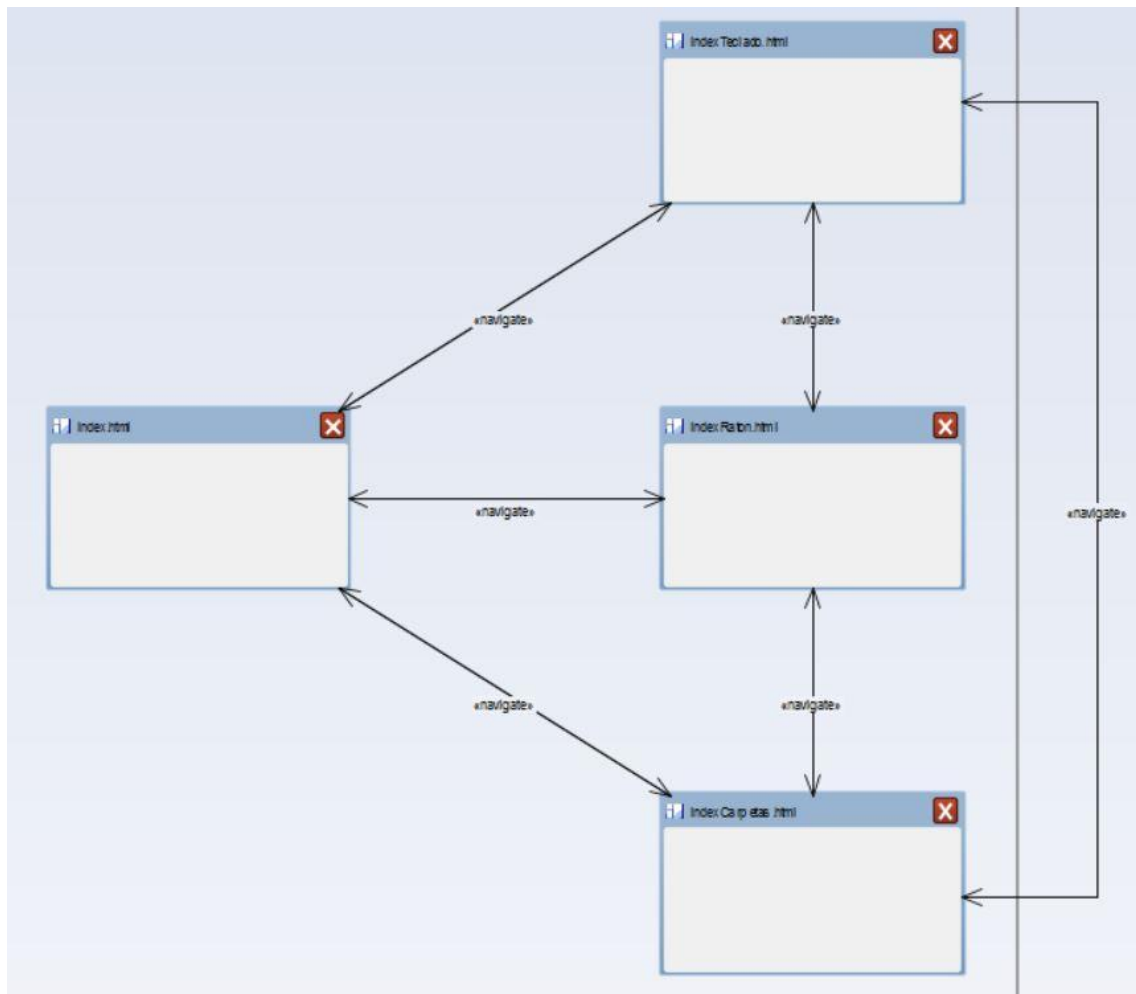
En el diagrama de navegabilidad se muestran cómo va es la navegación entre páginas HTML de nuestra aplicación (las pantallas que hay y desde donde se puede acceder a cada una). Vamos a tener 4 ventanas:

- Index.html: es la pantalla principal a través del menú podemos acceder a las distintas secciones
- indexTeclado.html: es la pantalla de teclado donde podemos ver teoría del teclado y acceder a los distintos ejercicios de la parte de teclado. Además, a través del menú principal podemos acceder a las distintas secciones incluida la pantalla principal.
- IndexRaton.html: es la pantalla de ratón donde podemos ver teoría del ratón y acceder a los distintos ejercicios que conforman esta



sección. A través del menú principal podemos acceder a las distintas secciones incluida la pantalla principal.

- IndexCarpetas.html: es la pantalla de carpetas donde podemos ver teoría de las carpetas y acceder a los distintos ejercicios que conforman esta sección. A través del menú principal podemos acceder a las distintas secciones incluida la pantalla principal.



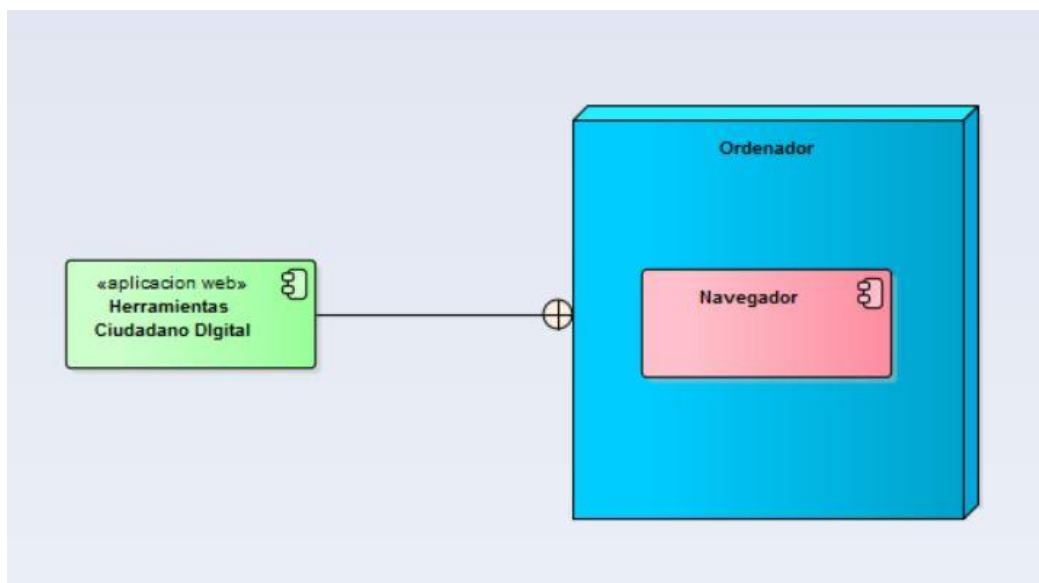
Los ejercicios de cada sección no aparecen en el diagrama de navegabilidad ya que se cargan con el método load de JavaScript dentro de la página HTML que corresponde a su sección.

## Vista de desarrollo:

La vista de desarrollo nos permite describir los componentes del sistema. Para describir esta vista he realizado los siguientes diagramas:

- Diagrama de componentes
- Trazabilidad
- Estructura de carpetas

## Diagrama de componentes:

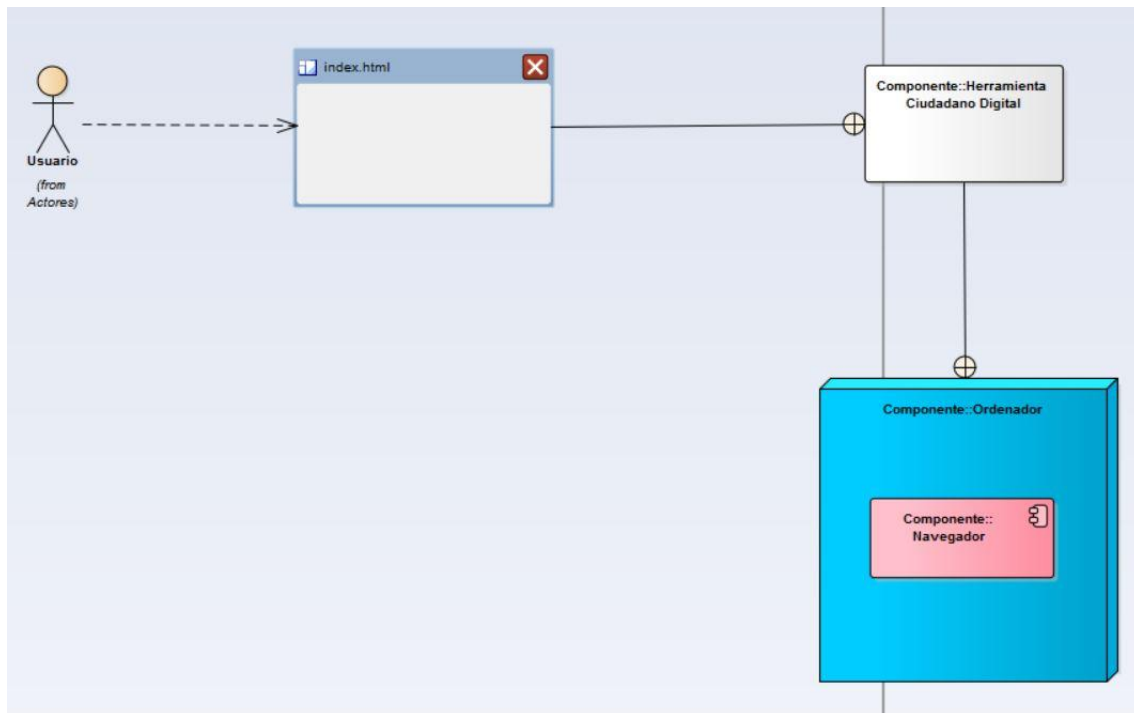


En el que tenemos la aplicación Herramientas Ciudadano Digital que deberá estar almacenado en un ordenador que posea un navegador para poder ejecutar la aplicación.

Los distintos archivos (CSS, JS y HTML) que conforman la aplicación se mostrara en detalle en la vista lógica.

## Trazabilidad

En este diagrama podemos ver como interactúa el usuario con los componentes del sistema.



Tenemos el actor Usuario que depende de la página principal para poder iniciar la aplicación que está contenida en la aplicación HCD que a su vez debe estar en un ordenador que tenga un navegador.

#### Estructura de carpetas:

En este diagrama podemos observar la estructura de carpetas donde están almacenados los distintos archivos que forman la aplicación.

Carpeta: almacena los archivos HTML que pertenecen a la sección carpetas.

Imágenes: contiene las imágenes que necesita la aplicación para su correcto funcionamiento, dentro de ella está la carpeta iconos que contiene los iconos que utiliza la aplicación.

CSS: contiene los archivos de estilo CSS de la aplicación, dentro tenemos las carpetas:

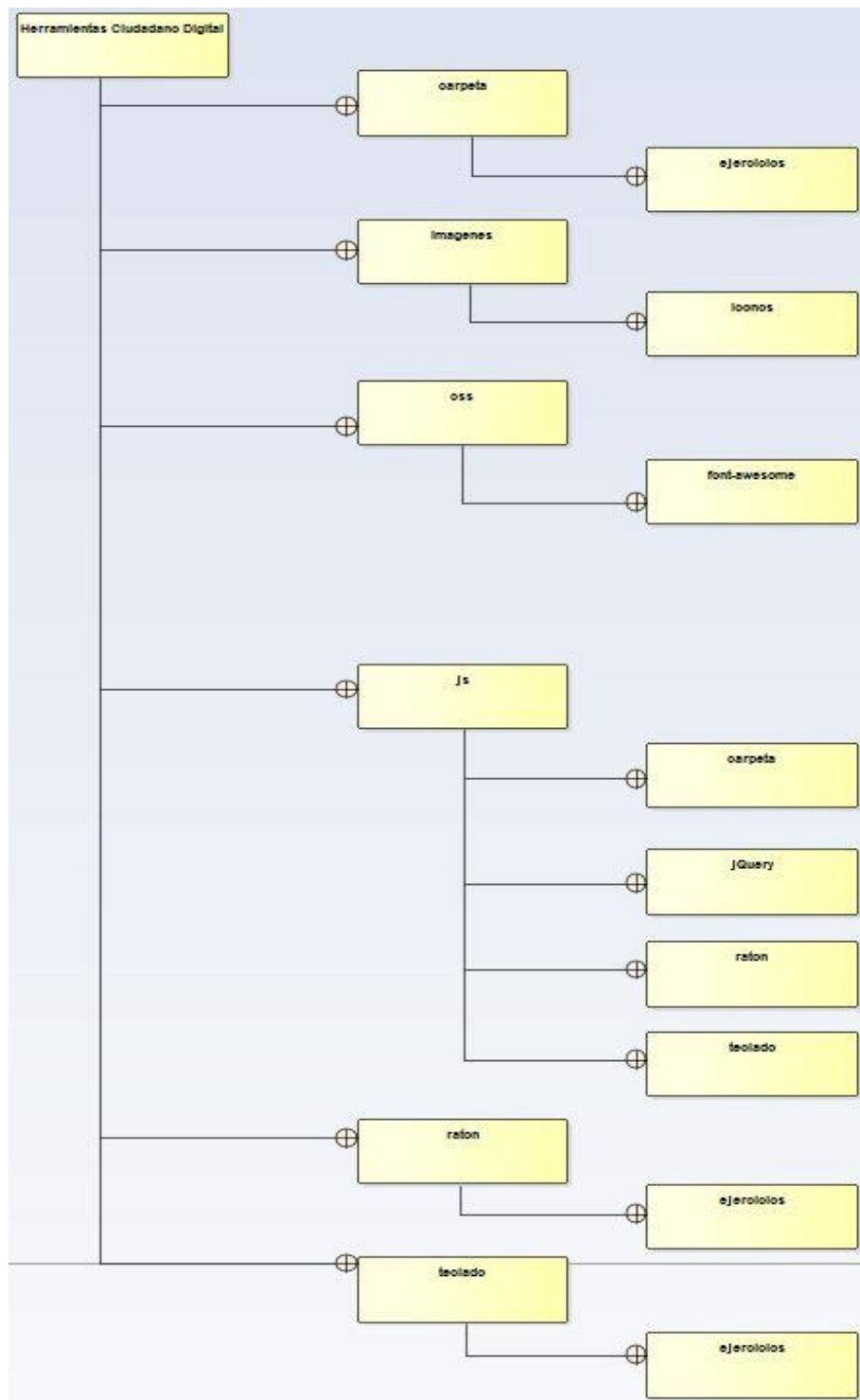
- Font-awesome: almacena los archivos de la tipografía de la aplicación.

JS: almacena los archivos JavaScript que necesita la aplicación para su correcto funcionamiento dentro de ella encontramos las carpetas teclado,

ratón y carpetas que incluyen los archivos JS vinculados a cada una de las secciones y la carpeta JQuery que contiene los archivos JS para el correcto funcionamiento de JQuery.

Ratón: almacena los archivos HTML asociados a la sección ratón.

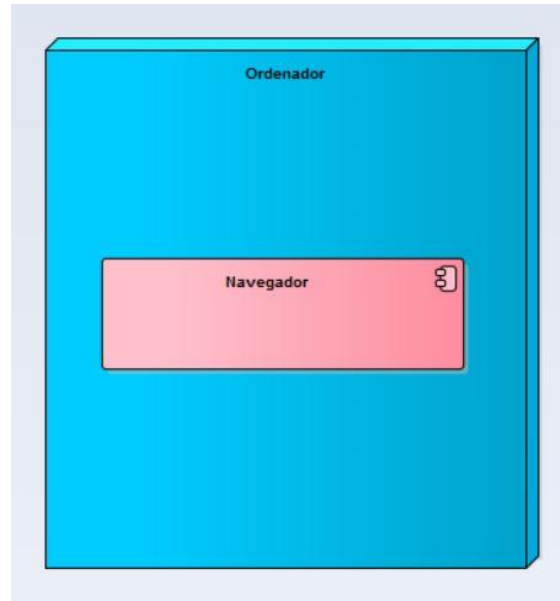
Teclado: almacena los archivos HTML asociados a la sección teclado.



### Vista física:

La vista física nos permite representar las conexiones físicas entre componentes. Para representarla he utilizado el diagrama de Despliegue.

### Diagrama de Despliegue:



El diagrama de despliegue está formado por el único componente físico que tiene nuestro sistema, ya que HDC se ejecuta en modo local sin necesidad de conectarse a otros servidores ni a Internet. Debido a esto el componente físico que necesitamos es el ordenador, donde se ejecutará la aplicación web, que deberá tener un navegador.

