



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Entrenador gráfico de biestables

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Guillermo Jesús Rodrigo García

Tutor: Antonio Martí Campoy

2015-2016

Agradecimientos

A mi madre por haber sido el sustento, el apoyo, el consejo, la sabiduría, la paciencia y la fuerza sin la cual nada de esto habría sido posible.

A mi padre por haberme hecho persistente, tenaz, valiente y no haber dejado que me rindiera.

A mi familia, por el ánimo y el apoyo incondicional que han demostrado para con mi proyecto de futuro.

A mis compañeros y amigos, la familia que se elige, por haberme dado esos fantásticos momentos de descanso y diversión que lo han hecho todo un poco más fácil.

A mis profesores por darme las herramientas para sacar todo mi potencial y así poder hacer cosas grandes.

Por último, al personal de la escuela, en especial a Cristina, por su esfuerzo, su dedicación y porque sin ellos la escuela no sería lo mismo.



Resumen

El profesorado de la asignatura “Fundamentos de Computadores” del primer curso del Grado en Ingeniería Informática ha mostrado su preocupación, puesto que los alumnos de la misma presentan dificultades a la hora de resolver correctamente ejercicios sobre biestables y sus cronogramas.

Es por ello que se ha decidido realizar este proyecto, que consiste en una aplicación web que plantea una serie de ejercicios en los que al alumno se le muestran una serie de cronogramas asociados a un determinado biestable, donde aparece la señal de reloj y la entrada o entradas del biestable, teniendo el alumno que dibujar la señal que falta, es decir la salida del biestable.

Finalmente, y además de todo ello, se puede concluir que es una herramienta docente muy útil debido no sólo a su sencillez de uso, sino porque permite al alumno llevar un seguimiento de sus avances y le permite resolver sus dudas en las diferentes secciones de ayuda de que dispone.

Palabras clave: herramienta, cronograma, señal, biestable, ejercicio, alumno, ayuda, docente, sencillez, útil.

Abstract

The professorship of the subject "Fundamentos de Computadores" of the first degree course in Computer Engineering have expressed concern, due to the fact that students have some difficulties in properly solving exercises about flip-flops and digital signals.

That is why it was decided to carry out this project, which is a web application that presents a series of exercises in which the student is shown a series of digital signals associated with a particular flip-flop, namely the corresponding clock signals and input, and he should draw the digital signal missing corresponding to the output signal.

Finally, and besides all this, one can conclude that it is a very useful teaching tool not only because of its simplicity, but because it allows students to track their progress and allows you to answer your questions in the different help sections that has.

Keywords: tool, digital, signal, flip-flop, exercise, student, help, teaching, simplicity, useful.

Tabla de contenidos

1	Introducción.....	9
1.1	Motivación	9
1.2	Conceptos generales	10
1.2.1	Biestables D	10
1.2.2	Biestables T.....	11
1.2.3	Biestables JK	12
2	Especificación de requisitos	13
2.1	Introducción	13
2.1.1	Propósito	13
2.1.2	Ámbito	13
2.1.3	Definiciones, acrónimos y abreviaturas	13
2.2	Descripción general	14
2.2.1	Perspectiva del producto	14
2.2.2	Funciones del producto	15
2.2.3	Características del usuario	15
2.2.4	Restricciones generales	16
2.3	Requerimientos específicos	16
2.3.1	Requerimientos funcionales.....	16
2.3.2	Requerimientos de interfaces externos.....	18
2.3.3	Requerimientos de eficiencia	21
2.3.4	Restricciones de diseño	21
2.3.5	Atributos.....	21
2.3.6	Otros requerimientos	22
3	Análisis.....	23
3.1	Casos de uso.....	23
3.1.1	Casos de uso: Usuario no registrado	23
3.1.2	Casos de uso: Usuario registrado	23
4	Diseño.....	25
4.1	Arquitectura de tres capas	25
4.1.1	Capa de presentación	25



4.1.2	Capa de negocio	25
4.1.3	Capa de acceso a datos	26
5	Tecnologías.....	27
5.1	Capa de presentación.....	27
5.1.1	HTML	27
5.1.2	CSS.....	27
5.1.3	JavaScript.....	28
5.2	Capa de negocio	28
5.2.1	PHP.....	28
5.3	Capa de acceso a datos.....	29
5.3.1	MySQL.....	29
6	Implementación	29
6.1	Capa de presentación.....	30
6.2	Capa de negocio	37
6.3	Capa de acceso a datos.....	39
6.4	Funcionalidad esencial	40
7	Evaluación y pruebas	41
7.1	Prueba de validación de CSS.....	41
7.2	Prueba de validación de navegadores web	43
7.3	Prueba de seguridad de acceso	44
8	Conclusiones	47
8.1	Trabajo realizado	47
8.2	Valoración personal	47
8.3	Futuras mejoras	47
9	Bibliografía.....	49
Anexos		51
A.	Herramientas utilizadas	51
a.	ArgoUML	51
b.	Moqups	51
c.	Sublime Text	52
d.	WampServer	53

1 Introducción

Este proyecto consiste en el diseño y la implementación de una aplicación Web que plantea una serie de ejercicios sobre biestables a los alumnos para que estos los resuelvan y guía a aquellos estudiantes que no den con la solución correcta facilitando así el aprendizaje de esta materia.

Ante el usuario aparecerá un cronograma, es decir, una representación gráfica en el tiempo de la evolución de la señal del reloj y de las entradas del biestable. Sobre este cronograma el alumno deberá dibujar la gráfica correcta correspondiente a la salida del biestable en función de las entradas mostradas.

La aplicación propone al usuario ejercicios sobre diferentes tipos de biestables, a saber biestables D, biestables T y biestables JK activos o bien por flanco de subida o bien por flanco de bajada.

Además, el entrenador permite al usuario comprobar si la solución que ha propuesto es correcta o no y en función de eso modifica su contador de aciertos y fallos y estos quedan registrados en una base de datos para su posterior consulta.

Finalmente, si el usuario no logra encontrar la solución correcta de un problema propuesto, la aplicación puede mostrar, a petición de este, la solución correcta del ejercicio.

1.1 Motivación

La realización de este proyecto está motivada por la asignatura “Fundamentos de Computadores” del primer año de Grado en Ingeniería Informática en la ETSINF. Dentro de los contenidos de esta asignatura, los biestables son una parte importante por sí mismos y también para contenidos posteriores como el diseño de sistemas secuenciales.

No obstante, se ha observado que los alumnos que cursan esta asignatura presentan algunas dificultades a la hora de resolver los problemas de la parte de biestables dado que, normalmente, son conocimientos totalmente nuevos para el estudiante.

Debido a ello, se ha decidido desarrollar esta aplicación Web para que sirva de herramienta a los futuros alumnos de la asignatura en particular y a todos aquellos que quieran aprender a resolver este tipo de ejercicios en general.

También hay que destacar que se ha decidido realizar esta aplicación en formato Web para que el alumno no necesite descargar e instalar nada, sino que pueda acceder al entrenador desde cualquier ordenador con conexión a Internet.

Además, de este modo el mantenimiento de la aplicación es más sencillo, puesto que los cambios se distribuyen automáticamente cuando el usuario vuelve a acceder a la web y no tiene que actualizar ninguna aplicación.

1.2 Conceptos generales

Un biestable es un circuito digital capaz de permanecer en uno de dos estados posibles durante un tiempo indefinido. Además, existen dos tipos de biestables, los asíncronos, que solamente dispone de entradas de control y los síncronos, que además cuentan con una entrada de sincronismo llamada entrada de reloj.

Esta aplicación web trata los ejercicios y problemas relacionados con los biestables síncronos y más concretamente los relacionados con biestables D, T y JK, todos ellos activos por flanco y cuyas características van a ser descritas a continuación.

1.2.1 Biestables D

Los biestables D son también conocidos como biestables de datos o seguidores, disponen de una única entrada llamada D además de la entrada C de la señal de reloj.

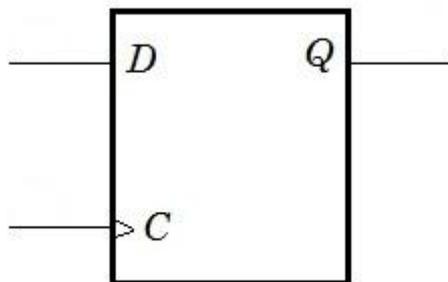


Figura 1.1 Biestable D activo por flanco de subida

La salida Q de un biestable D toma el valor de la entrada D cuando se produce en la entrada de reloj el flanco activo, que puede ser de subida o de bajada. No

obstante, para entender mejor cómo actúa un biestable D, debemos fijarnos en su tabla de funcionamiento.