### Vista de proceso

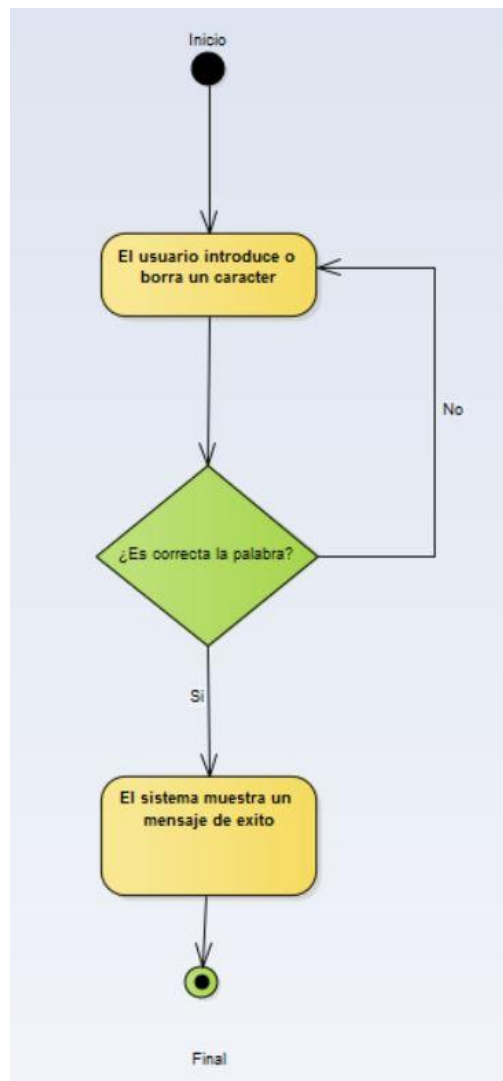
La vista de proceso nos permite representar los procesos del sistema y como se comunican. Para representar esta vista he utilizado el diagrama de actividad.

He paquetizado los diagramas de actividad según la paquetización del caso de uso al que pertenezcan: teclado, carpeta, ratón y visualización.

## Teclado

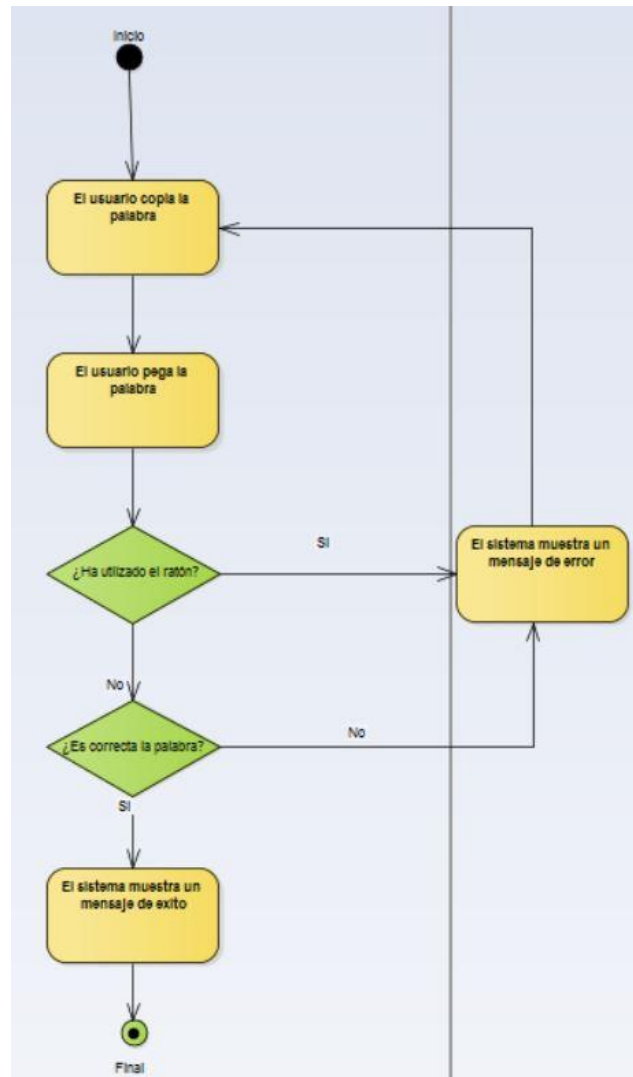
### Realizar ejercicio de escritura

Ejercicio donde aparece una palabra y debemos completarla correctamente.



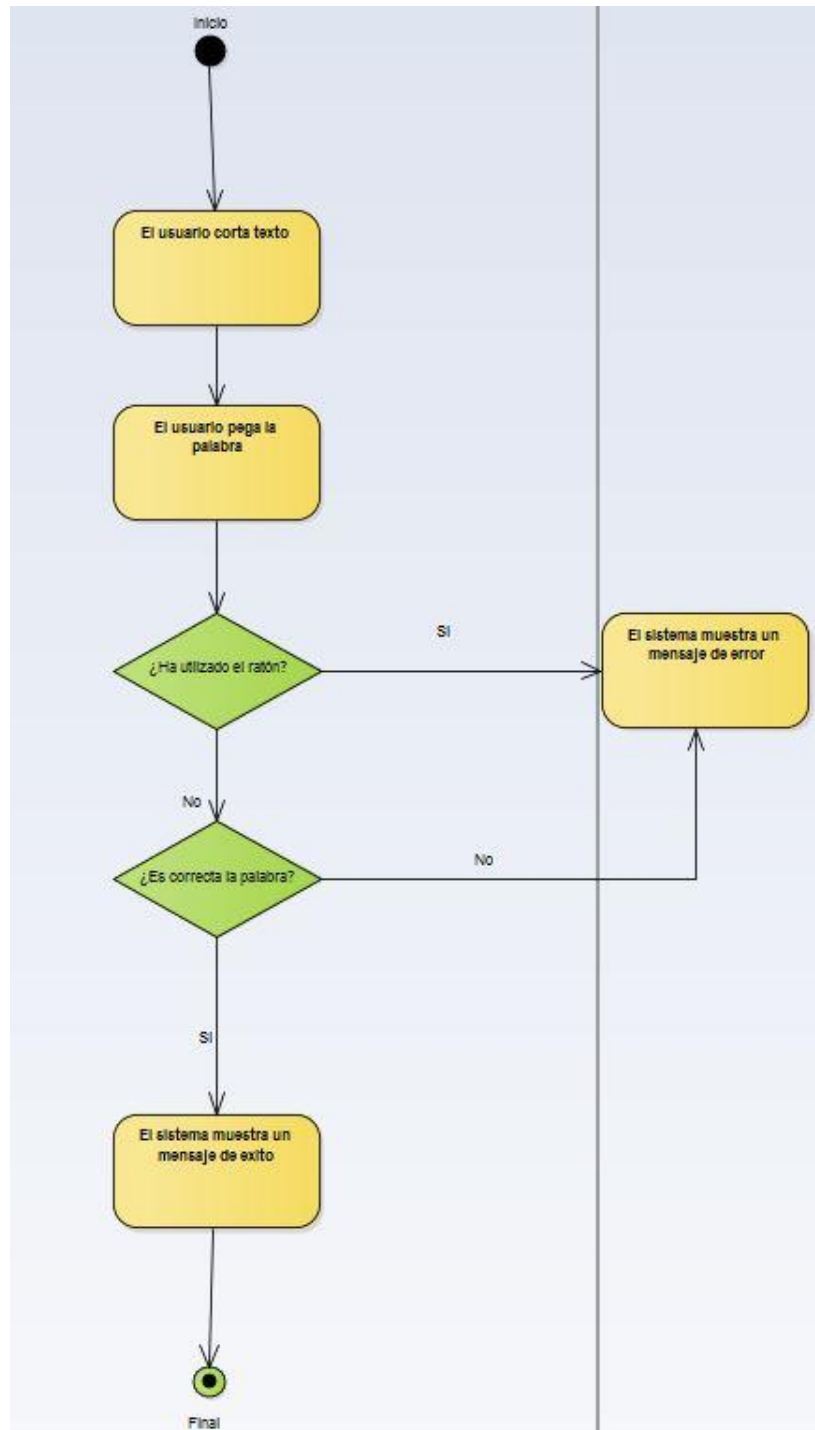
## Realizar ejercicio copiar texto con teclado

Ejercicio donde debemos copiar la palabra que nos indique la aplicación y pegarla en el recuadro correspondiente usando únicamente los controles del teclado.



## Realizar ejercicio de cortar texto con teclado

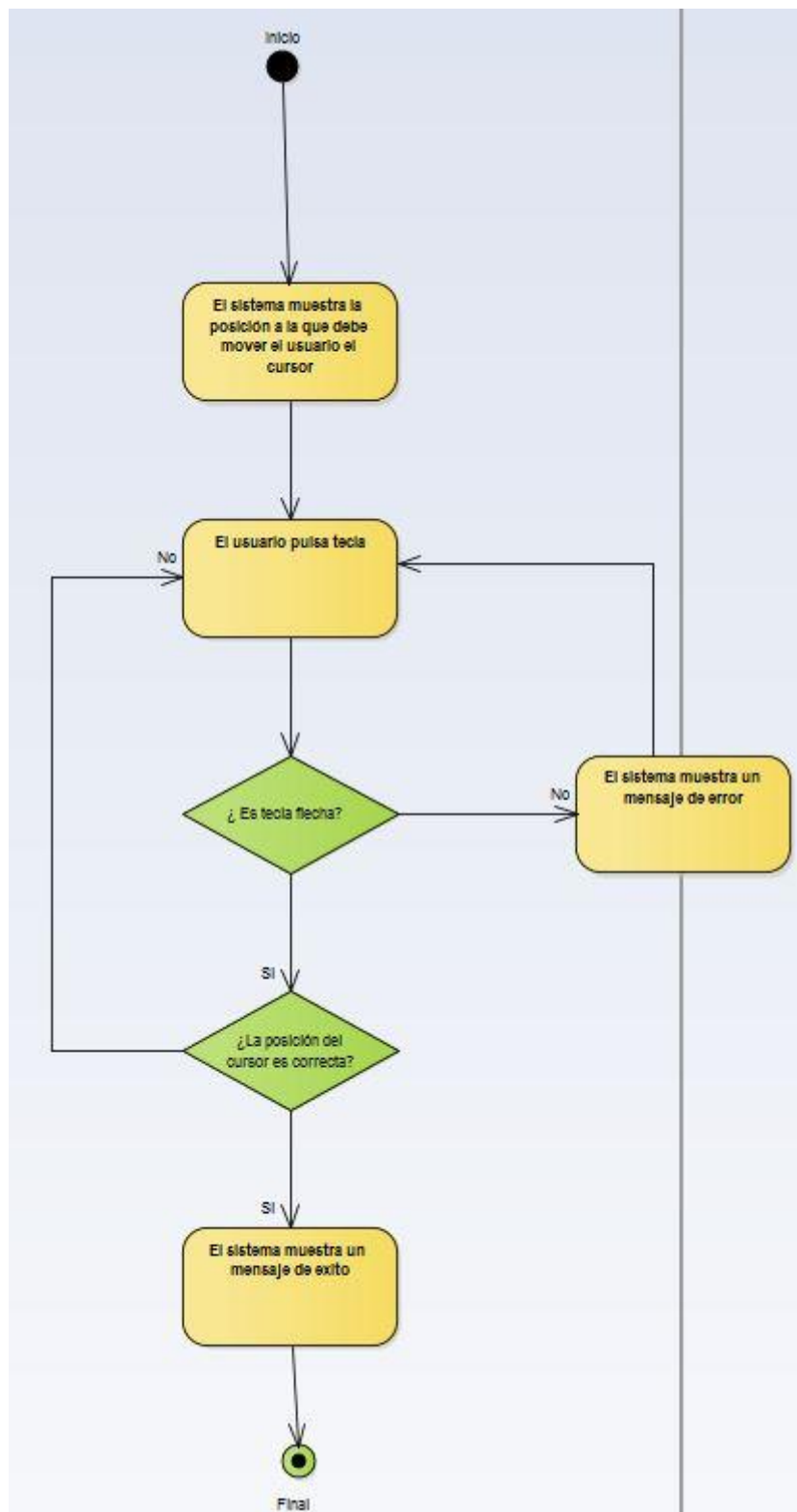
Ejercicio donde debemos copiar la palabra que nos indique la aplicación y pegarla en el recuadro correspondiente usando únicamente los controles del teclado.





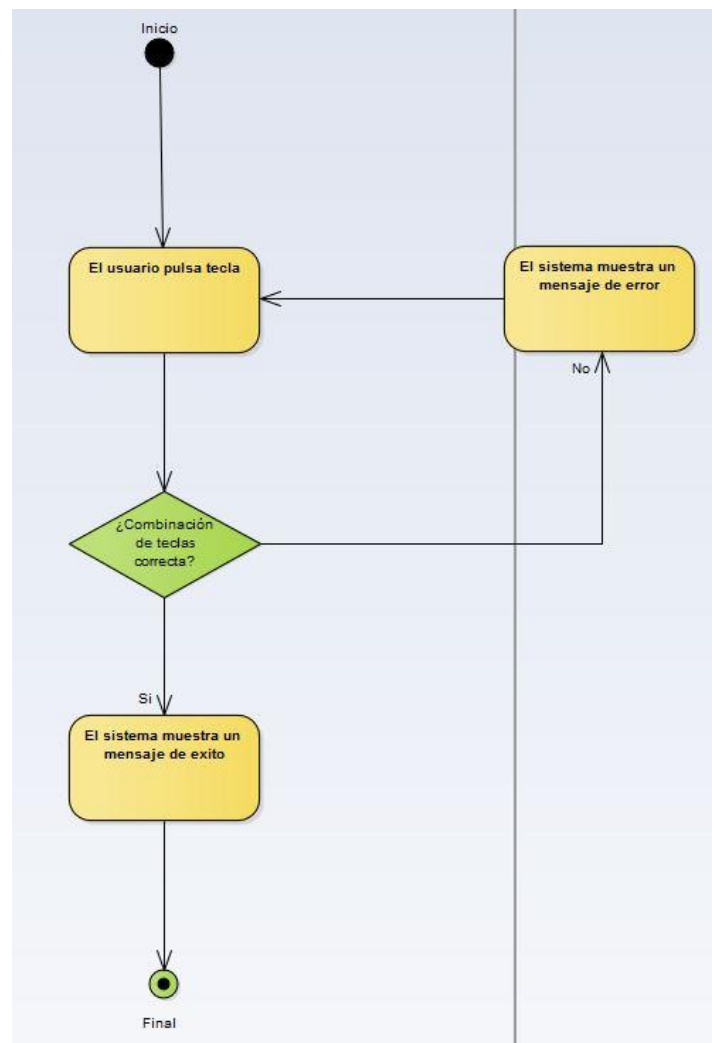
## Realizar ejercicio mover cursor con teclado

Ejercicio donde debemos mover el cursor hasta una posición indicada por la aplicación.