C	D	Q(t)	Q(t+1)
↑	0	X	0
↑	1	X	1
0/1/↓	X	X	Q(t)

Tabla 1.1 Funcionamiento del biestable D activo por flanco de subida (I)

1.2.2 Biestables T

Los biestables T o *toggle* disponen de una única entrada T además de la entrada C de la señal de reloj.

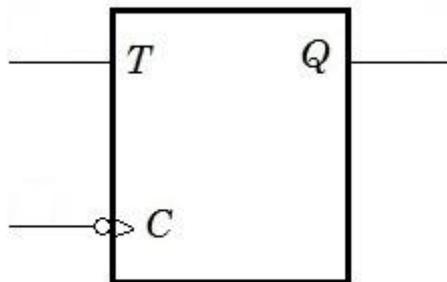


Figura 1.2 Biestable T activo por flanco de bajada

Al producirse el flanco activo y su entrada T tener el valor 1, la salida Q cambiará su valor, sea cual sea, al contrario. Es decir, si el biestable T se activa, su entrada T tiene el valor 1 y la salida Q tiene el valor 0, esta última pasará a tener el valor 1.

En caso de producirse el flanco activo y la entrada T tener el valor 0, la salida Q mantendrá su valor anterior hasta el siguiente flanco activo.

El comportamiento de este dispositivo, se entiende con mayor claridad al observar su tabla de funcionamiento.

C	T	Q(t)	Q(t+1)
↓	0	0	0
↓	0	1	1

↓	1	0	1
↓	1	1	0
0/1/↑	X	X	Q(t)

Tabla 1.2 Funcionamiento del biestable T activo por flanco de bajada (X=No importa)

1.2.3 Biestables JK

Los biestables JK disponen de una entrada J, una entrada K y la entrada C de la señal de reloj.

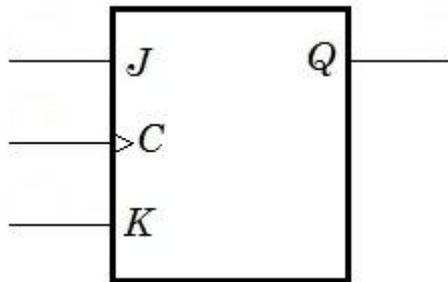


Figura 1.3 Biestable JK activo por flanco de subida

La entrada J actúa para la puesta a 1 o *set* y la entrada K actúa para la puesta a 0 o *reset*. En caso de que ambas entradas estén a 1, la salida Q del biestable invierte su valor como el biestable T y cuando ambas están a 0 se mantiene el valor de salida.

Para entender mejor cómo actúa un biestable JK, debemos estudiar su tabla de funcionamiento.

C	J	K	Q
↑	0	0	Q
↑	0	1	0
↑	1	0	1
↑	1	1	/Q(t)
0/1/↓	X	X	Q(t)

Tabla 1.3 Funcionamiento del biestable JK activo por flanco de subida (X=No importa)

2 Especificación de requisitos

2.1 Introducción

2.1.1 Propósito

El propósito de este apartado es especificar los requerimientos para la aplicación Web que se está tratando en este documento. La especificación de requisitos resulta necesaria a la hora de establecer una guía sobre la que empezar a trabajar para el desarrollo de dicha aplicación Web, así como para dar a conocer este proyecto a todo el que desee saber más acerca de él.

2.1.2 Ámbito

El entrenador gráfico de biestables consiste en una aplicación Web que va a permitir al usuario que la utilice realizar ejercicios relacionados con biestables D, biestables T y biestables JK.

En estos ejercicios el usuario conoce las gráficas correspondientes a la señal de reloj y a la señal o señales de entrada y debe dibujar la gráfica correspondiente a la salida Q haciendo *click* en los diferentes semiperíodos. Así, el usuario puede dar la respuesta de forma gráfica y sencilla utilizando el ratón para ello.

También existen una serie de botones que permiten comprobar si el ejercicio está resuelto correctamente, empezar uno nuevo o mostrar la solución correcta de un determinado problema. Por consiguiente, si el usuario no consigue resolver un ejercicio propuesto, podrá pasar a otro o simplemente ver la solución haciendo que el entrenador se adapte al usuario y no al revés.

Además de todo ello, la aplicación cuenta con una sección de ayuda que guiará al usuario y le ayudará a resolver correctamente aquellos ejercicios que se le propongan así como a conocer más en profundidad datos concretos sobre los biestables ya mencionados.

Por último, el entrenador dispone de un apartado donde se puede ver los aciertos y fallos globales de cada uno de los tipos de ejercicio desde el momento en que alguien empezó a utilizar la aplicación hasta el instante actual, dando así una idea de los avances y motivar a quien lo utiliza. Esto es posible gracias a una base de datos que registra todos los aciertos y fallos de cada usuario en cada tipo de ejercicio.

2.1.3 Definiciones, acrónimos y abreviaturas

Usuario registrado: Cualquier persona que accede al entrenador gráfico de biestables y que deberá haberse registrado previamente en dicha aplicación.

Sistema operativo: Conjunto de órdenes y programas que controlan los procesos básicos de una computadora y permiten el funcionamiento de otros programas.

UML: Lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

HTML: Lenguaje que describe la estructura y el contenido semántico de un documento web.

CSS: Lenguaje usado para definir y crear la presentación de un documento estructurado escrito en HTML o XML.

PHP: Lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

JavaScript: Lenguaje de programación interpretado, orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Biestable: Célula elemental de memoria capaz de almacenar un bit de información.

Navegador web: Software, aplicación o programa que permite el acceso a la Web, interpretando la información de distintos tipos de archivos y sitios web para que estos puedan ser visualizados.

Software: Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.

Sistema de gestión de bases de datos: Conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos.

2.2 Descripción general

2.2.1 Perspectiva del producto

Para el desarrollo y la ejecución de esta aplicación Web, se ha utilizado el sistema operativo Windows 7 Professional. Así pues, esta aplicación puede utilizarse en cualquier sistema operativo que tenga conexión a Internet y un navegador web que soporte el elemento *canvas* de HTML5.

Además, las tecnologías utilizadas para la implementación de este producto han sido:

- HTML5 para proporcionar una estructura a las diferentes páginas Web de las que se compone la aplicación.
- PHP para acceder a la base de datos de la aplicación y así poder registrar y autenticar usuarios así como los datos pertenecientes a estos.
- Plantillas CSS para mejorar la interfaz gráfica del producto en función de las estructuras hechas en HTML5.
- JavaScript para implementar las funcionalidades tales como cambiar gráfica y lógicamente el valor de los semiperíodos de la gráfica de salida, dibujar las diferentes gráficas, etc.
- MySQL como sistema de gestión de base de datos.

Cabe destacar que dentro de HTML5 se ha utilizado el elemento *canvas* para crear el área donde el usuario ya registrado podrá dibujar la gráfica de salida para resolver el ejercicio en cuestión.

Por último, se necesita un servidor para alojar la Web que soporte las tecnologías mencionadas anteriormente.

2.2.2 Funciones del producto

Usuario registrado:

- Resolver ejercicios de cronogramas de biestables D, T y JK activos por flanco de subida y de bajada.
- Comprobar si la solución propuesta es correcta.
- Solicitar la solución correcta al ejercicio planteado.
- Solicitar un nuevo ejercicio.
- Consultar el registro de aciertos y fallos.
- Consultar la sección de ayuda.

Usuario no registrado:

- Registrarse en el sistema.

Sistema:

- Abrir sesión de usuario registrado.
- Cerrar sesión de usuario registrado.
- Registrar aciertos y fallos del usuario registrado.

2.2.3 Características del usuario

Existen dos tipos de usuario: el usuario no registrado y el usuario registrado.

Por un lado está el usuario no registrado, que es aquel que no puede realizar acciones en la aplicación salvo registrarse en ella. Una vez decida registrarse, se le presentará un formulario que debe rellenar con su nombre de usuario y contraseña para así tener un acceso total e ilimitado a las funcionalidades que ofrece el entrenador gráfico de biestables.

Por otro lado está el usuario registrado, que tiene un acceso total a las funcionalidades que ofrece la aplicación. Este usuario puede realizar los ejercicios propuestos para todos los tipos de biestables existentes en el entrenador; a saber, biestables D, biestables T y biestables JK activos tanto por flanco de subida como por flanco de bajada.

Si lo desea, este usuario puede también consultar la sección de ayuda que se ofrece para así entender mejor cómo resolver este tipo de ejercicios y problemas.

2.2.4 Restricciones generales

Para acceder a esta aplicación Web, se requiere únicamente de una computadora con acceso a Internet y de un navegador Web que soporte el elemento *canvas* de HTML5, dado que este se ha utilizado para el desarrollo del entrenador.

2.3 Requerimientos específicos

2.3.1 Requerimientos funcionales

2.3.1.1 Usuario registrado

Resolver ejercicios de cronogramas de biestables D, T y JK activos por flanco de subida y bajada.

Cualquier usuario registrado que entre a la aplicación puede seleccionar el tipo de problema que quiere intentar resolver. Una vez seleccionado, aparecen tres gráficas en la pantalla y el usuario procederá a resolverlo haciendo *click* en los diferentes semiperíodos de la gráfica de salida para cambiar su valor y dibujar así la gráfica de salida correcta.

Un requerimiento importante es que se permita al usuario definir su solución, es decir, dibujar la gráfica de la salida del biestable, con una resolución mínima a nivel de semiperíodo de reloj. De esta forma el usuario debe decidir si el biestable con el que está trabajando cambia su estado en los flancos de subida o de bajada.

Comprobar si la solución propuesta es correcta.

Cuando el usuario ha dibujado la gráfica de salida, puede comprobar si es correcta o no haciendo *click* en el botón “Comprobar” y se le indicará si ha acertado o ha fallado aumentando el contador de aciertos o fallos que se muestra debajo de los botones.

Solicitar la solución correcta al ejercicio planteado.

Si el usuario no logra resolver un determinado problema propuesto, se le da la opción de solicitar la solución correcta. Así pues, haciendo *click* sobre el botón “Mostrar solución” del ejercicio, se dibuja automáticamente la gráfica de salida correcta.

Solicitar nuevo ejercicio.

Cuando el usuario ha terminado con un ejercicio, se ha rendido o simplemente quiere intentar resolver un nuevo problema del mismo tipo, se le da la opción de hacer *click* en el botón “Nuevo ejercicio” y automáticamente aparecerá en la pantalla un nuevo ejercicio del mismo tipo.

Consultar el registro de aciertos y fallos.

En cualquier momento, el usuario de la aplicación puede consultar sus estadísticas de aciertos y fallos de cada tipo de ejercicio existente en el entrenador. Estos datos se muestran en forma de tabla para mayor claridad y legibilidad.

Consultar la sección de ayuda.

Cuando surjan dudas al usuario durante el uso del entrenador, este podrá consultar la sección de ayuda disponible en la aplicación. Dentro de esta sección se puede navegar por los diferentes apartados, a saber, biestables D, biestables T, biestables JK y flancos de activación.

2.3.1.2 Usuario no registrado

Registrarse en el sistema.

A todos los usuarios que no están registrados en la aplicación se les da la opción de registrarse. Para ello, deben rellenar un formulario en el que deben incluir su nombre de usuario y la contraseña que van a querer utilizar para acceder a la aplicación.

Una vez hecho esto, se incluye al usuario en la base de datos y pasa a ser un usuario registrado en la aplicación Web.

2.3.1.3 Sistema

Abrir sesión de usuario registrado.

Cuando un usuario ya registrado en la aplicación introduce su nombre de usuario y la contraseña correspondiente en el formulario de acceso y hace *click* en el botón “Acceder”, el sistema le otorga acceso ilimitado a todas las funciones de la aplicación.

Cerrar sesión de usuario registrado.

Una vez el usuario termine de utilizar la aplicación, puede hacer *click* en el enlace “Cerrar sesión” para salir de la sesión que ha iniciado en el navegador. De ese modo, el sistema cierra la sesión del actual usuario y permite así que otros usuarios puedan acceder a su sesión en el mismo navegador.

Registrar aciertos y fallos del usuario registrado.

A medida que el usuario registrado vaya intentando resolver los ejercicios y problemas que se le plantean, el sistema irá registrando los aciertos y fallos de este en cada ejercicio y guardándolos en una base de datos.

2.3.2 Requerimientos de interfaces externos

2.3.2.1 Interfaces de usuario

Se pretende que la interfaz de usuario sea sencilla e intuitiva, de modo que se ha optado por el uso de *frames* o marcos para diferenciar claramente las tres zonas de la interfaz principal de la aplicación. Dichas zonas son las siguientes:

- **Barra superior:** Contiene el título del entrenador.
- **Barra lateral izquierda:** Consta de un menú con todas las secciones a las que el usuario registrado puede acceder.

- **Zona central:** En ella se muestran los ejercicios solicitados por el usuario para ser resueltos por este, así como una imagen del biestable y los botones de “Comprobar”, “Mostrar solución” y “Nuevo ejercicio”.

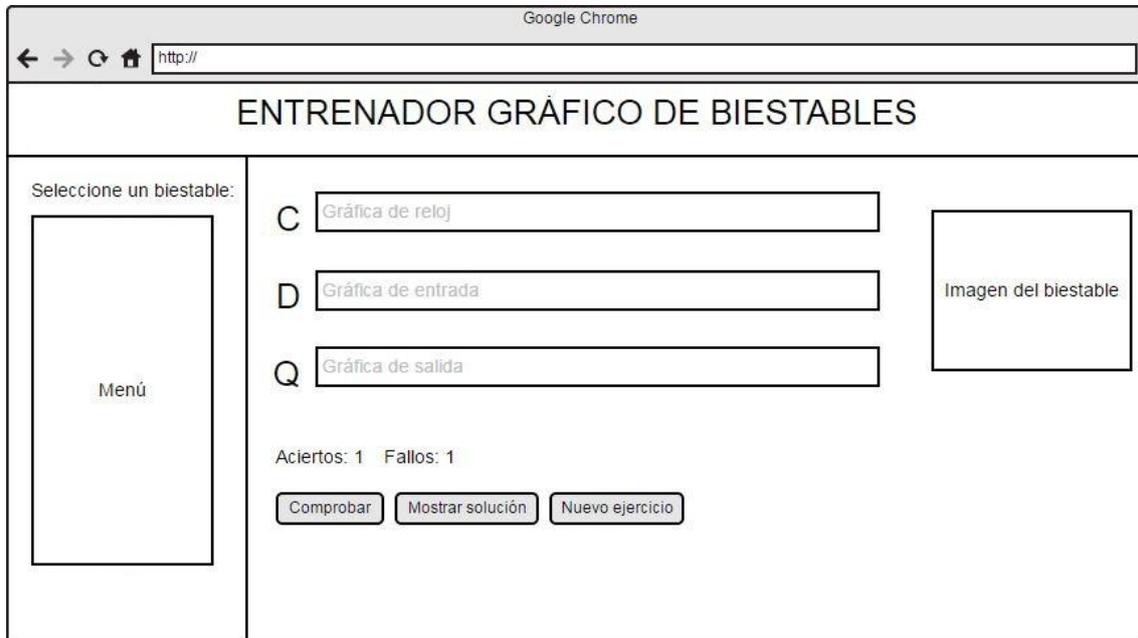


Figura 2.1 Boceto de la interfaz principal del entrenador

Además, la aplicación cuenta con una página de *login* que consta del logotipo de la Universitat Politècnica de València y un formulario para la introducción de usuario y contraseña y su posterior validación.

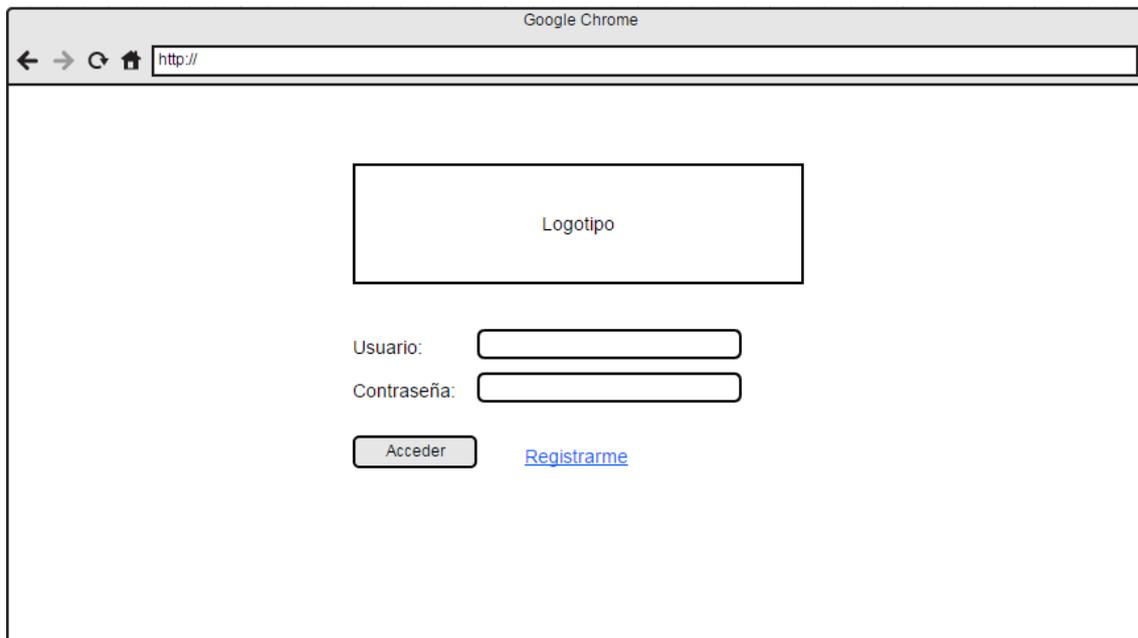


Figura 2.2 Boceto de la interfaz del login

También se dispone de una página de registro donde el usuario no registrado podrá registrarse en la aplicación. Esta página tiene una estructura

extremadamente similar a la página de *login*, dado que consta de un formulario en el que se han de introducir los datos a registrar, a saber, usuario y contraseña.