## Realizar ejercicio de seleccionar texto con teclado

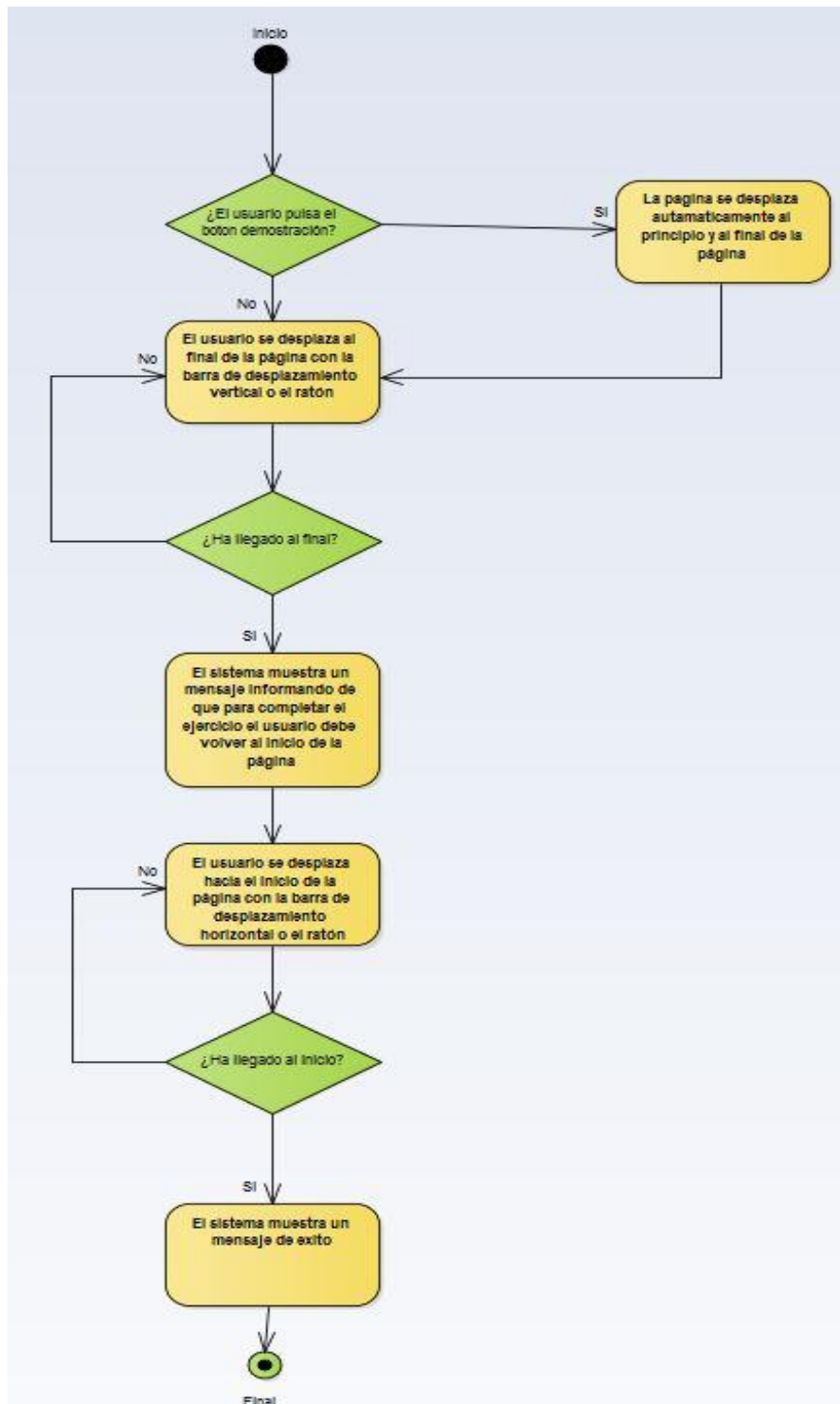
Ejercicio donde deberemos seleccionar un texto las veces indicadas por la aplicación.



Ratón:

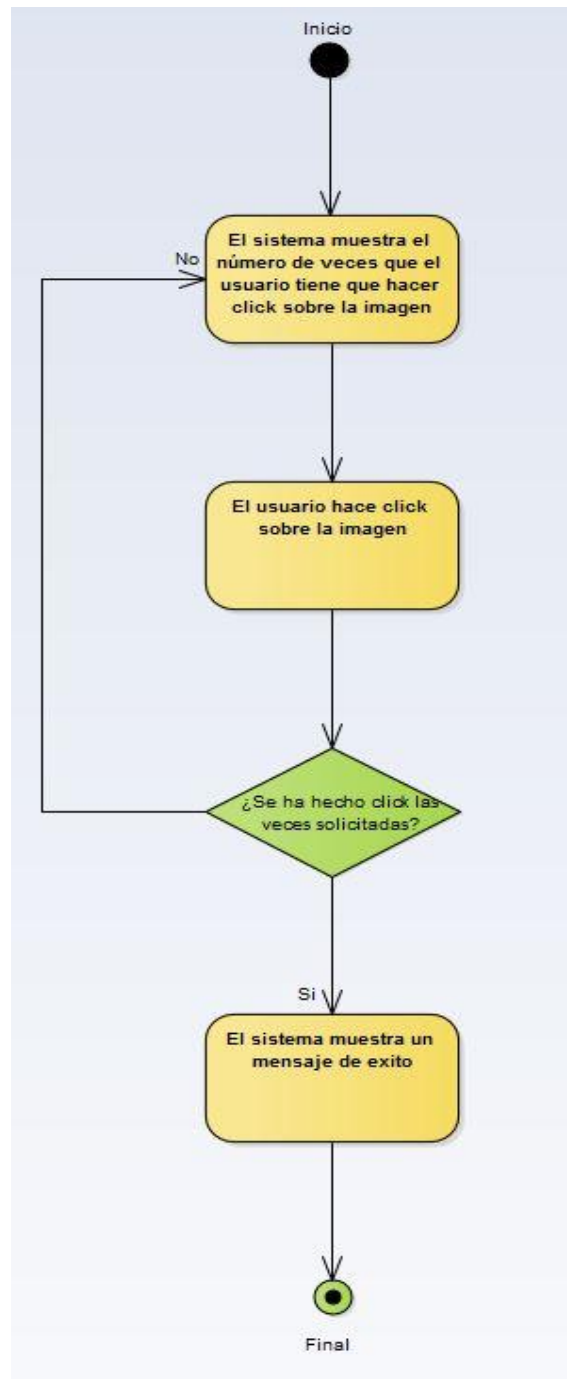
### Realizar ejercicio desplazamiento con ratón

Ejercicios en los que debemos desplazarnos por la página utilizando en uno de ellos la barra de desplazamiento vertical y en el otro la barra de desplazamiento horizontal



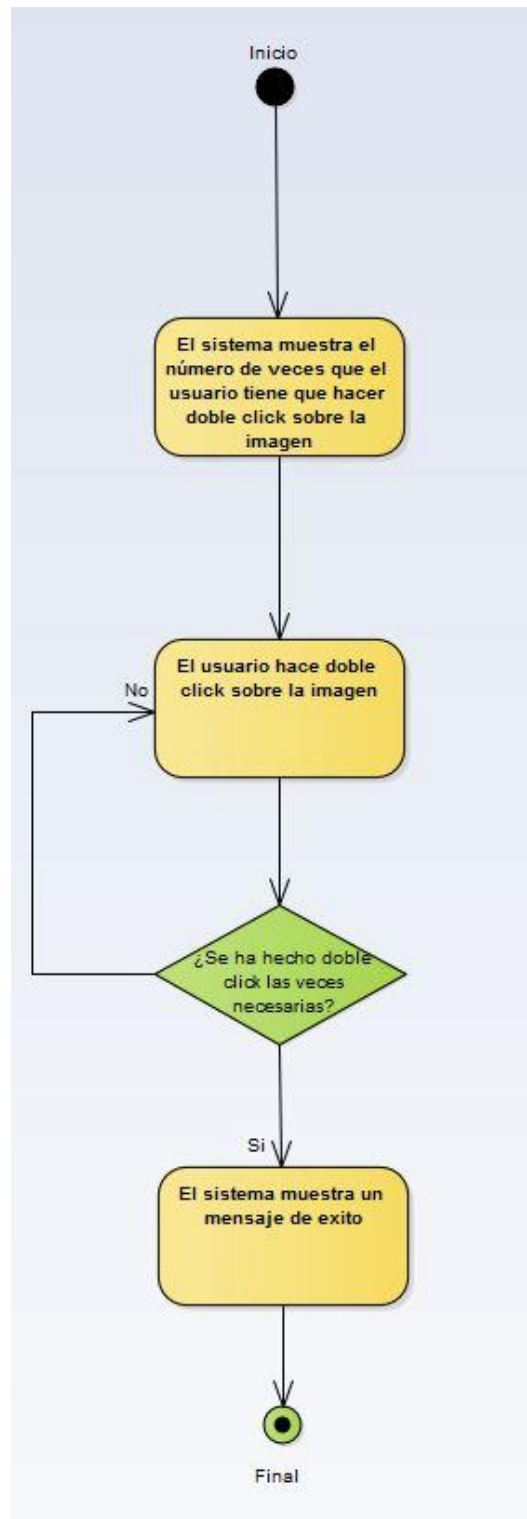
## Realizar ejercicio click

En este ejercicio aparecerá una imagen y deberemos hacer click sobre ella las veces que nos indique la aplicación mientras se mueve.



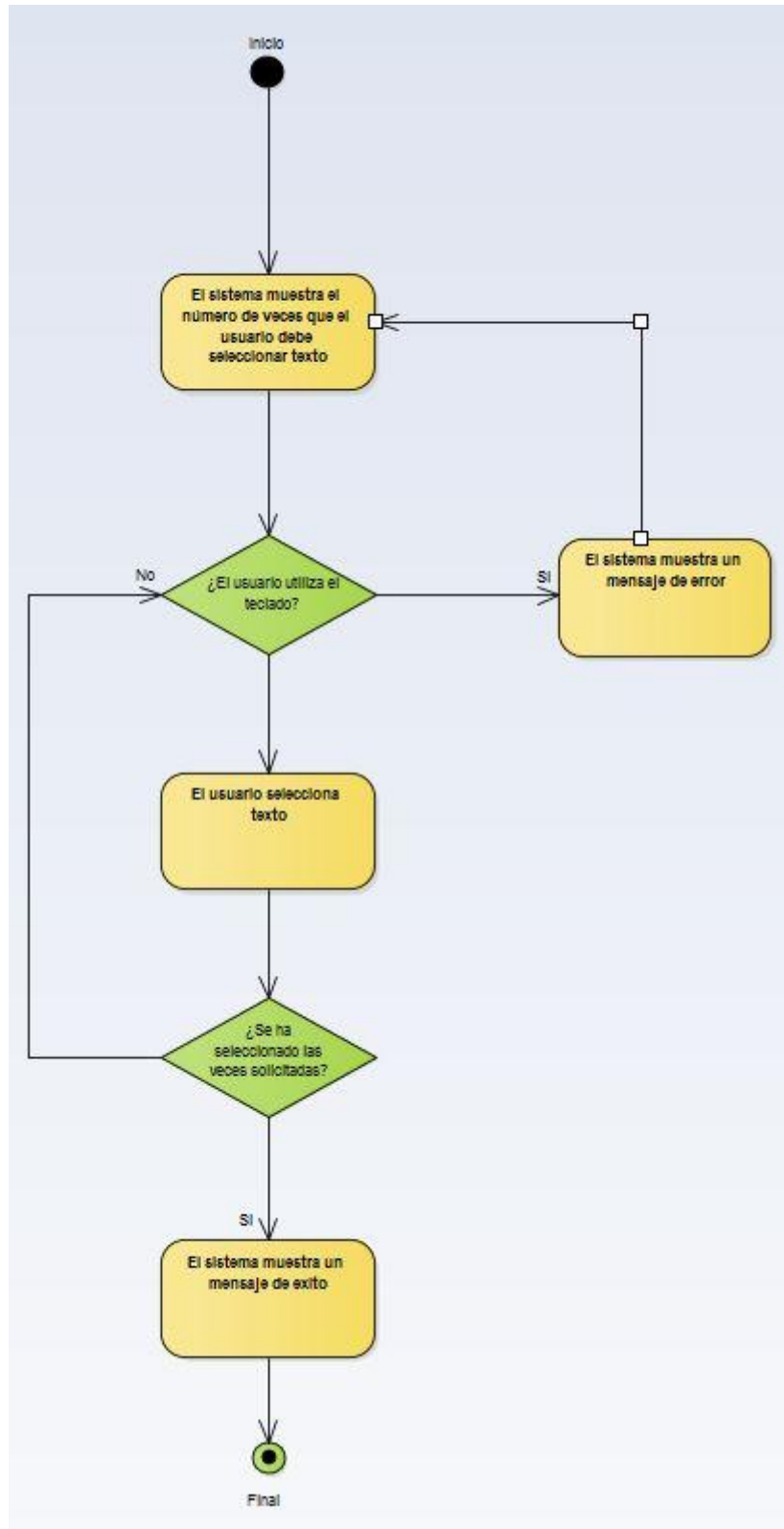
## Realizar ejercicio doble-click

En este ejercicio aparecerá una imagen y deberemos hacer doble click sobre ella las veces que nos indique la aplicación mientras se mueve.



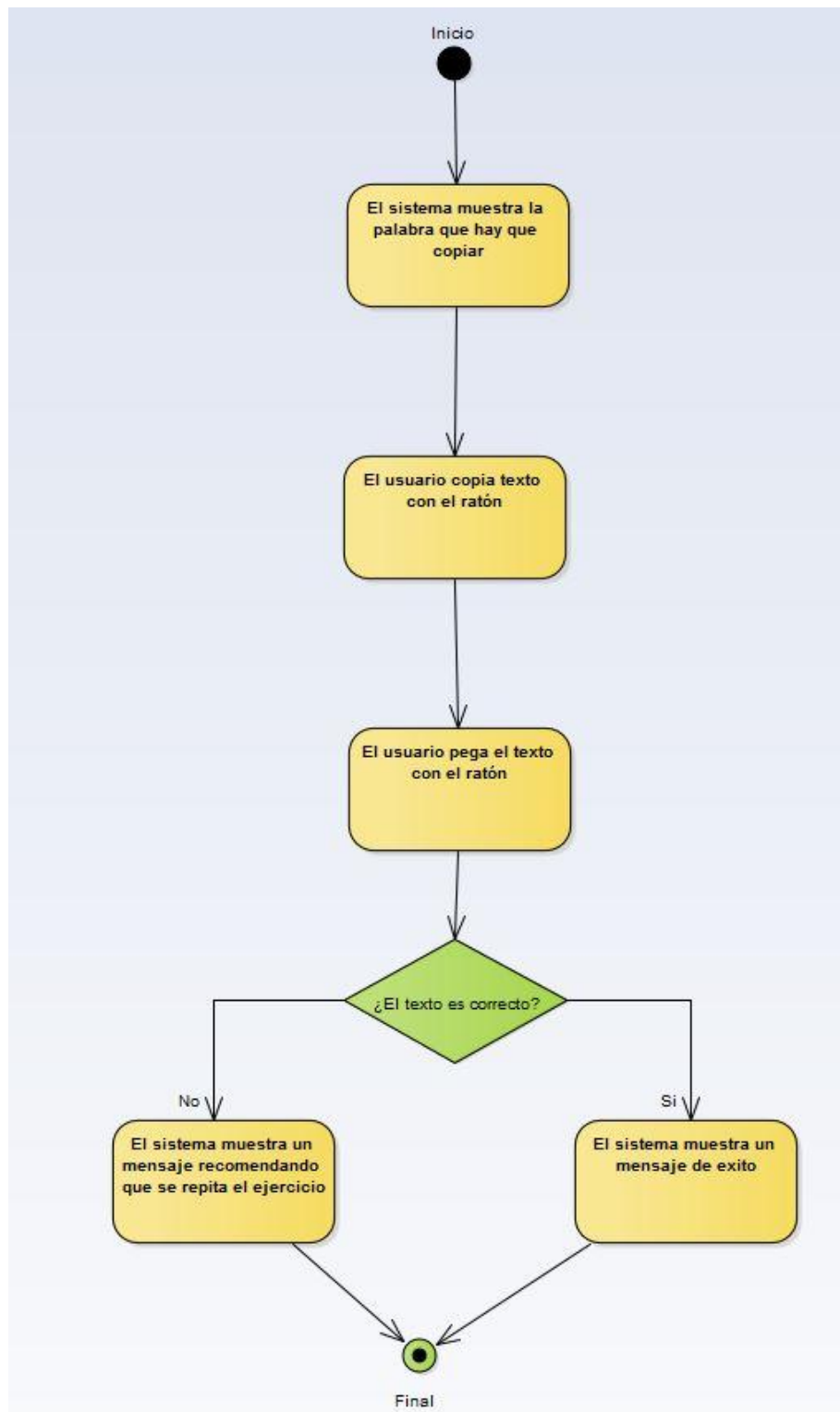
## Realizar ejercicio selección de textos con ratón

En este ejercicio deberemos seleccionar texto las veces que nos indique la aplicación.