Finalmente, existe una sección de ayuda en el entrenador en la que, siguiendo la línea de la interfaz principal, se ha optado por el uso de *frames* o marcos diferenciando claramente las siguientes dos zonas:

- **Zona superior:** En ella se encuentra el menú con todas las secciones de ayuda a las que puede acceder.
- **Zona inferior:** En esta zona aparece el contenido de la sección de ayuda que el usuario selecciona en la zona superior.



Figura 2.3 Boceto de la interfaz de ayuda

2.3.2.2 Interfaces hardware

Los usuarios del entrenador deben disponer de acceso a Internet y de un navegador Web que soporte el elemento *canvas* de HTML5.

2.3.2.3 Interfaces software

La aplicación está desarrollada para poder utilizarse en cualquier sistema operativo y navegador Web que soporte el elemento *canvas* de HTML5.

2.3.2.4 Interfaz de comunicación

La interfaz de comunicación entre cliente y servidor es el estándar HTTP a través del protocolo TCP/IP.

2.3.3 Requerimientos de eficiencia

Para la realización de esta aplicación Web se ha utilizado JavaScript, que es un lenguaje que se ejecuta en el cliente. Así pues, el servidor reduce el procesamiento a la identificación del usuario y al registro de aciertos y fallos del usuario registrado.

2.3.4 Restricciones de diseño

No existen restricciones con respecto a estándares ni limitaciones con respecto al hardware.

2.3.5 Atributos

2.3.5.1 Seguridad

Se debe cumplir con la Ley Orgánica de Protección de Datos para llevar el proyecto a la realidad y es por ello que se necesita incluir sistemas de seguridad en el mismo.

Dado que la aplicación utiliza comunicaciones en las que se incluyen datos confidenciales, a saber, usuario y contraseña, estas deben ser protegidas mediante conexiones seguras SSL.

2.3.5.2 Mantenimiento

La aplicación ha sido diseñada para que su mantenimiento sea sencillo. No obstante se recomienda la atenta lectura del presente documento para conocer a fondo el funcionamiento interno del entrenador.

Además, es altamente recomendable hacer una copia de seguridad de la Web y los datos almacenados en la base de datos antes de introducir cambios en la aplicación.



Finalmente, no es necesario que un administrador lleve a cabo el mantenimiento, ya que no hace falta mantener actualizada la información o los servicios del entrenador.

2.3.5.3 Portabilidad

La aplicación Web es compatible con todos los sistemas que sigan los estándares convencionales de navegación Web.

2.3.6 Otros requerimientos

2.3.6.1 Base de datos

El entrenador utiliza MySQL como sistema de gestión de bases de datos. Además, en la base de datos de la aplicación se almacenará información de los usuarios, a saber, nombre de usuario, contraseña y estadísticas de aciertos y fallos para cada tipo de ejercicio.

3 Análisis

Este apartado es necesario para entender correctamente todas las funciones que se pueden realizar en el entrenador, puesto que se describe la funcionalidad de la aplicación y su estructura utilizando para ello diagramas UML.

3.1 Casos de uso

Los diagramas de casos de uso muestran las acciones que se pueden realizar de forma clara, concisa e intuitiva. Además, son necesarios a la hora de entender qué actores participan en el sistema y cómo llevan a cabo las acciones mencionadas anteriormente.

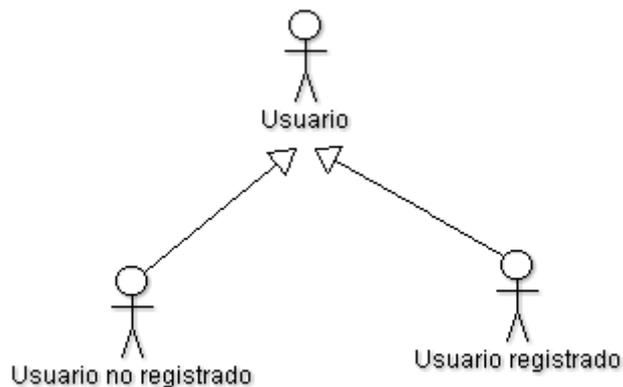


Figura 3.1 Diagrama de usuarios

3.1.1 Casos de uso: Usuario no registrado

En la siguiente figura se muestran los casos de uso para el usuario no registrado. Como se puede observar, las acciones de este tipo de usuario se reducen a registrarse en el sistema.

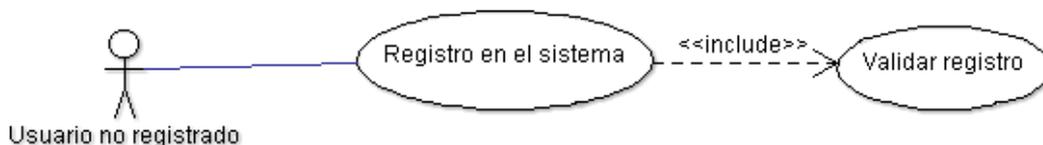


Figura 3.2 Casos de uso del usuario no registrado

3.1.2 Casos de uso: Usuario registrado

A continuación se muestran los casos de uso para el usuario registrado. Este usuario, al estar ya registrado en la aplicación, tiene acceso ilimitado a todas las acciones del entrenador.

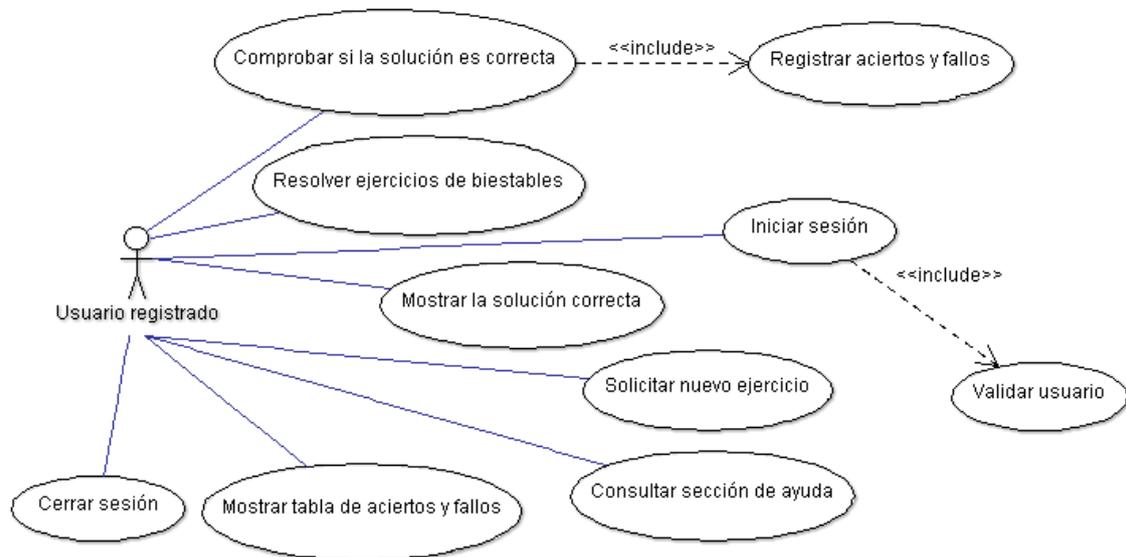


Figura 3.3 Casos de uso del usuario registrado

4 Diseño

4.1 Arquitectura de tres capas

En el presente proyecto se ha optado por utilizar una arquitectura de tres capas. Esta arquitectura separa el proyecto en tres capas bien diferenciadas, a saber capa de presentación, capa de negocio y capa de acceso a datos. A continuación se describen con mayor detalle las capas ya mencionadas.

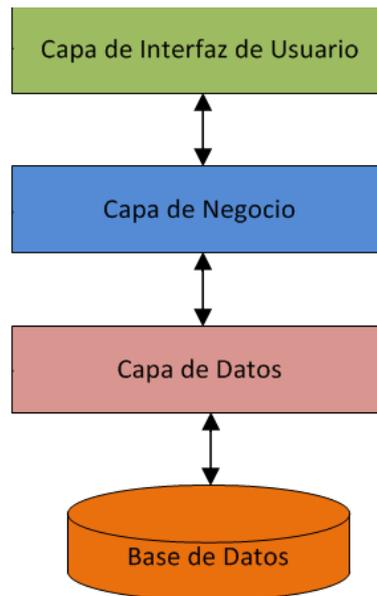


Figura 4.1 Esquema de la arquitectura de tres capas

4.1.1 Capa de presentación

Contiene el código necesario para mostrar al usuario de la aplicación toda la información gráficamente para que este pueda resolver los ejercicios que se le propongan. En ella se encuentran los códigos JavaScript y los de diseño Web, es decir, HTML y CSS.

4.1.2 Capa de negocio

Se comunica con la capa de presentación para recibir sus peticiones y presentarle los resultados y también se comunica con la capa de acceso a datos para almacenar o recuperar datos de la base de datos. En ella se encuentran los códigos PHP que sirven para identificar al usuario y registrar sus estadísticas entre otras funciones.

4.1.3 Capa de acceso a datos

En ella se encuentra el SGBD que se encarga de almacenar los nombres, contraseñas, aciertos y fallos de los usuarios proporcionando así persistencia a la aplicación. En este caso el SGBD utilizado es MySQL.

5 Tecnologías

5.1 Capa de presentación

5.1.1 HTML

HTML, cuyas siglas significan *HyperText Markup Language* es un lenguaje de marcado que se utiliza para el desarrollo de páginas Web.

En primer lugar, se debe entender qué es el hipertexto. El hipertexto es una herramienta estructural y secuencial que permite crear, agregar, enlazar y compartir información de diversas fuentes. Este término es acuñado en 1965 por el señor Ted Nelson.

Además, es necesario saber que un lenguaje de marcado es una forma de codificar un documento mediante etiquetas o marcas que contienen información adicional acerca de la estructura de la información o su presentación.

Dicho esto, el origen de este lenguaje de programación se remonta a 1980, cuando el físico Tim Berners-Lee propone un sistema de hipertexto para compartir documentos. Este sistema une el concepto de hipertexto con el *Standard Generalized Markup Language*, que es un sistema para la organización y etiquetado de documentos.

Así pues, este lenguaje se ha utilizado en este proyecto para describir la estructura y contenido de la Web así como para introducir formularios, botones, imágenes, etc. Además, el proyecto incluye código CSS y JavaScript, de modo que se ha incluido HTML para añadir ese código a la web mediante etiquetas `<style>` y `<script>` respectivamente.

El lenguaje HTML puede ser creado y editado en cualquier procesador de textos básico, por ejemplo, Notepad++, el bloc de notas de Windows o Sublime Text. También puede optarse por una herramienta para edición específica de código HTML como Adobe Dreamweaver, Kompozer o Microsoft FrontPage entre otros.

Finalmente destacar que, para el desarrollo de este proyecto, se ha utilizado la versión cinco de HTML, es decir, HTML5 y como editores se ha utilizado la herramienta Sublime Text.

5.1.2 CSS

CSS, cuyas siglas significan *Cascading Style Sheet*, es un lenguaje utilizado para, una vez creada la estructura de la Web con HTML, XML o XHTML, crear y definir su presentación.



Así pues, la información de este estilo puede ser adjuntada como un documento separado o incluida dentro del mismo código HTML. Si se da este último caso, pueden definirse estilos en la cabecera *<head>* o en cada etiqueta particular con el atributo *style*. Para este proyecto en concreto, se ha decidido utilizar la versión 3 de CSS, es decir, CSS3.

Finalmente, algunos de los motivos por los cuales se ha decidido utilizar hojas de estilo en cascada han sido los siguientes:

- En cualquier momento se puede modificar en parte o en su totalidad el diseño del entrenador modificando únicamente las hojas de estilo sin tener que recurrir a modificar el contenido.
- Al utilizar hojas de estilos independientes del código HTML, ambos códigos van a resultar más claros y limpios lo que se va a traducir en un mantenimiento menos costoso de la aplicación.
- CSS3 mantiene una compatibilidad permanente con respecto a sus predecesores en versión de modo que, los navegadores que no soporten CSS3, no tendrán problemas en asimilar el contenido CSS.

5.1.3 JavaScript

JavaScript es un lenguaje de programación interpretado, es decir, no requiere compilación para ser ejecutado a diferencia de los compilados, que sí necesitan de compilación. Además, este lenguaje sigue el paradigma imperativo de programación, es decir, que la programación se realiza en términos del programa y de sentencias que cambian su estado.

Este lenguaje de programación también sigue el paradigma de la programación orientada a objetos y más concretamente la programación basada en prototipos, que es un estilo de esta.

Finalmente, el entrenador tiene en su código el elemento *<canvas>* y este va a reaccionar a eventos, concretamente *clicks* del ratón del usuario, para dibujar gráficos. Dicho esto, hay que destacar que JavaScript es el único lenguaje disponible para hacer que un sitio Web se comunique con el navegador y observe eventos como *clicks* y es por ello que se ha decidido utilizar JavaScript en esta aplicación Web.

5.2 Capa de negocio

5.2.1 PHP

Las siglas PHP son un acrónimo recursivo que significa *PHP Hypertext Preprocessor*. El creador de esta tecnología fue Rasmus Lerdorf, aunque actualmente la implementación de PHP es producida por The PHP Group.

Además, PHP es un lenguaje de programación interpretado al igual que JavaScript, está diseñado para el desarrollo Web de contenido dinámico y es utilizado para la interpretación del lado del servidor, es decir, procesa las peticiones de un cliente mediante la interpretación de un *script* en el servidor Web para generar respuestas en forma de páginas HTML.

Para finalizar, algunos de los motivos por los que se ha decidido utilizar esta tecnología para el proyecto que en este documento se describe son los siguientes:

- Es un lenguaje multiplataforma y por ello permite operar en varios sistemas operativos.
- Es de código abierto.
- Al ser un lenguaje muy popular goza de gran soporte por lo que, llegado el momento, el mantenimiento de la aplicación sería menos costoso al disponer de más guías y referencias.
- Las cualidades principales de este lenguaje son la estabilidad, la flexibilidad y la velocidad.

5.3 Capa de acceso a datos

5.3.1 MySQL

Es un sistema de gestión de bases de datos relacional multihilo y multiusuario que goza de una gran popularidad. Algunos de los motivos por los que se ha decidido utilizar MySQL en este proyecto son los siguientes:

- La gran velocidad a la que se realizan las operaciones, ya que de ese modo se puede disponer de un mayor rendimiento.
- Soporta gran cantidad de sistemas operativos.
- Su configuración e instalación es relativamente fácil.
- Baja probabilidad de corromper datos aun cuando los errores no se producen en el propio gestor.

6 Implementación

Como se ha explicado anteriormente, la aplicación sigue una arquitectura de tres capas bien diferenciadas.

6.1 Capa de presentación

Contiene todos los componentes y archivos que dan forma a la aplicación, bien de forma gráfica, en el caso de los archivos CSS y HTML o en cuanto a funcionalidad, como es el caso de los archivos JavaScript.

login.html

Es la página encargada de permitir o denegar el acceso por parte de los usuarios a la parte privada de la aplicación.

Si los datos del usuario, es decir, nombre de usuario y contraseña son correctos y coinciden con los almacenados en la base de datos del servidor, se permite el acceso al usuario y se crea una sesión. En caso contrario, se le redirecciona a una página que le indica que los datos introducidos no son correctos.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Usuario:

Contraseña:

[Registrarme](#)

Figura 6.1 Aspecto del fichero login.html

registro.html

Esta página es la encargada de registrar un nuevo usuario en la aplicación. Para ello, este debe introducir el nombre de usuario y la contraseña asociada que desea utilizar. Al hacer *click* en el botón Registrar se creará una entrada en la base de datos con los datos de este usuario.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Usuario:

Contraseña:

Repetir
Contraseña:

[Volver](#)

Figura 6.2 Aspecto del fichero registro.html

principal.html

Es la página inicial y principal de la aplicación, actúa como contenedor de páginas y se divide en tres zonas: título, borde y zona central.