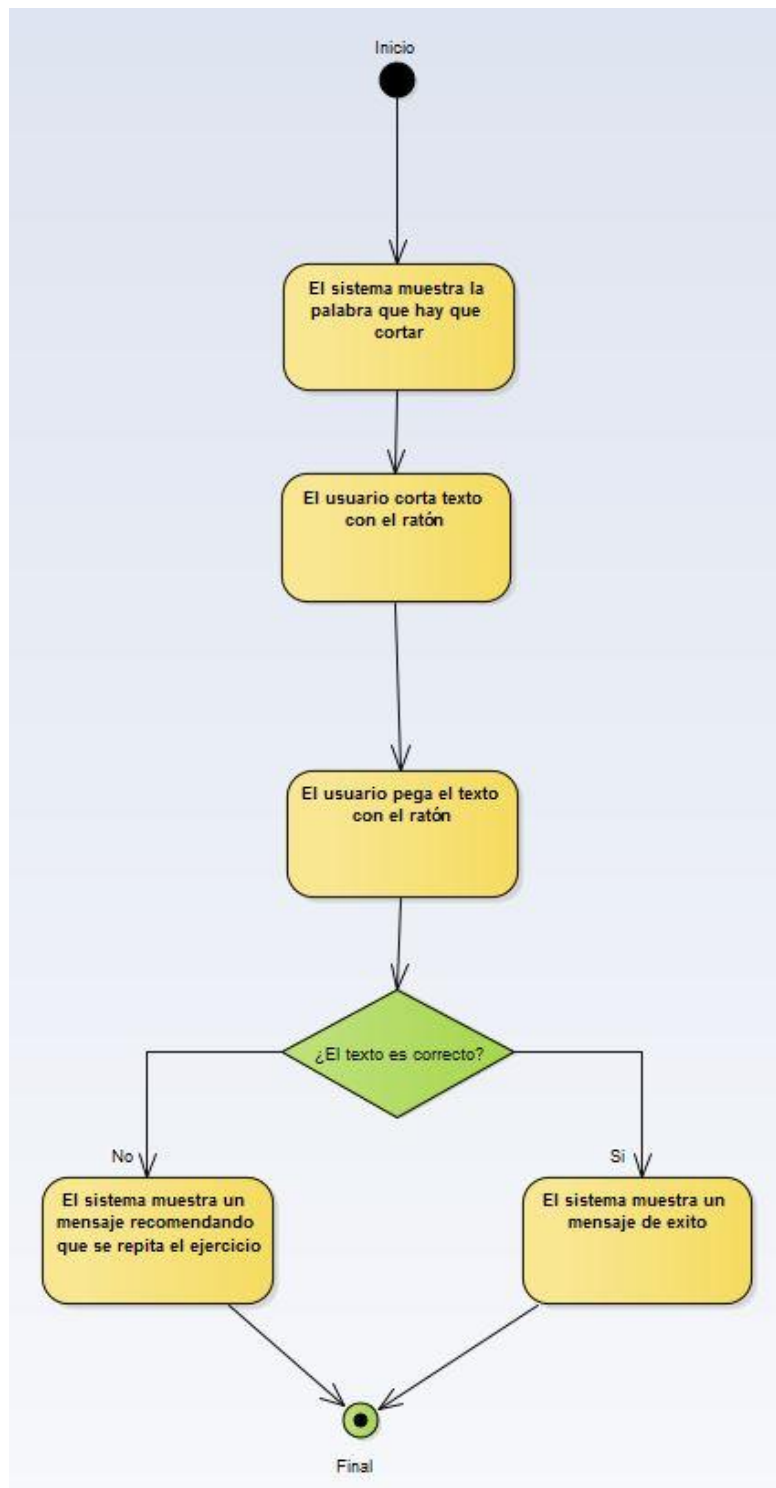
## Realizar ejercicio copiar texto con ratón

En este ejercicio deberemos copiar de una frase una palabra indicada por la aplicación y pegarla en el lugar correspondiente usando únicamente los controles del ratón.



## Realizar ejercicio cortar texto con ratón

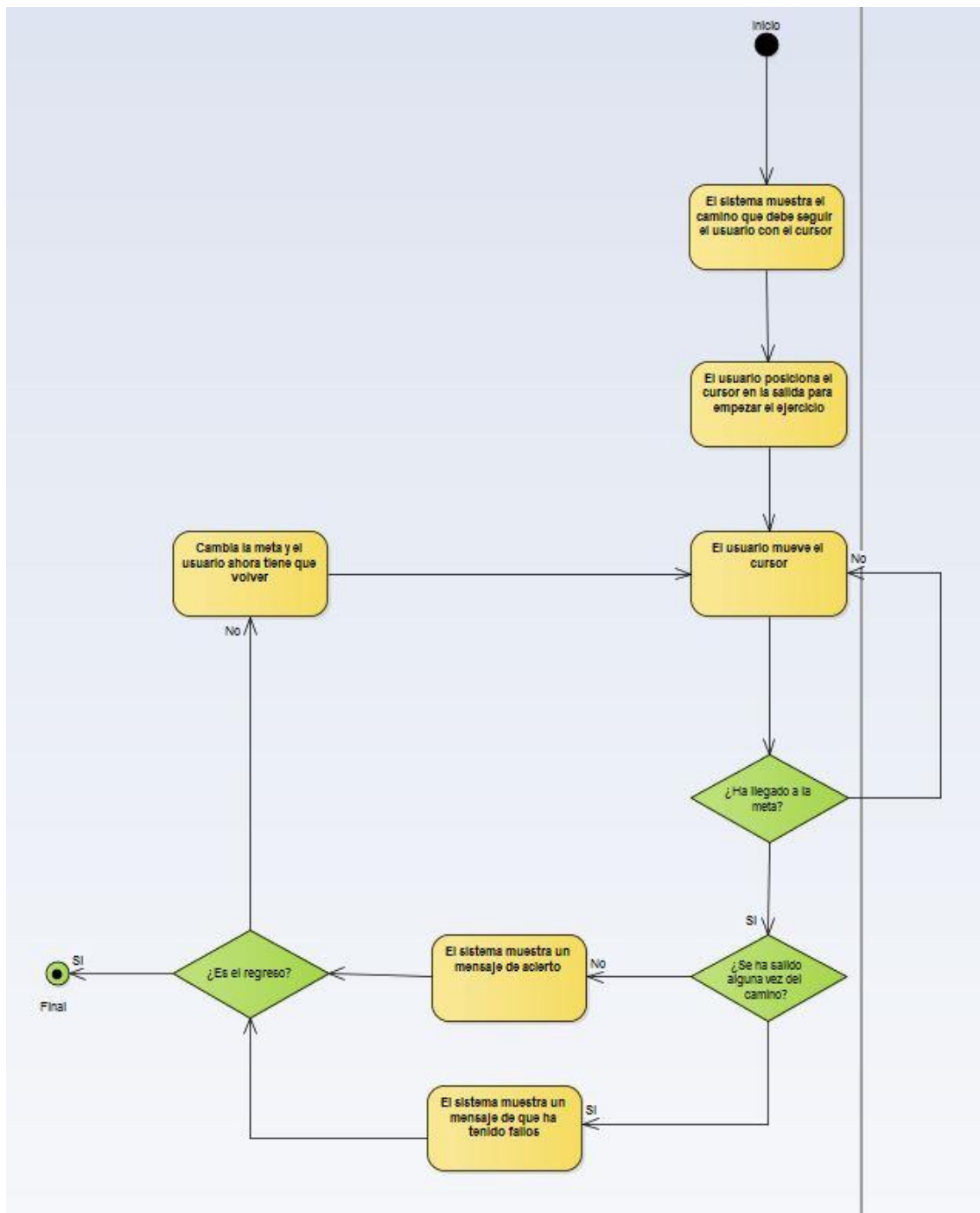
En este ejercicio deberemos cortar de una frase una palabra indicada por la aplicación y pegarla en el lugar correspondiente usando únicamente los controles del ratón.





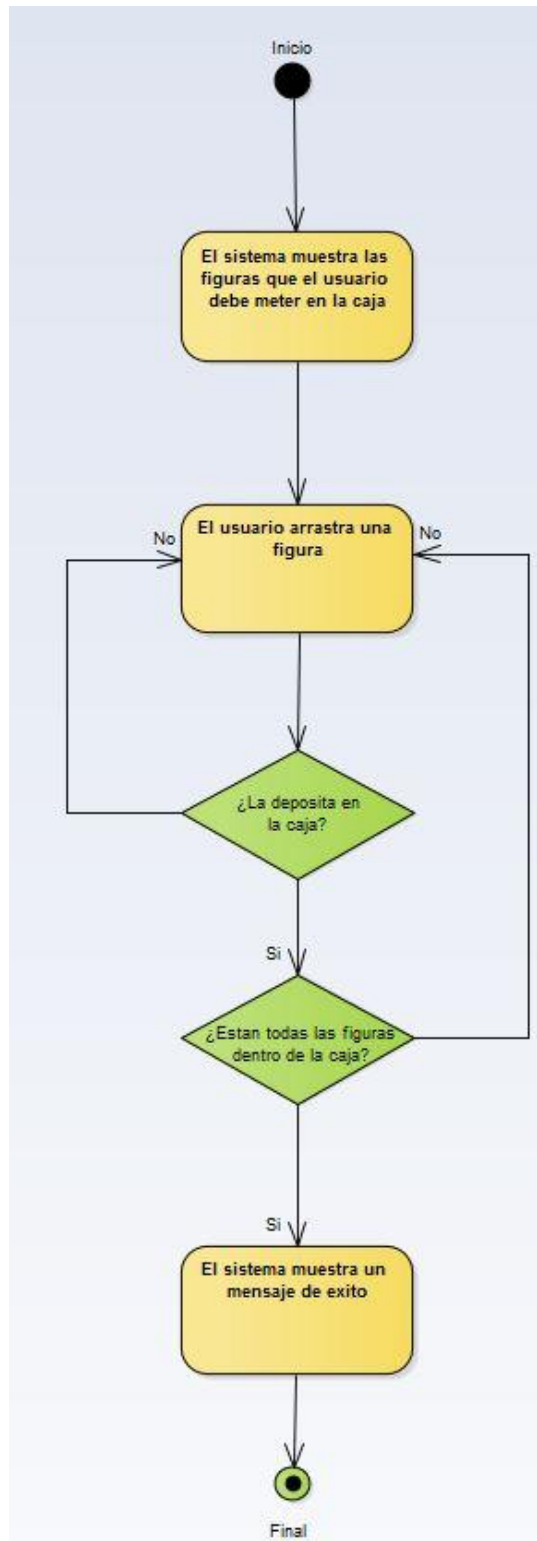
## Realizar ejercicio desplazamiento cursor con ratón

Este ejercicio consiste en desplazar el cursor por un circuito que aparece sin salirse, está compuesto por dos ejercicios el primero de desplazamiento vertical y el segundo que mezcla desplazamiento vertical y horizontal.



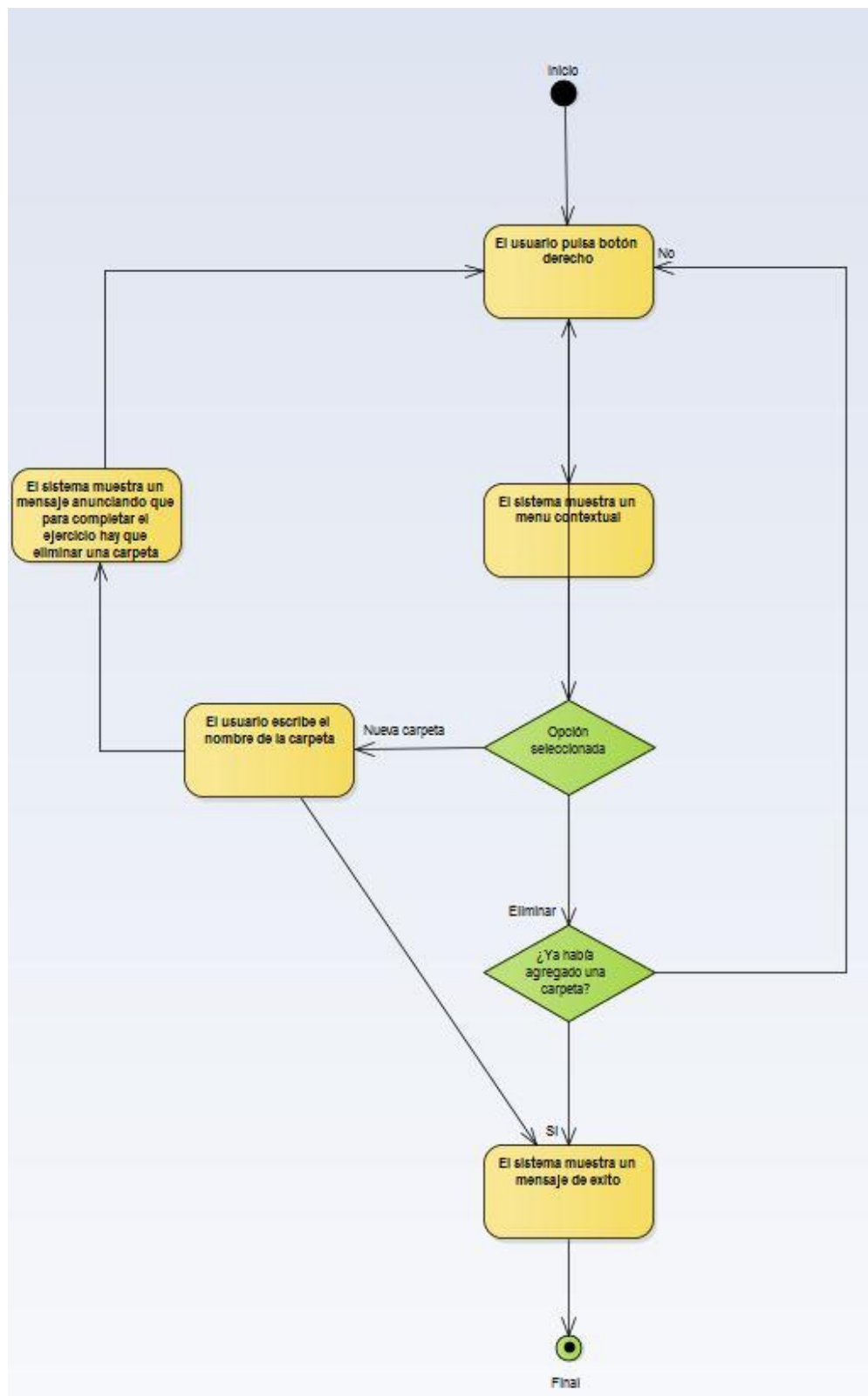
## Realizar ejercicio de arrastrar

Este ejercicio consiste en arrastrar una serie de figuras (cuadrados) a su lugar correspondiente.



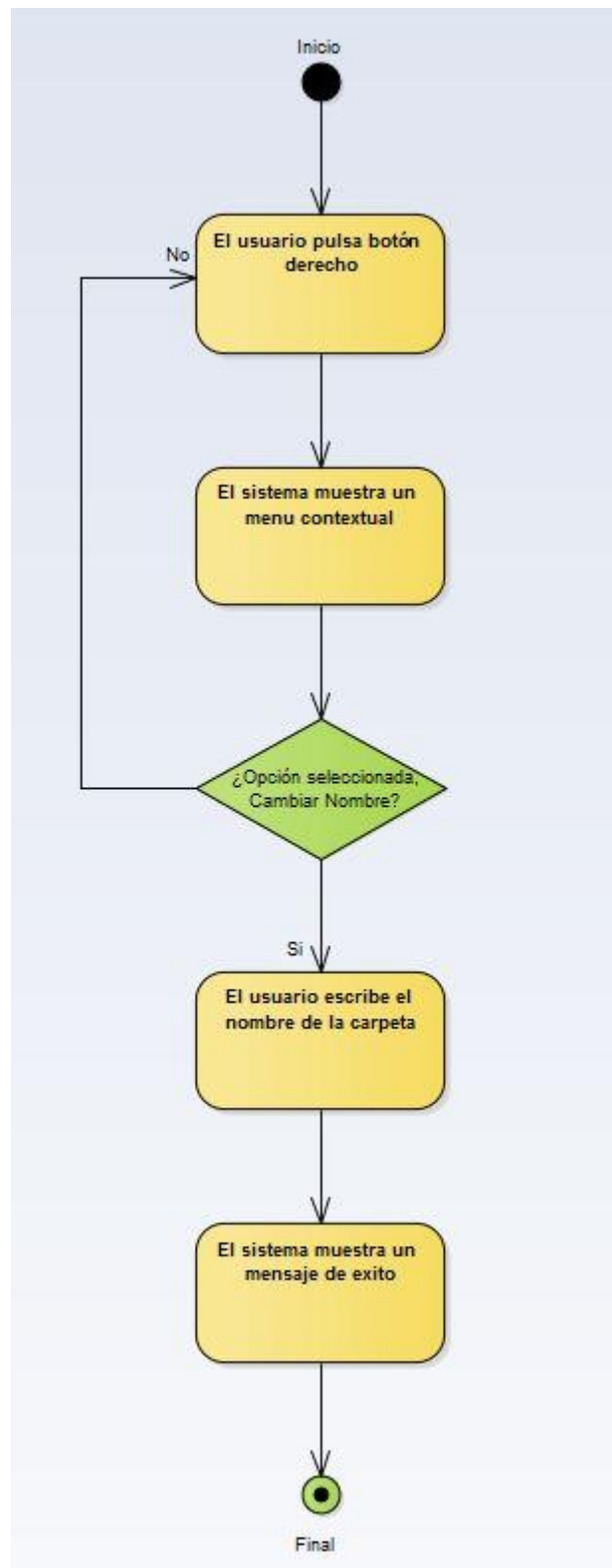
## Realizar ejercicio creación y borrado de carpetas

Este ejercicio consiste en crear una carpeta y después borrar una carpeta o un archivo que deseemos.



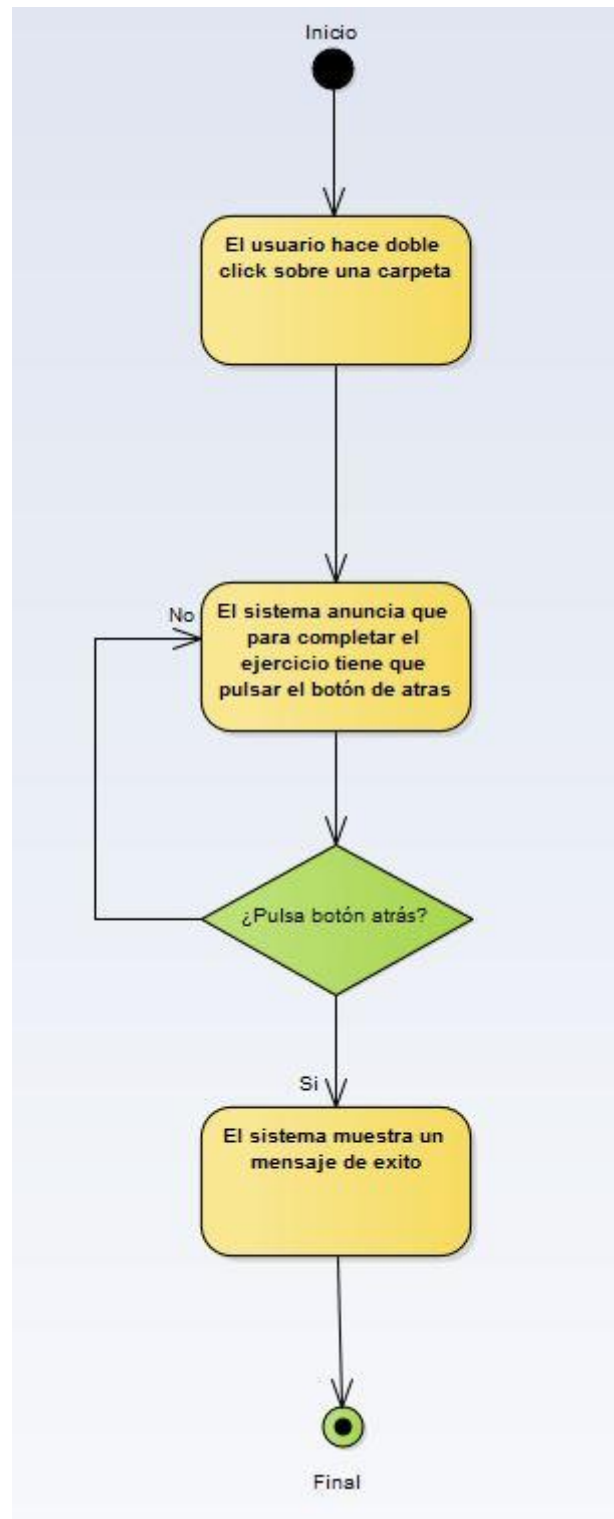
## Realizar ejercicio cambiar nombre de carpeta

El ejercicio consiste en cambiar el nombre a alguna carpeta o archivo.



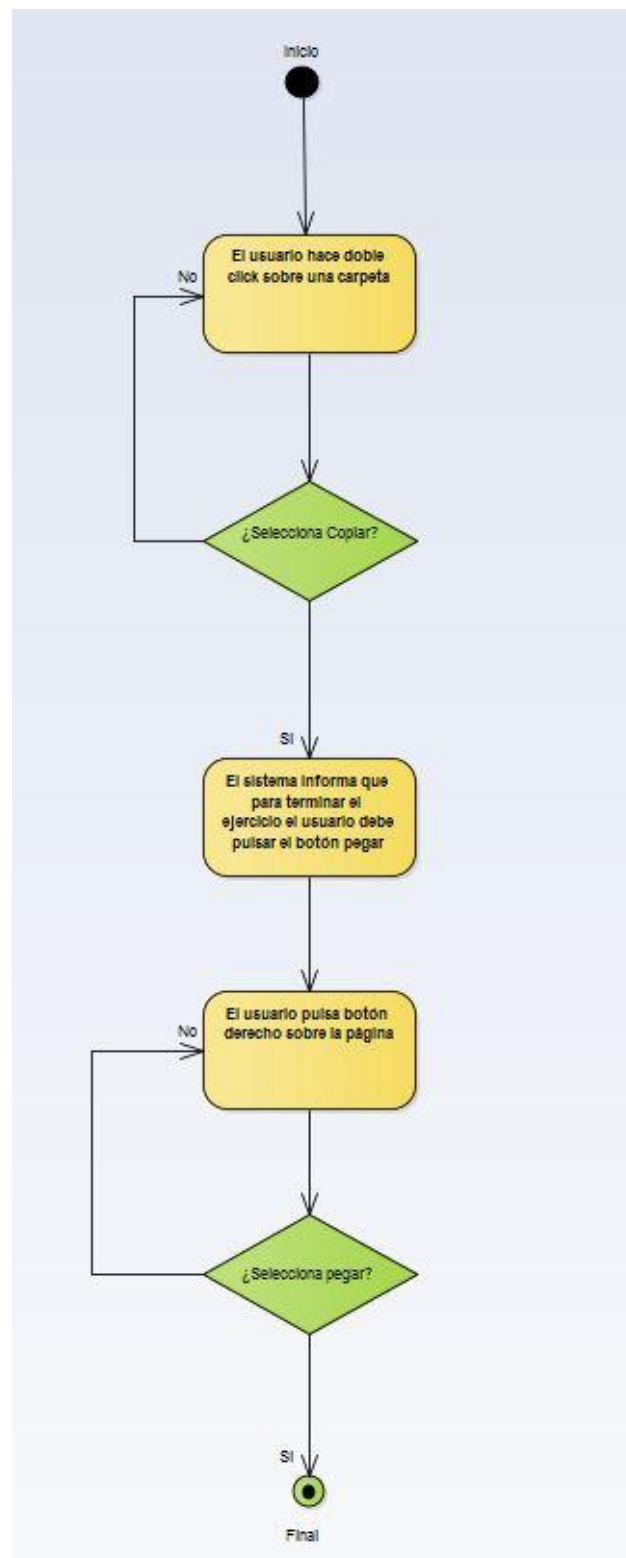
## Realizar ejercicio navegación de carpetas

El ejercicio consiste en entrar alguna carpeta para ver su contenido y volver a la ruta anterior de la que veníamos.



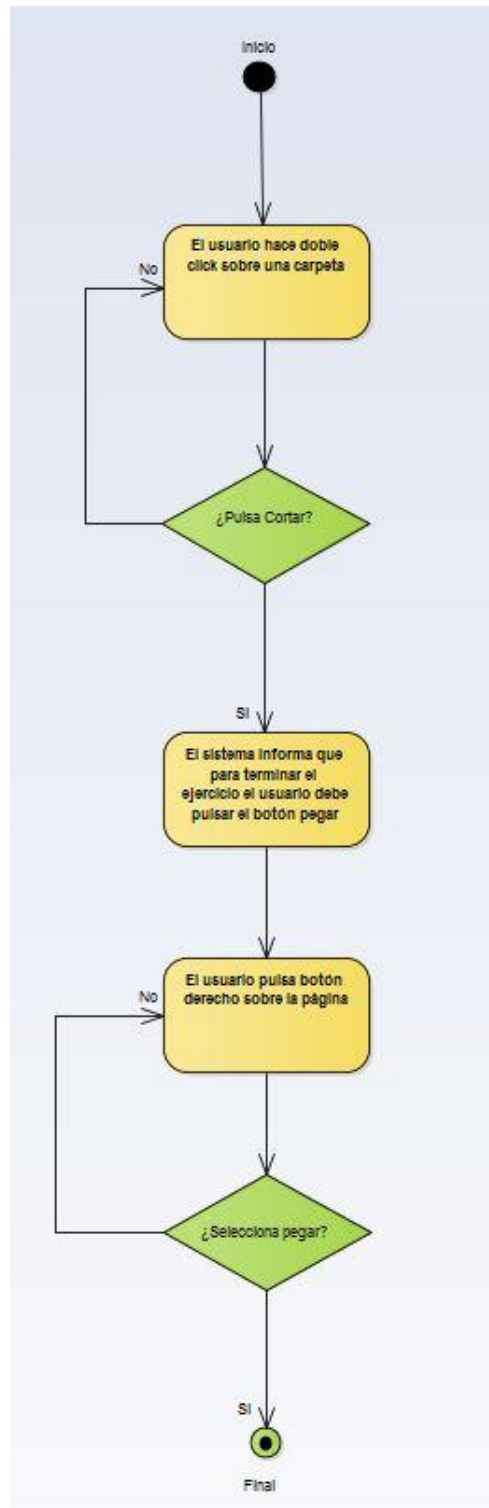
## Realizar ejercicios copiar carpeta

Este ejercicio consiste en copiar una carpeta o archivo que deseemos y pegarlo en otra ruta.



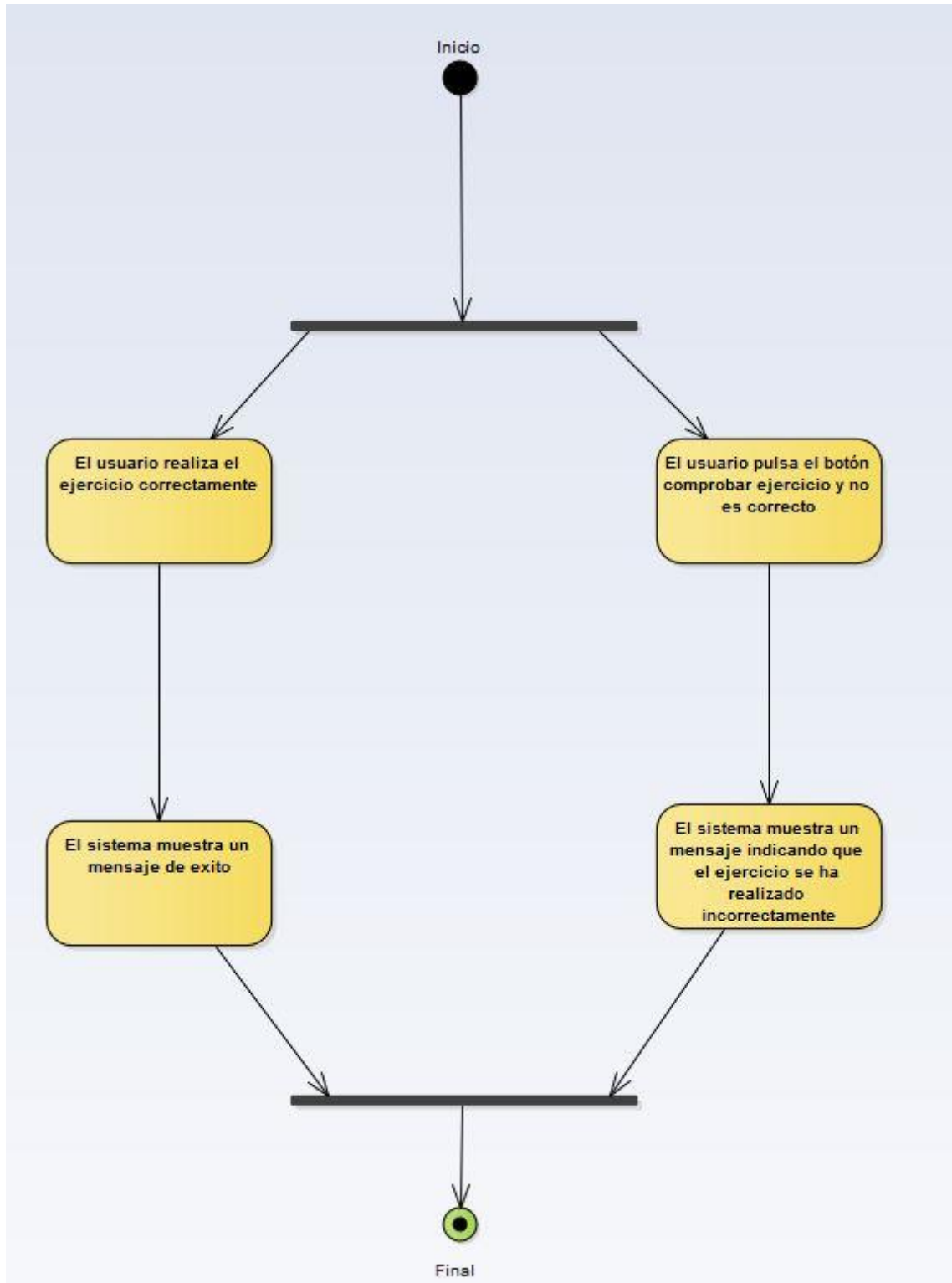
## Realizar ejercicios cortar carpeta

Este ejercicio consiste en cortar una carpeta o archivo que deseemos y pegarlo en otra ruta.



## Ver resultado

Cada vez que realicemos un ejercicio correctamente el sistema muestra un mensaje confirmando que se ha realizado correctamente. También aparece un mensaje cuando el usuario pulse el botón de comprobar resultado y no sea correcto.





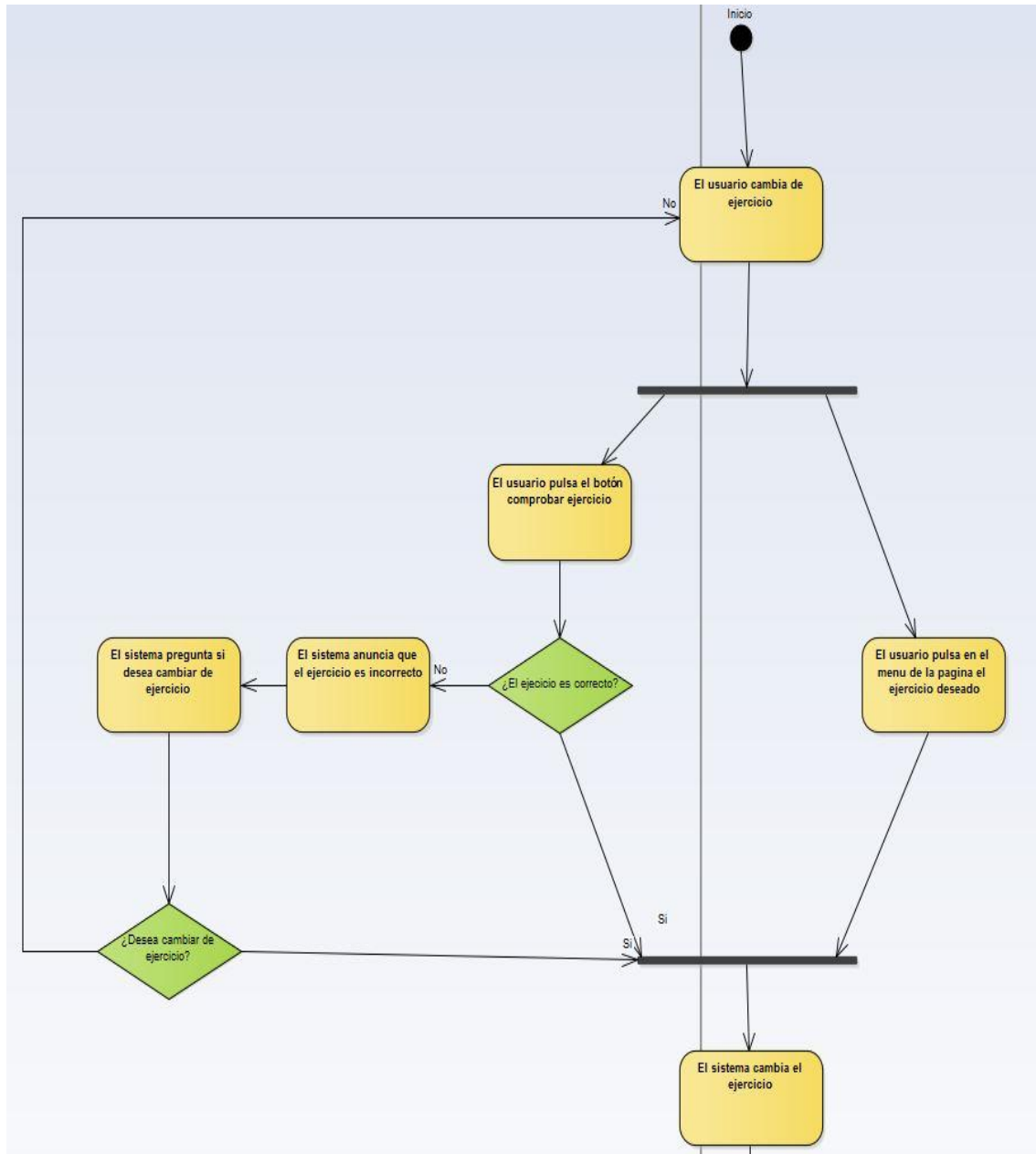
## Ver sección

El usuario puede navegar entre las ventanas de secciones y las ventanas principales cuando desee.