Para la implementación de esta página se ha decidido utilizar *frames* para la división de la misma en tres contenedores de páginas y para que la zona del borde indique a la zona central el ejercicio que debe mostrar.

Además, utilizar este tipo de tecnología hace que el mantenimiento de la aplicación sea menos costoso porque si se desea cambiar la estructura de la página principal, únicamente hay que modificar el fichero de la misma sin tener que modificar otros ficheros.

Finalmente hay que destacar que los ficheros que se cargan por defecto en título, menú y zona central de “principal.html” son “titulo.html”, “borde.html” y “bienvenida.html” respectivamente.

ENTRENADOR GRÁFICO DE BIESTABLES



Figura 6.3 Aspecto por defecto del fichero principal.html

borde.html

Es la página que ocupa el *frame* del borde izquierdo del fichero “principal.html” y sirve a este mismo fichero como panel de control, dado que sus funciones, organizadas en un menú y explicadas a continuación, son muchas.

Por un lado, el menú contiene los nombres de todos y cada uno de los tipos de ejercicio disponibles. Al hacer *click* en cualquiera de ellos se carga, mediante una función en lenguaje JavaScript, la página asociada a dicho ejercicio en la zona central de “principal.html”.

Por otro lado, existe una opción en el menú llamada “Ayuda”. Al hacer *click* sobre esta se abre, mediante un hipervínculo, una nueva pestaña en el navegador con una sección de ayuda.

Finalmente se puede encontrar una opción que permite cerrar la sesión que el usuario ha iniciado en la aplicación. Al hacer *click* sobre ella, se ejecuta una función JavaScript que cierra la sesión y devuelve al usuario a la página “login.html”.

Seleccione un biestable

D por flanco de subida
D por flanco de bajada
J-K por flanco de subida
J-K por flanco de bajada
T por flanco de subida
T por flanco de bajada
Mostrar resultados
Cerrar sesión
Ayuda

Figura 6.4 Aspecto de borde.html

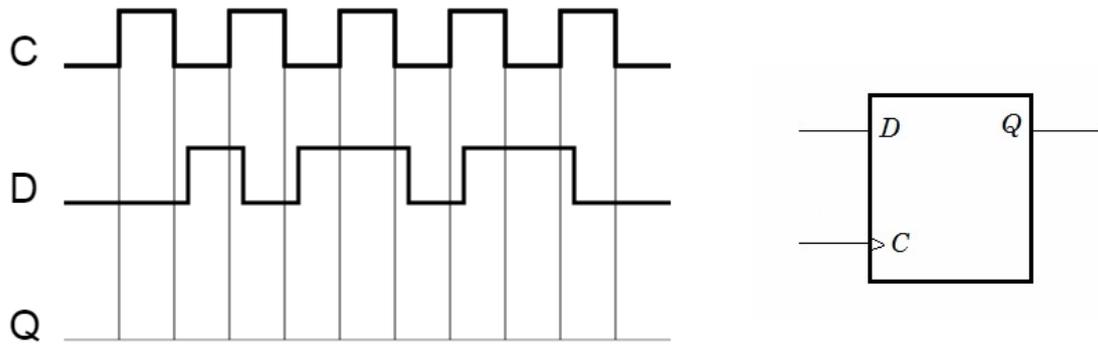
titulo.html

Esta página ocupa el *frame* superior del fichero “principal.html” y únicamente sirve para sintetizar, en pocas palabras, en qué consiste la aplicación.

biestableDSubida.html

Esta es la página que ocupa el *frame* central de “principal.html” cuando se ha elegido la opción “D por flanco de subida” en el menú lateral. En ese *frame* central se carga un ejercicio de biestables D activos por flanco de subida y el usuario puede proceder a resolver el ejercicio propuesto. En la figura pueden verse las líneas de referencia que guían al usuario en la ubicación de los flancos de reloj, tanto de subida como de bajada, pues es responsabilidad del usuario saber en cuáles puede cambiar la salida del biestable.

Estas líneas de referencia delimitan las áreas donde el usuario puede hacer click con el ratón para indicar que quiere poner la salida, en ese semiperiodo, a nivel alto o nivel bajo.



Aciertos: Fallos:

Comprobar Mostrar Solución Nuevo Ejercicio

Figura 6.5 Aspecto de un ejercicio de biestables D activos por flanco de subida.

Como puede apreciarse en la figura anterior, al usuario registrado se le muestran dos gráficas, una correspondiente a la señal de reloj y otra a la señal de entrada del biestable, y una gráfica correspondiente a la señal de salida que, en principio está vacía y que debe ser dibujada por el usuario.

Se muestran también tres botones: “Comprobar”, “Mostrar Solución” y “Nuevo Ejercicio”.

Además, este fichero está enlazado con una serie de ficheros con código JavaScript, que son los que permiten que el usuario registrado pueda realizar todas las funciones del entrenador.

Para finalizar, los ficheros HTML correspondientes a los ejercicios sobre el resto de biestables son extremadamente similares a este salvo por las pequeñas diferencias propias entre ejercicios, de modo que su funcionalidad queda ya sintetizada en este apartado. Los ficheros del resto de ejercicios son: “biestableDBajada.html”, “biestableJKSubida.html”, “biestableJKBajada.html”, “biestableTSubida.html” y “biestableTBajada.html”.

ayuda.html

Es la página principal de la sección de ayuda de la aplicación. Del mismo modo que “principal.html”, esta página está formada por *frames*, aunque en esta ocasión se organizan de un modo distinto puesto que esta vez existen dos *frames* únicamente.

La página consta de un menú en el *frame* de la parte superior y de un área de presentación de la información en el *frame* de la parte inferior. La selección de

una opción del menú hace que se muestre una información u otra en el *frame* inferior permitiendo así al usuario informarse de aquello que más le interese.

Flancos de activación

Un flanco de subida es un instante de tiempo en el que la señal de reloj *C* cambia de 0 a 1. Se indica en el biestable con un triángulo en la entrada de reloj.

Un flanco de bajada es un instante de tiempo en el que la señal de reloj *C* cambia de 1 a 0. Se indica en el biestable con un triángulo con círculo en la entrada de reloj.

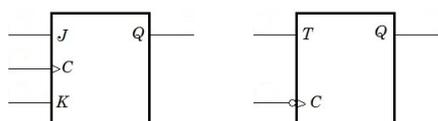


Figura 6.6 Aspecto de *ayuda.html* con el apartado “Flancos de activación” seleccionado.

Finalmente, hay que destacar que la sección de ayuda ofrece información sobre biestables D, biestables T, biestables JK y flancos de activación.

funciones.js

Contiene todas las funciones necesarias para generar ejercicios de biestables D y T. Estas funciones son:

- graficaReloj: dibuja la señal de reloj en el elemento *canvas* que constituye el ejercicio.
- dec2bin: recibe un valor aleatorio entre 1 y 1023, lo pasa a binario y guarda el resultado en un vector de diez elementos.
- graficaEntrada: genera un valor aleatorio entre 1 y 1023, llama a la función “dec2bin” pasándole ese valor como argumento, recibe un vector de diez elementos y dibuja la señal de entrada en función de ese vector.
- lineasReferencia: dibuja las líneas de referencia de las gráficas para que el usuario sepa dónde se encuentran los flancos de la señal de reloj, tanto los flancos de subida como de bajada.
- baseSalida: puesto que la gráfica de la señal de salida debe dibujarla el usuario, se ha optado por general una señal inicial plana y esa señal es dibujada por esta función.
- borrarSalidaAnterior: deja en blanco la señal de salida para dibujar una nueva señal en función de un vector que contiene el valor binario de la salida.
- continuarLineas: al borrar la señal de salida, también se borra el trozo de líneas de referencia de la zona de la señal. Es por ello que esta función se encarga de que ese trozo vuelva a aparecer.

- escribir: hace aparecer el número de aciertos y fallos actuales justo debajo de las gráficas del ejercicio.
- enviarEstadistica: envía los aciertos y fallos de un tipo de ejercicio a la página “registrarejercicio.php” para que esta los registre en la base de datos.

funcionesJK.js

Los ejercicios para el biestable JK son similares a los de los biestables D y T, pero no son exactamente iguales. Algunas de estas diferencias son el número de gráficas que deben mostrarse (cuatro en lugar de tres), la longitud de las líneas de referencia y el área de dibujo de la gráfica de entrada, entre otras diferencias.