## Cambiar ejercicio

El usuario puede cambiar de ejercicio cuando desee ya sea desde el menú principal de la sección donde se encuentre o desde el propio ejercicio a través del botón comprobar ejercicio.



## Vista lógica

La vista lógica nos permite describir la estructura del sistema. Para representarla he utilizado diagrama de clases

He dividido los diagramas de clases del sistema en:

- Diagramas de clases de archivos HTML y CSS
- Diagrama de clases de archivos HTML y JS

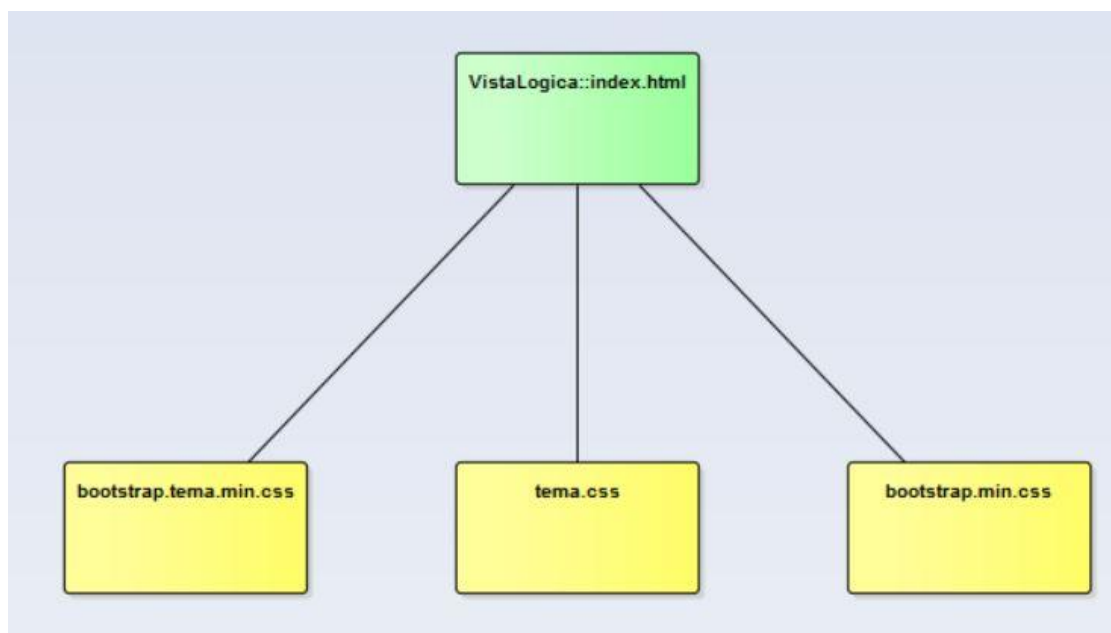
### Diagrama de clases archivos HTML y CSS

En este diagrama se representan los archivos HTML con los archivos CSS de los que depende cada uno.

#### **Index.html**

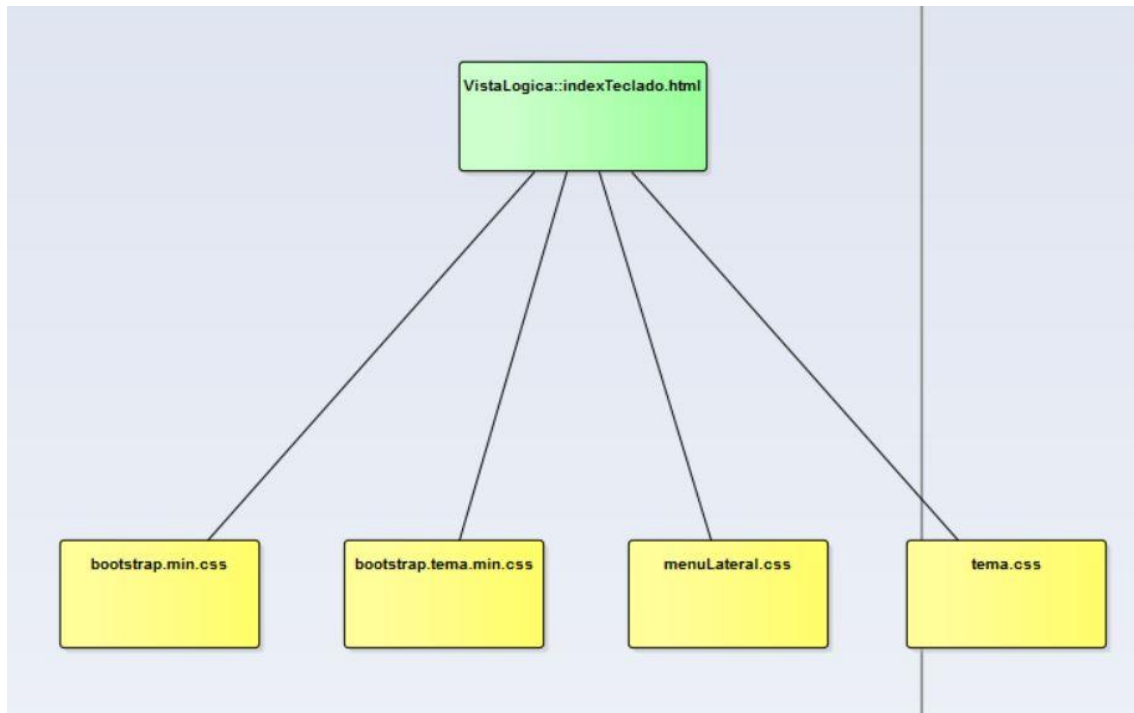
El archivo index.html utiliza los archivos CSS:

- Bootstrap.tema.min.css: contiene el código CC del estilo Bootstrap del tema.
- Tema.css: contiene el código del estilo del tema de la aplicación
- Bootstrap.min.css: contiene el código de estilo Bootstrap como bootstrap.tema.min.css pero aquí está el código de estilo Bootstrap que utilizan alguna página y ejercicios y así tenerlo separado



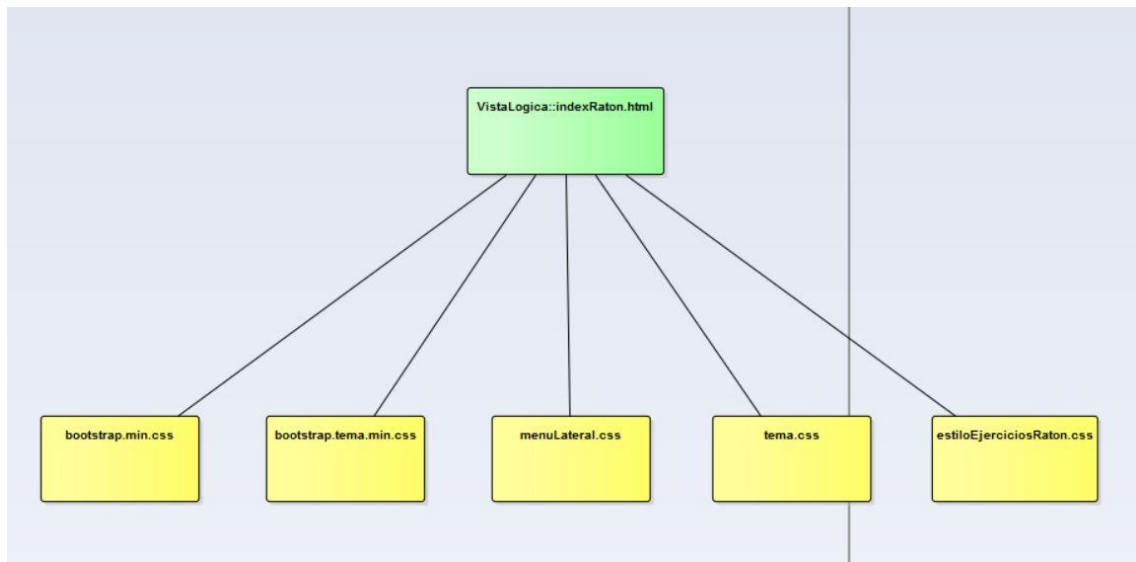
## indexTeclado.html

El archivo indexTeclado.html utiliza los archivos CSS que utiliza index.html más el archivo menuLateral.css que contiene el código del estilo del menú lateral.



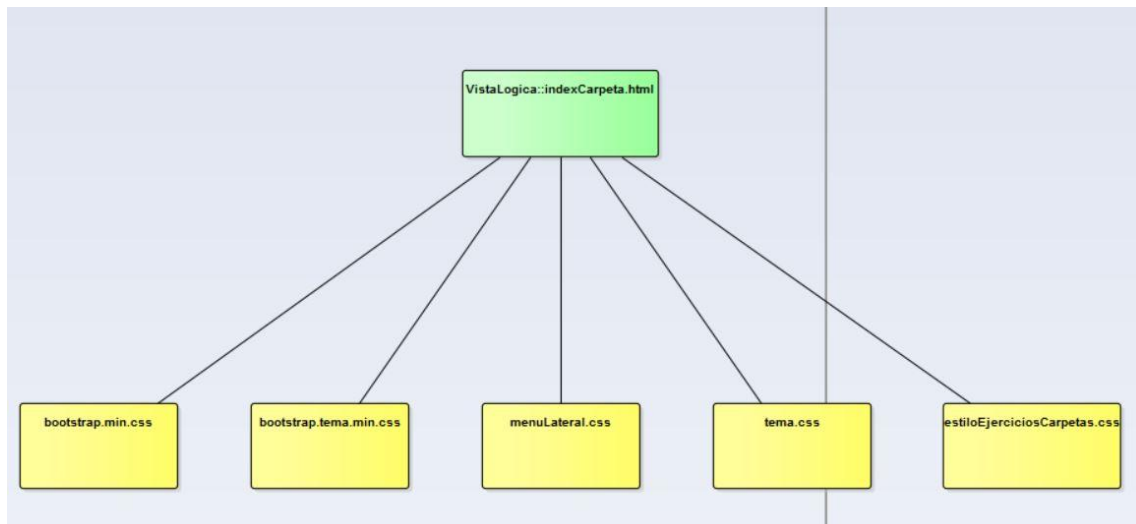
## indexRaton.html

El archivo indexRaton.html utiliza los archivos CSS que utiliza indexTeclado.html más el archivo CSS: estiloEjerciciosRaton que contiene el código del estilo que utilizan algunos ejercicios del ratón, como por ejemplo los ejercicios de desplazamiento del cursor, donde el archivo CSS contiene el color de las distintas partes del circuito y otros atributos como el float: left, para que la caja que contiene la propiedad obtenga un posicionamiento flotante a la izquierda.



## indexCarpetas.html

El archivo `indexCarpetas.html` utiliza los archivos CSS que utiliza `indexTeclado.html` más el archivo CSS: `estiloEjerciciosCarpeta` que contiene el código del estilo que utilizan algunos ejercicios de la carpeta, que por ejemplo contiene los atributos de las cajas que contiene cada carpeta.



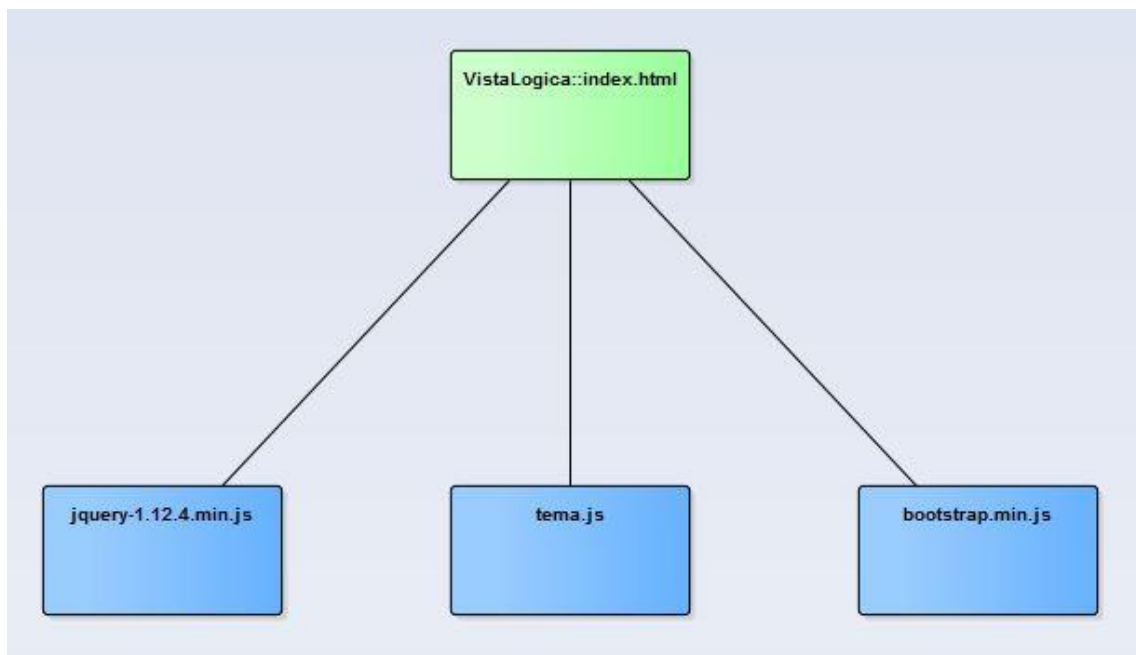
## Diagrama de clases archivos HTML y JS

En este diagrama se representan los archivos HTML con los archivos JS de los que depende cada uno.