Por eso el fichero “funcionesJK.js” tiene las mismas funciones que el fichero “funciones.js”. Estas funciones comparten su estructura y funcionalidad, pero tienen diferencias significativas en el detalle de la implementación. Implementarlas en un fichero separado hace que la organización y mantenimiento de la aplicación sea más sencillo.

scriptDSubida.js y equivalentes para cada biestable

Estos ficheros contienen tres vectores de diez elementos que representan la gráfica de reloj, la gráfica de entrada y la gráfica de salida, puesto que contienen el valor binario de estas con cada dígito en una posición del vector. Cabe destacar que en el caso de los biestables JK las gráficas son cuatro y dos de ellas están dedicadas a las entradas J y K.

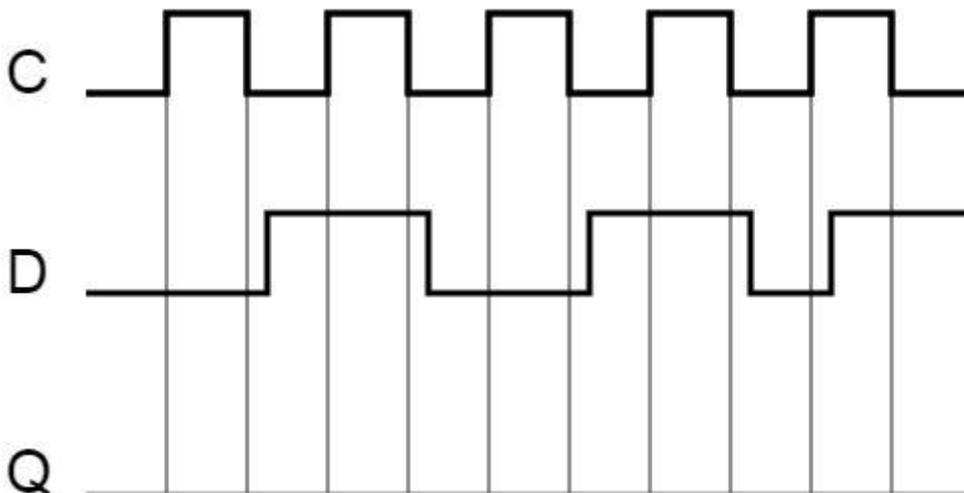


Figura 6.7 Representación gráfica de la señal de entrada D cuyo vector asociado contiene el valor binario 0110011011

Además, dispone de variables para contar los aciertos y fallos en el ejercicio, así como las siguientes funciones:

- salidaQ: esta función recibe como parámetro un evento capturado al hacer *click* en cualquier punto de la gráfica de salida del elemento *canvas* y llama a la función llamada “dibujar()”. Así pues, detecta el semiperiodo en el que se ha pulsado y cambia su valor en el vector de la gráfica de salida para ser dibujado con posterioridad.
- dibujar: se encarga de borrar la gráfica de salida existente y de dibujar la nueva gráfica en función del nuevo valor del vector de salida. Al acabar, llama a la función “continuarLineas()” mencionada anteriormente.
- comprobar: utiliza los tres vectores mencionados anteriormente para comprobar si la solución ofrecida es correcta. Además, aumenta el contador de aciertos y fallos en función del resultado y llama a la función “escribir()” para mostrar los resultados actuales en el elemento *canvas*.
- solucionar: si finalmente el usuario no logra resolver el ejercicio, esta función lo resuelve por él. Toma como referencia los vectores de reloj y de entrada para calcular el vector de salida y dibujarlo posteriormente.
- imagenBiestable: incluye una imagen del biestable correspondiente al ejercicio para evitar confusiones por parte del usuario.
- letrasCanvasD: indica el nombre de la gráfica a la izquierda de la misma. En el caso que se indica, se muestran las letras de los ejercicios de biestables D.

menu.js y borde.js

Como se comentó anteriormente en este documento, tanto la página “principal.html” como la página “ayuda.html” están formadas por *frames* y que cada una de ellas contiene un *frame* específico que contiene un menú para controlar el resto de *frames*.

Así pues, estos ficheros JavaScript contienen el código necesario para que, desde ese *frame* ya mencionado, el usuario pueda elegir un elemento del menú y que la información elegida aparezca en la pantalla.

Finalmente, el fichero “borde.js” contiene una opción con el código necesario para que un usuario registrado pueda cerrar la sesión iniciada previamente.

Páginas CSS

La aplicación cuenta con una serie de ficheros CSS que se encargan de definir y crear la interfaz gráfica de las páginas “login.html”, “bienvenida.html”, “borde.html”, “menu.html”, “titulo.html” y “registro.html”.

6.2 Capa de negocio

Consta de una serie de ficheros escritos en lenguaje PHP que van a permitir a la aplicación recibir una serie de peticiones a la base de datos por parte de la capa de presentación antes mencionada y atenderlas, produciendo un cambio en la base de datos y/o devolviendo un resultado.

Cabe destacar que, a diferencia de las funciones JavaScript descritas en el apartado anterior que se ejecutan en el cliente, las funciones PHP se ejecutan en el servidor de la aplicación.

db.php

Contiene la conexión al sistema de gestión de base de datos MySQL. Se utiliza para acceder a la base de datos y así leer o escribir datos en ella y además se incluye en todas las páginas que requieren conexión a la base de datos.

login.php

Se encarga de recibir el nombre de usuario y la contraseña de la página “login.html” y con ellos realiza una consulta a la base de datos para comprobar si existe ese usuario y su contraseña asociada es la correcta.

De ser así, se inicia una sesión con el nombre de usuario como identificador y se le redirige a la página “principal.html”. En caso contrario, se le indica al usuario que no se ha podido acceder a la aplicación.

registro.php

Recibe el nombre y la contraseña del nuevo usuario de la página “registro.html” para registrar ese nuevo usuario en la base de datos. Previamente se realiza una serie de controles como que el usuario no exista ya en la base de datos, que la longitud mínima del nombre de usuario y/o contraseña sea de seis caracteres, etc.

Finalmente, y una vez pasadas todas las comprobaciones de forma satisfactoria, se registra el nuevo usuario en la base de datos con sus aciertos y fallos en el sistema con valor inicial cero y se le redirige a la página “login.html” de nuevo. En caso contrario, se indica al usuario que algo ha fallado en el registro.

logout.php

Se encarga de destruir la sesión iniciada y de redirigir al usuario registrado nuevamente a la página de *login*.

bienvenida.php

Muestra un mensaje de bienvenida en la página “principal.html” al usuario registrado que acaba de iniciar sesión.

registrarejercicio.php

Cada vez que un usuario registrado hace *click* en el botón “Comprobar” de un ejercicio propuesto, esta página registra los aciertos y fallos de este en la base de datos sumándolos a los que ya había.

resultados.php

Muestra al usuario registrado una tabla con los aciertos y fallos que ha tenido a lo largo del tiempo en todos y cada uno de los ejercicios disponibles en la aplicación.

6.3 Capa de acceso a datos

El entrenador dispone de una base de datos llamada “biestables.sql” formada por una tabla llamada “alumnos” en la que se almacenan todos aquellos usuarios que se registran en la aplicación así como sus aciertos y fallos en cada uno de los ejercicios.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
1	<u>id</u>	int(100)			No	Ninguna	AUTO_INCREMENT
2	usuario	varchar(100) latin1_swedish_ci			No	Ninguna	
3	password	varchar(100) latin1_swedish_ci			No	Ninguna	
4	aciertos1	int(100)			No	Ninguna	
5	fallos1	int(100)			No	Ninguna	
6	aciertos2	int(100)			No	Ninguna	
7	fallos2	int(100)			No	Ninguna	
8	aciertos3	int(100)			No	Ninguna	
9	fallos3	int(100)			No	Ninguna	
10	aciertos4	int(100)			No	Ninguna	
11	fallos4	int(100)			No	Ninguna	
12	aciertos5	int(100)			No	Ninguna	
13	fallos5	int(100)			No	Ninguna	
14	aciertos6	int(100)			No	Ninguna	
15	fallos6	int(100)			No	Ninguna	

Figura 6.8 Estructura de la tabla “alumnos” de la base de datos “biestables.sql”

Como se puede observar en la figura anterior, la tabla “alumnos” dispone de una serie de campos que, para mayor claridad, pasan a ser explicados a continuación:

- id: Este campo es el identificador de cada entrada en la tabla. Es la clave primaria de dicha tabla y además es autoincremental, es decir, aumenta una unidad por cada entrada nueva.
- usuario: Es el campo encargado de almacenar el nombre de usuario con el que se ha registrado un alumno en el sistema.

- **password**: Este campo contiene la contraseña asociada a un alumno registrado en el sistema.
- **aciertosX y fallosX**: Son los campos encargados de almacenar los aciertos y fallos de un usuario en un determinado ejercicio. A continuación se muestra una tabla con las correspondencias entre los números y su tipo de ejercicio asociado.