### Index.html

El archivo index.html utiliza los archivos JS:

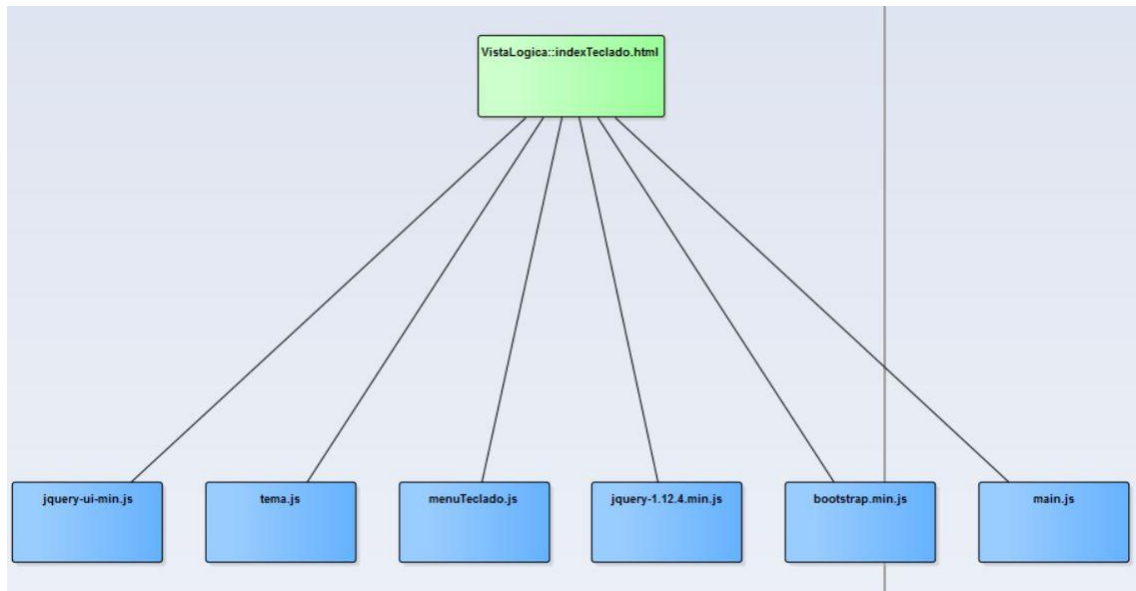
- JQuery-1.12.4.min: contiene el código JS necesario para que JQuery funciones correctamente.
- Tema.js: contiene el código JS que utiliza el tema para funcionar como por ejemplo el posicionamiento del menú principal o su anchura cuando haces scroll en la página.
- Bootstrap.min.js: contiene el código JS para que Bootstrap funcione. correctamente



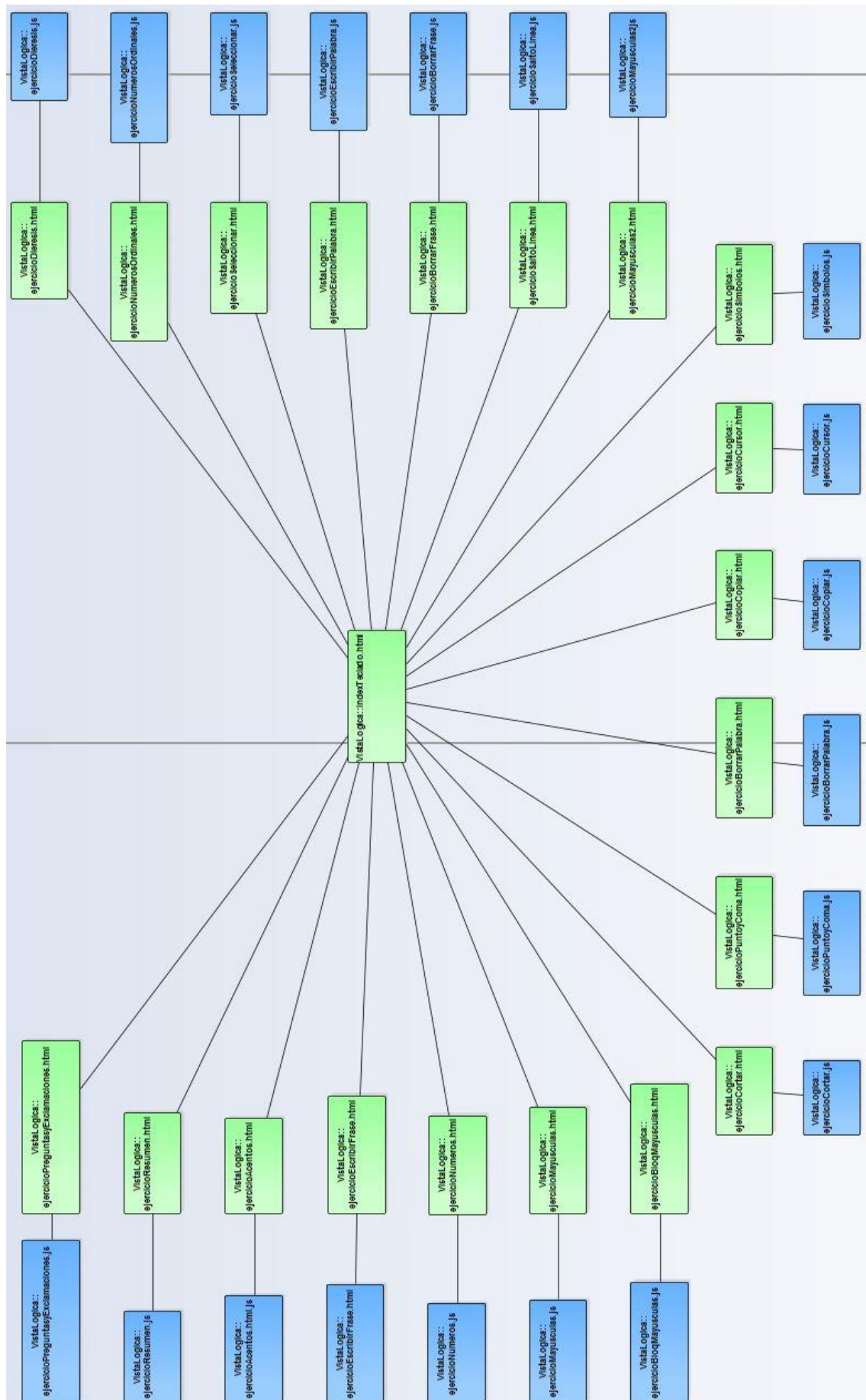
## indexTeclado.html

El archivo indexTeclado.html utiliza los archivos JS que utiliza index.html más los archivos JS:

- JQuery-ui-min.js: que implementa funcionalidades extra para JQuery por ejemplo la funcionalidad draggable que sirve para arrastrar elementos.
- MenuTeclado.js: que contiene el código JS para cargar los distintos ejercicios de la sección teclado.
- Main.js: contiene el código JS para dar un tamaño al div donde se cargan los ejercicios.



Cada ejercicio de la sección teclado contiene un archivo HTML y un archivo JS con el mismo nombre donde se implementa la funcionalidad del ejercicio

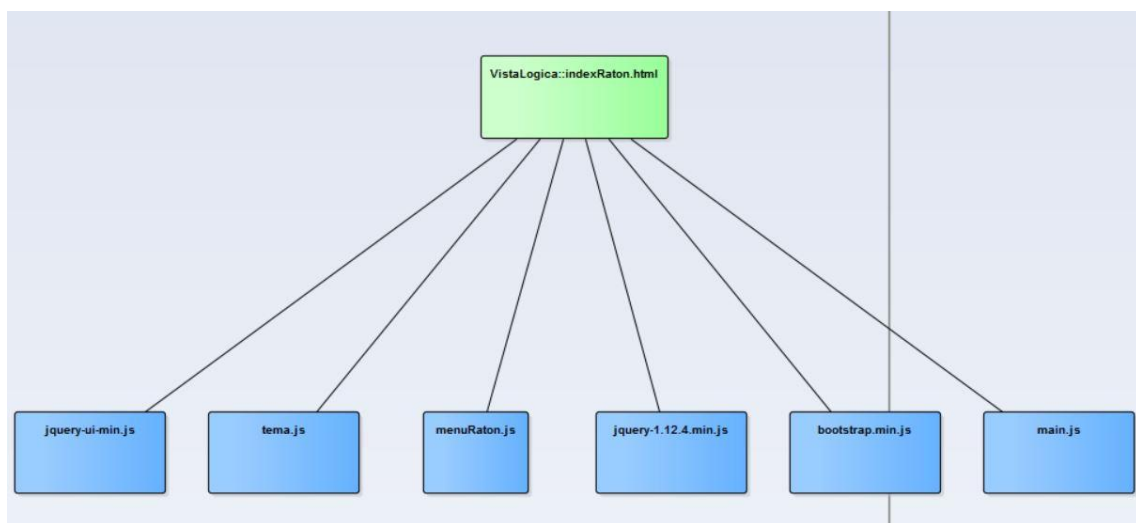




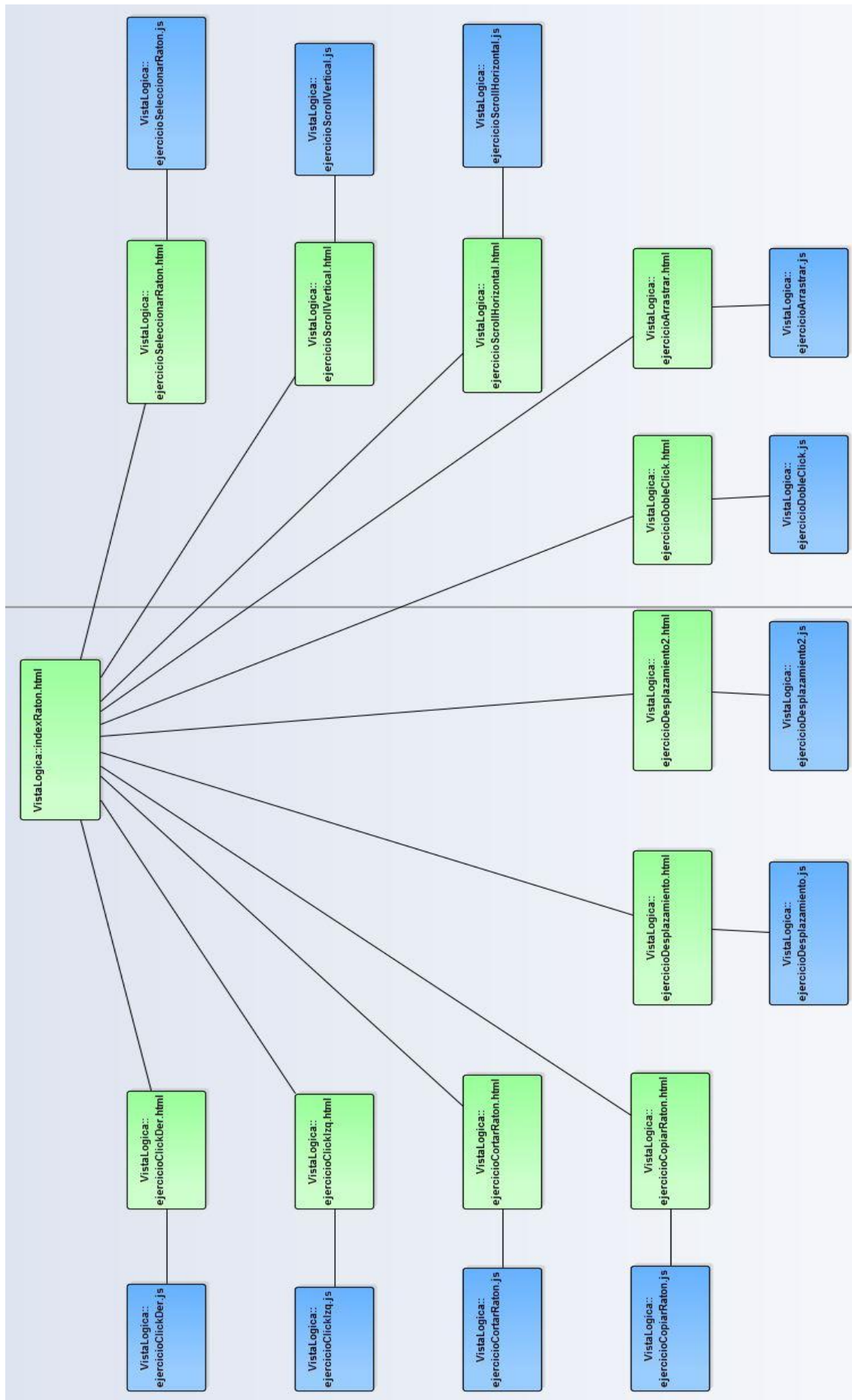
## indexRaton.html

El archivo indexRaton.html utiliza los archivos JS que utiliza index.html más los archivos JS:

- JQuery-ui-min.js
- MenuRaton.js: que contiene el código JS para cargar los distintos ejercicios de la sección de ratón.
- Main.js: contiene el código JS para dar un tamaño al div donde se cargan los ejercicios.



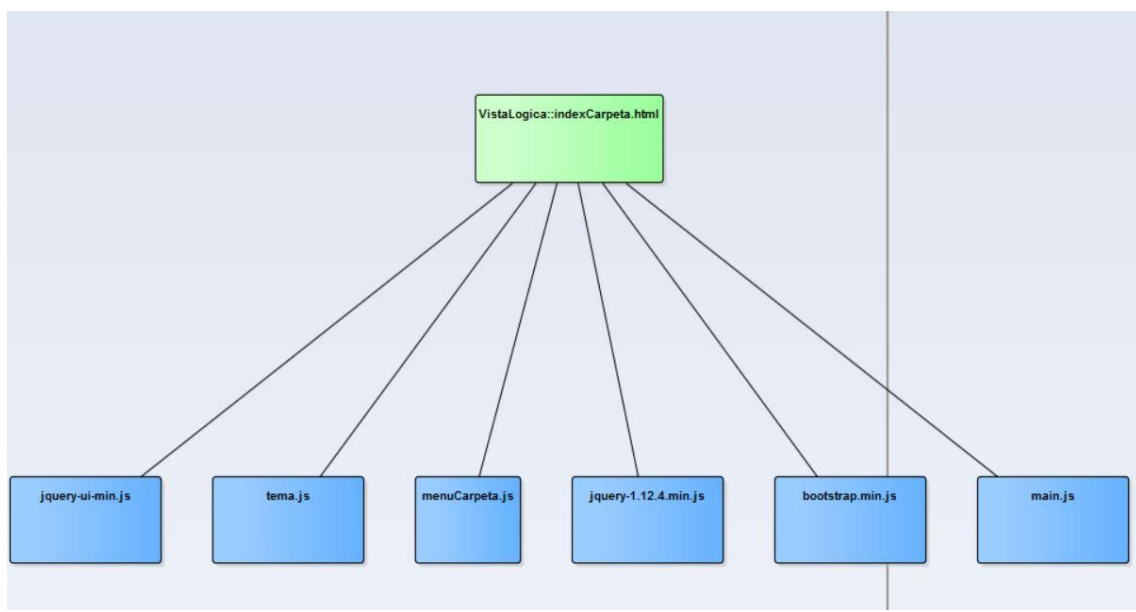
Cada ejercicio de la sección ratón contiene un archivo HTML y un archivo JS con el mismo nombre donde se implementa la funcionalidad del ejercicio.



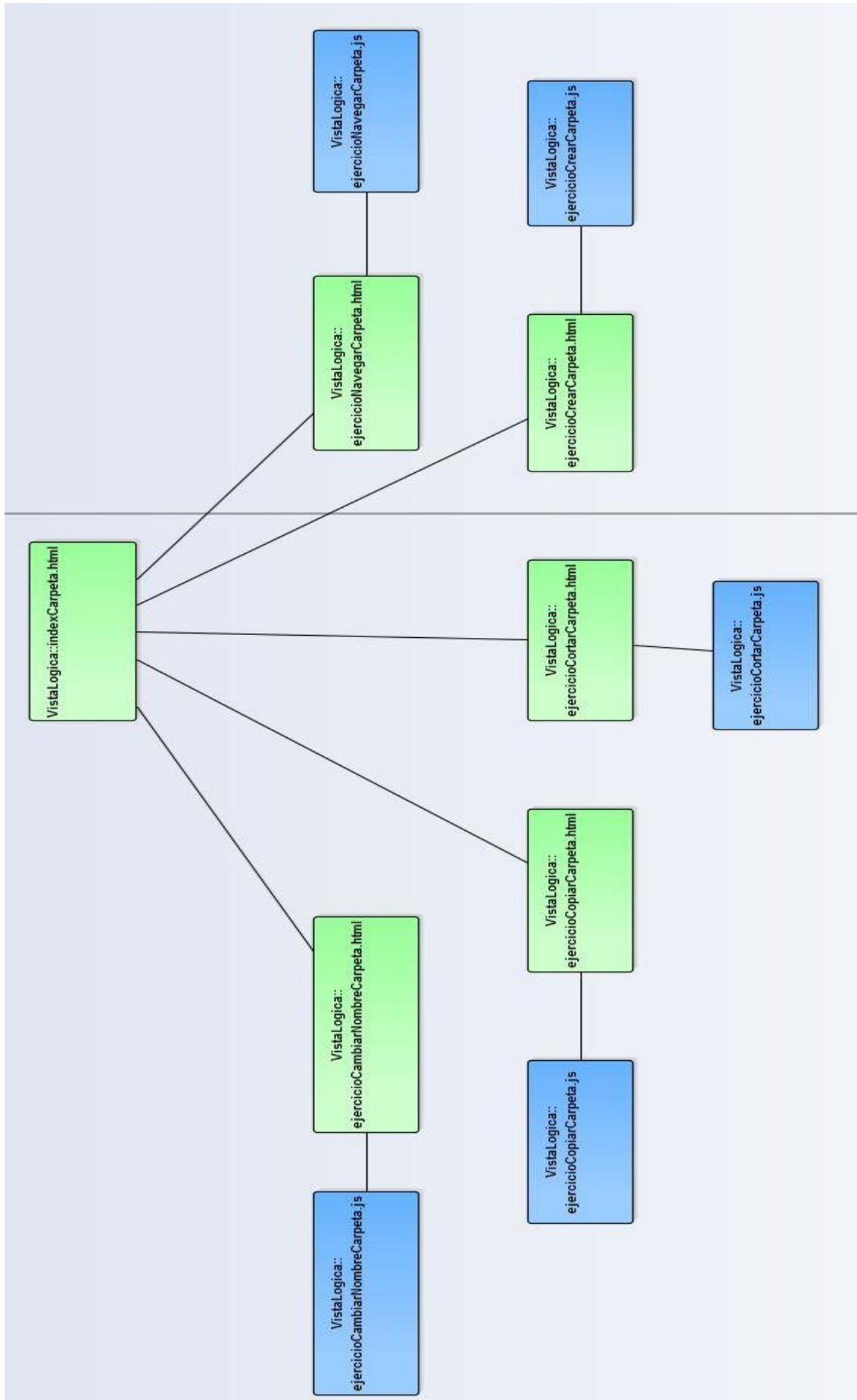
## indexCarpeta.html

El archivo indexCarpeta.html utiliza los archivos JS que utiliza index.html más los archivos JS:

- JQuery-ui-min.js.
- MenuCarpeta.js: que contiene el código JS para cargar los distintos ejercicios de la sección de carpetas.
- Main.js: contiene el código JS para dar un tamaño al div donde se cargan los ejercicios.



Cada ejercicio de la sección carpeta contiene un archivo HTML y un archivo JS con el mismo nombre donde se implementa la funcionalidad del ejercicio.



## Pruebas

Durante el proceso de desarrollo y sobre todo al final fui realizando una serie de prueba que clasifiqué de la siguiente manera:

- Pruebas de funcionamiento: en cada ejercicio realice una serie de pruebas, teniendo en cuenta: los datos de entrada, los datos de salida esperados y los datos obtenidos al acabar el ejercicio.
- Pruebas de visualización: estas pruebas se centraron en probar la aplicación en distintas pantallas con diferentes resoluciones y tamaños para comprobar que la página y los ejercicios se adaptaban correctamente.

## Herramientas y tecnologías utilizadas

En esta sección se explicará las herramientas que se han utilizado para poder desarrollar el trabajo y las tecnologías que se han utilizado.

### Herramientas

#### NetBeans

El entorno de desarrollo utilizado ha sido NetBeans. NetBeans es un entorno de desarrollo libre, está hecho principalmente para el lenguaje de programación Java, pero también soporta muchas otras tecnologías como C, C++, PHP y lo más importante para nosotros HTML5 y JavaScript.



He utilizado este entorno de desarrollo debido a las siguientes razones:

- Es libre.
- Es un entorno que ya conocía, ya que lo hemos utilizado en la universidad para programar en Java

- Su soporte para HTML5 y JavaScript con funciones como el autocompletar o su control de versiones en local.
- Su gran cantidad de plugin

## Enterprise Architect

Para la realización de diagramas he utilizado la herramienta UML Enterprise Architect. Enterprise Architect abarca todo el ciclo de vida del desarrollo de software, con herramientas que facilitan el diseño del software, la ingeniería de sistemas, gestión de requisitos y mucho más.



He utilizado Enterprise Architect debido a las siguientes razones:

- Es una herramienta que ya había utilizado en la universidad y conocía su potencial.
- Tengo una licencia activa.
- La gran cantidad de diagramas que puedes realizar fácilmente
- Las herramientas que posee para exportar el proyecto a HTML y la herramienta para el control de la trazabilidad entre los elementos.

## Gimp

Para la edición de imágenes de la aplicación y de la memoria se ha utilizado la herramienta de edición de imágenes Gimp.



He utilizado Gimp por las siguientes razones:

- Es libre
- Puedes exportar las imágenes a una gran cantidad de formatos.
- La gran cantidad de opciones que tiene incluidas.

## Tecnologías

### JavaScript

El lenguaje de programación utilizado para desarrollar la funcionalidad de la aplicación ha sido JavaScript.

JavaScript es un lenguaje de programación interpretado, además es orientado a objetos, imperativo, débilmente tipado y dinámico.



### JQuery

He utilizado la biblioteca de JavaScript JQuery. Implementa funcionalidades basadas en JavaScript que sin ella requeriríamos de más tiempo y mayor dificultad de implementación.

He utilizado JQuery por las siguientes razones:

- Permite.
- Nunca la había utilizado y quería aprenderla.
- Simplifica la manera de interactuar con paginas HTML y de manipular el árbol DOM.
- Simplifica manejar eventos.
- Las posibilidades que ofrece en animaciones.



### HTML5

Para la elaboración de las páginas web he utilizado el lenguaje de marcado HTML en su versión 5. HTML define un código que permite definir el contenido de nuestra página como textos e imágenes. HTML5 incorpora respecto a versiones anteriores una serie de nuevos elementos y atributos que permiten dar un aspecto a las páginas web más modernas.

He utilizado HTML5 por las siguientes razones:



- Incorpora etiquetas como <nav> y <footer> que permite darle un toque más actual a la página.
- Mejoras en los formularios.
- Nuevas funcionalidades Drag and Drop que permite arrastrar imágenes.
- En un lenguaje que quería aprender.

## CSS

Para dar un estilo a la aplicación web he utilizado CSS. CSS son hojas de estilo en cascada donde se definen a información del estilo que queramos que tenga nuestra aplicación web.



## Bootstrap

Para realizar el diseño de la aplicación web he utilizado Bootstrap. Bootstrap es un conjunto de herramientas para diseñar sitios web. Está basado en HTML y CSS y posee plantillas con tipografías, botones, menús de navegación y muchos más.

He utilizado Bootstrap por las siguientes razones:

- Me gusta mucho el estilo de páginas con Bootstrap.
- Facilita el uso del diseño responsivo.





## Manual de usuario

### Página principal:

Página principal es la página que aparece cuando arrancamos la aplicación.

**Título:** el título de la aplicación.

**Secciones:** desde este menú podremos acceder al contenido de las distintas secciones de la aplicación donde encontraremos juegos y explicaciones.

**Teoría de las secciones:** pulsando en cada una de las imágenes se nos desplegará la página que veremos a continuación.



**Teoría:** Explicación teórica de la sección.

**Cerrar:** botón para cerrar la página.



## Secciones:

Las secciones teclado, ratón y carpetas tienen la misma estructura, vamos a ver la sección de teclado que nos servirá para aprender el funcionamiento de las demás secciones.

### Título de la sección

**Menú de ejercicios:** desde este menú podremos acceder a los distintos ejercicios que forman la sección.

**Teoría:** teoría del elemento que se practica en la sección.



## Ejercicios

Todos los ejercicios poseen la misma estructura.

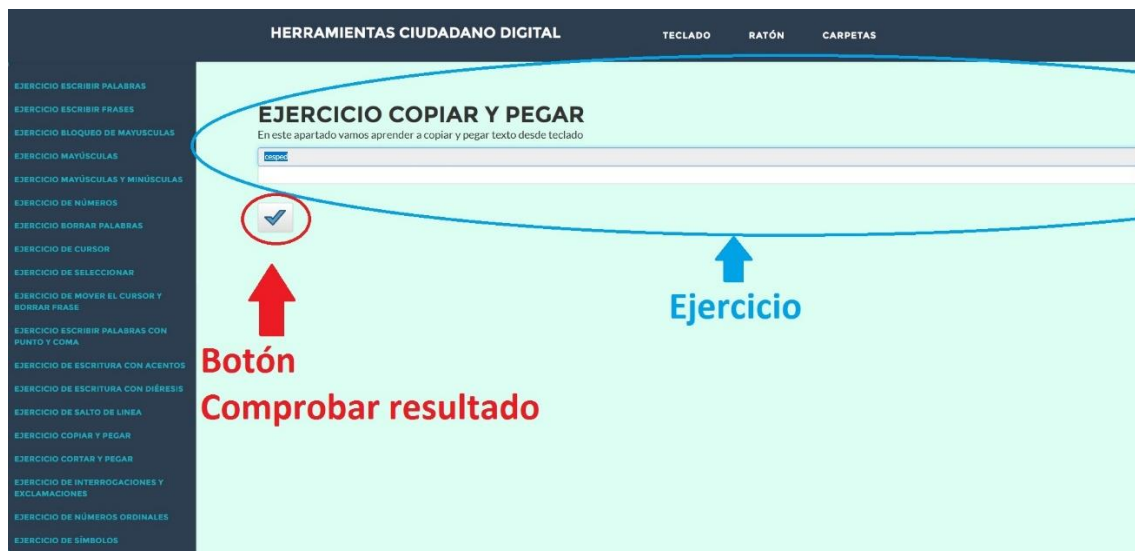
**Ejercicio:** donde podremos ver:

- El título del ejercicio.
- La explicación de la realización del ejercicio.
- El resultado del ejercicio satisfactorio.
- Mensajes que irán apareciendo durante la realización del ejercicio

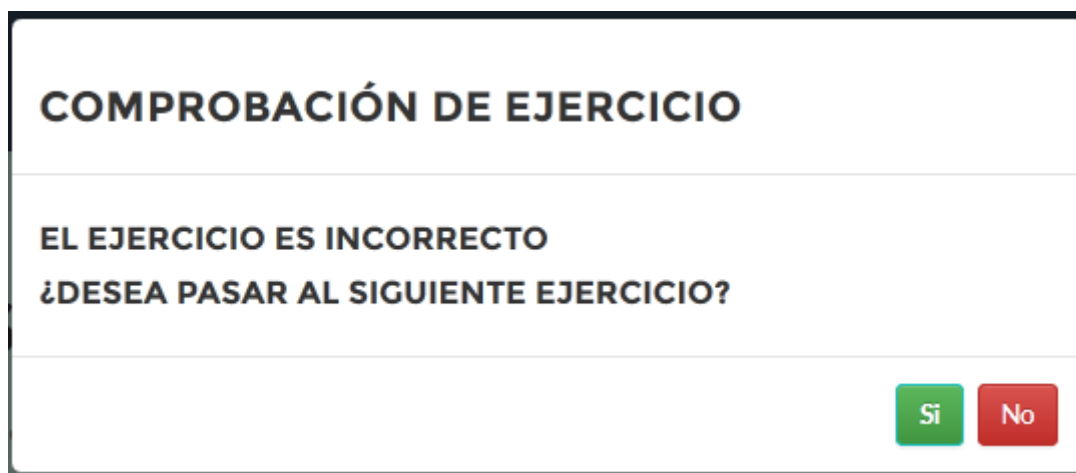
Cuando el ejercicio se resuelva correctamente la aplicación mostrará un mensaje automáticamente.

**Botón comprobar resultado:** al presionar el botón si el ejercicio es incorrecto nos aparecerá una ventana emergente (que veremos a continuación) informándonos y preguntando si queremos pasar al siguiente ejercicio.

Desde el menú de ejercicios podremos cambiar al ejercicio que deseemos cuando queramos.



Ventana emergente botón comprobar resultado.



## Conclusión

Al realizar el trabajo, he podido aprender lo importante que es en la sociedad actual ayudar a las personas a que desarrollen las competencias básicas del ciudadano digital. Muchas personas sobre todo personas mayores pueden sentirse aisladas del resto de la sociedad, una sociedad en la que cada vez estamos más conectados y realizamos más gestiones y trámites con el ordenador, sin pensar en las personas que no tienen los suficientes conocimientos y que cada vez sienten como crece la necesidad de aprender y no quedarse atrás en el uso de la tecnología.

Por estas razones es importante que organizaciones y fundaciones, como la fundación ECDL, realicen programas y cursos para que personas de cualquier ámbito y edad puedan aprender y desarrollar estas competencias básicas. Por esta razón estoy muy contento de haber elegido este trabajo y que sirva para poder ayudar a que estas personas, aprendan las competencias básicas del ciudadano digital.

Con la realización del trabajo he podido aprender y comprender (una vez más) lo importante que es realizar una buena ingeniería del software para poder realizar un proyecto de buena calidad y tener un código lo más limpio y comprensible posible, sin código repetido ni código muerto.

Muchas veces vamos directos a realizar la programación saltándonos la ingeniería del software, con la idea de que lo importante es el resultado final sin importar como se llegue a él. Esto es una idea errónea, ya que tan importante es el resultado final como el proceso de hacerlo, ya que si realizamos una buena ingeniería del software podremos implementar un código más limpio y que permita su fácil mantenibilidad y extensibilidad.

Como posibilidad de extensibilidad se podrían realizar ejercicios específicos para dispositivos como tablets y móvil ya que ha ahora mismo hay ejercicios compatibles como algunos de escritura y de ratón. Además, la aplicación tiene un diseño adaptativo que se ajusta a pantallas más pequeñas.

Por último, Herramientas Ciudadano Digital es una aplicación web con muchas posibilidades, pudiendo aumentar sus secciones sin afectar al resto de la aplicación ya que como hemos visto en este documento, la funcionalidad de cada ejercicio está contenida en su archivo JavaScript correspondiente, por lo que se podría extender su funcionalidad

fácilmente. Además, es una gran herramienta de aprendizaje que puede ayudar a las personas a aprender las competencias básicas del ciudadano digital y además poderlas entretener.

Por conclusión, como he mencionado anteriormente estoy muy contento de haber elegido este trabajo ha sido en momentos difícil sobre todo en la sección de carpetas, donde he tenido que pensar como poder simular un explorador de carpetas que permita crear carpetas dinámicamente asignándolas una funcionalidad definida.

Pero sobre todo, Herramientas Ciudadano Digital me ha permitido aprender mucho de las aplicaciones web aprendiendo JavaScript, JQuery, HTML5 y Bootstrap. Además, he podido comprobar que estas son tecnologías muy potentes, que están muy presentes en el entorno web actualmente y que con ellas puedes desarrollar todo lo que se te pueda ocurrir.

Espero que Herramientas Ciudadano Digital ayude a personas a aprender y a divertirse a la vez.

## Bibliografía

Usuarios de Wikipedia. (2016). Modelo de Vistas de Arquitectura 4+1.

Recuperado de:

[https://es.wikipedia.org/wiki/Modelo\\_de\\_Vistas\\_de\\_Arquitectura\\_4%2B1](https://es.wikipedia.org/wiki/Modelo_de_Vistas_de_Arquitectura_4%2B1)

Fundación ECDL. Página oficial fundación ECDL. (2016). Recuperado de:

<http://www.ecdl.es/>

Ministerio de Educación, Cultura y Deporte (MECD). (2015). Competencia Digital. Recuperado de:

<http://www.mecd.gob.es/educacion-mecd/mc/lomce/el-curriculo/curriculo-primaria-eso-bachillerato/competencias-clave/digital.html>

NetBeans. (2016). NetBeans IDE Features. Recuperado de:

<https://netbeans.org/features/index.html>

Sparx Systems. (2016). Enterprise Architect. Recuperado de:

<https://www.sparxsystems.es/>

Murphey R. (2013). Fundamentos de JQuery. Recuperado de:

<http://librojquery.com/>

Usuarios Wikipedia. (2016). HTML5. Recuperado de:

<https://es.wikipedia.org/wiki/HTML5>

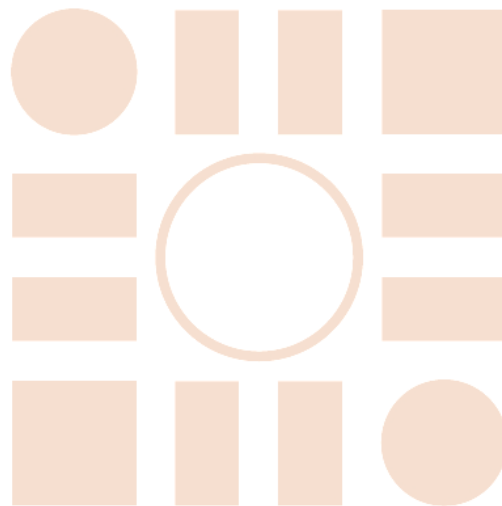
Bootstrap. (2016). Página oficial Bootstrap. Recuperado de:

<http://getbootstrap.com/>

Usuarios de Wikipedia. (2016). Hoja de Estilos en Cascada. Recuperado de:

[https://es.wikipedia.org/wiki/Hoja\\_de\\_estilos\\_en\\_cascada](https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada)

Universidad de Alcalá  
Escuela Politécnica Superior



ESCUELA POLITECNICA  
SUPERIOR



Universidad  
de Alcalá