Número	Tipo de ejercicio
1	Biestables D activos por flanco de subida
2	Biestables D activos por flanco de bajada
3	Biestables T activos por flanco de subida
4	Biestables T activos por flanco de bajada
5	Biestables JK activos por flanco de subida
6	Biestables JK activos por flanco de bajada

Tabla 6.1 Correspondencias entre número y ejercicio de los campos aciertos y fallos de la tabla “alumnos”.

6.4 Funcionalidad esencial

La aplicación dispone de una serie de archivos divididos por capas debido a la arquitectura software que se ha aplicado y que se comunican con los archivos de la propia capa o con archivos de otras capas tal y como se explica a continuación de forma simplificada.

En primer lugar las páginas “login.html” y “registro.html” son las encargadas de registrar nuevos usuarios en el sistema y para ello se comunican con sus respectivos archivos en la capa de negocio, es decir, “login.php” y “registro.php”.

En segundo lugar la página “principal.html” es aquella a la que se redirige al usuario una vez ha accedido correctamente a la aplicación. En ella se carga una página a la que, para simplificar, llamaremos “biestableX.html” siendo X el tipo de ejercicio que quiere resolver el usuario.

Además, la página “biestableX.html” cuenta con los archivos “funciones.js” y “scriptBiestableX.js” que contienen toda la funcionalidad específica y común del ejercicio.

En tercer lugar el archivo “funciones.js” se comunica con el archivo de la capa de negocio llamado “registrarejercicio.php” para acceder a la base de datos y almacenar los aciertos y fallos que el usuario está teniendo en un determinado ejercicio. Del mismo modo, y cuando el usuario lo solicita, la página “principal.html” se comunica con el archivo “resultados.php” de la capa de

negocio y muestra una tabla con los resultados acumulados del usuario registrado.

En cuarto lugar la página “principal.html” puede, a petición del usuario, cerrar la sesión del mismo accediendo al archivo “logout.php” de la capa de negocio.

Finalmente todos los archivos de la capa de negocio que necesitan acceder a la base de datos se comunican con el archivo “db.php” que contiene la conexión a la misma, es decir, el acceso a los datos almacenados.

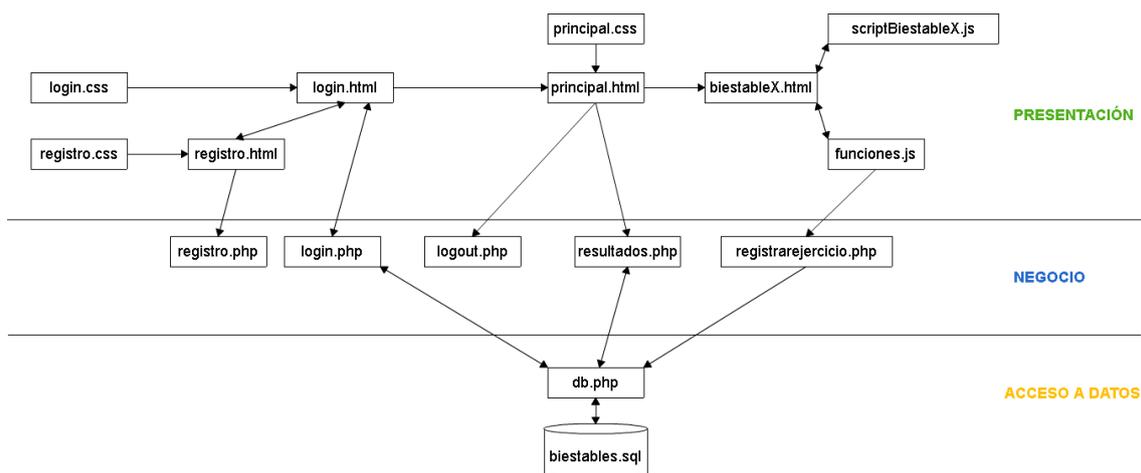


Figura 6.9 Esquema de la funcionalidad esencial del entrenador gráfico de biestables

7 Evaluación y pruebas

En esta etapa de evaluación y pruebas concluye el ciclo de vida de este proyecto software y, además, deja el producto preparado para su uso.

Así pues, en esta etapa no sólo se mide el nivel de calidad que ofrece la aplicación, también se comprueba que el producto funciona correctamente.

7.1 Prueba de validación de CSS

Se ha utilizado varios ficheros con código CSS para crear la interfaz de la aplicación Web y es por ello que se debe validar ese código. Para ello se ha utilizado el validador de la W3C y se ha ido corrigiendo el código hasta que ha pasado satisfactoriamente todas las pruebas.

Así pues, cabe destacar que el código CSS ha sido comprobado con la versión 3 como se muestra a continuación.

Finalmente, el enlace a la página donde se ha validado el código CSS es el siguiente: <http://jigsaw.w3.org/css-validator/>



Resultados del Validador CSS del W3C para registro.css (CSS versión 3)

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

Figura 7.1 Validación de registro.css

Resultados del Validador CSS del W3C para login.css (CSS versión 3)

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

Figura 7.2 Validación de login.css

Resultados del Validador CSS del W3C para bienvenida.css (CSS versión 3)

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

```
<p>
  <a href="http://jigsaw.w3.org/css-validator/check/referer">
    
  </a>
</p>
```

Figura 7.3 Validación de bienvenida.css

Resultados del Validador CSS del W3C para borde.css (CSS versión 3)

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/check/referer">  
  
</a>  
</p>
```

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/check/referer">  
  
</a>  
</p>
```

Figura 7.4 Validación de borde.css

Resultados del Validador CSS del W3C para menu.css (CSS versión 3)

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/check/referer">  
  
</a>  
</p>
```

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/check/referer">  
  
</a>  
</p>
```

Figura 7.5 Validación de menu.css

Resultados del Validador CSS del W3C para titulo.css (CSS versión 3)

¡Enhorabuena! No error encontrado.

¡Este documento es [CSS versión 3](#) válido!

Puede mostrar este icono en cualquier página que valide para que los usuarios vean que se ha preocupado por crear una página Web interoperable. A continuación se encuentra el XHTML que puede usar para añadir el icono a su página Web:

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/check/referer">  
  
</a>  
</p>
```

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/check/referer">  
  
</a>  
</p>
```

Figura 7.6 Validación de titulo.css

7.2 Prueba de validación de navegadores web

Se ha comprobado que el entrenador funciona correctamente en los navegadores web Mozilla Firefox y Google Chrome.



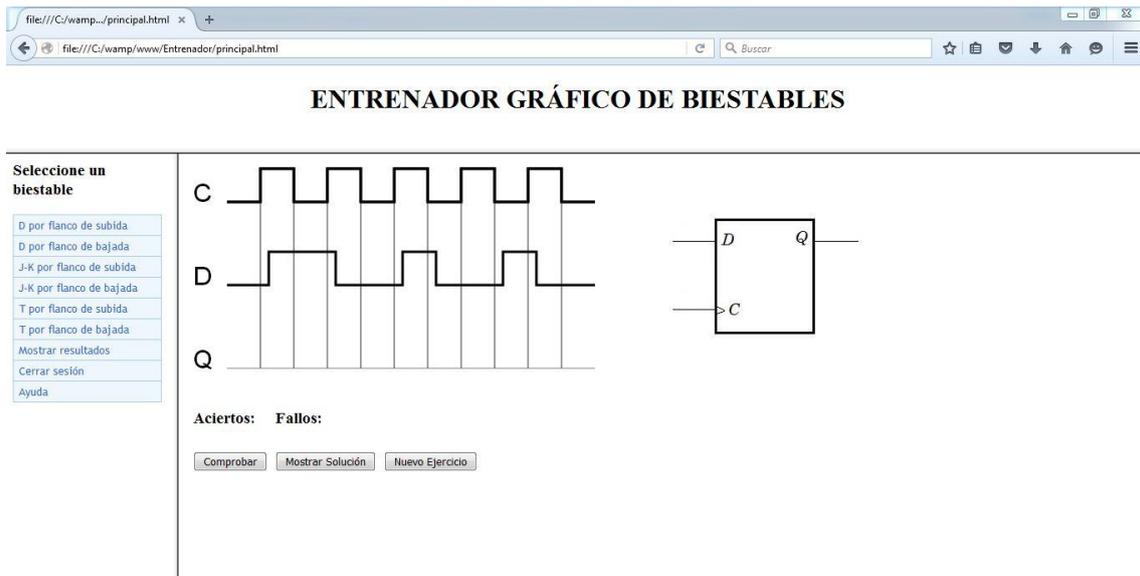


Figura 7.7 Validación en Mozilla Firefox.

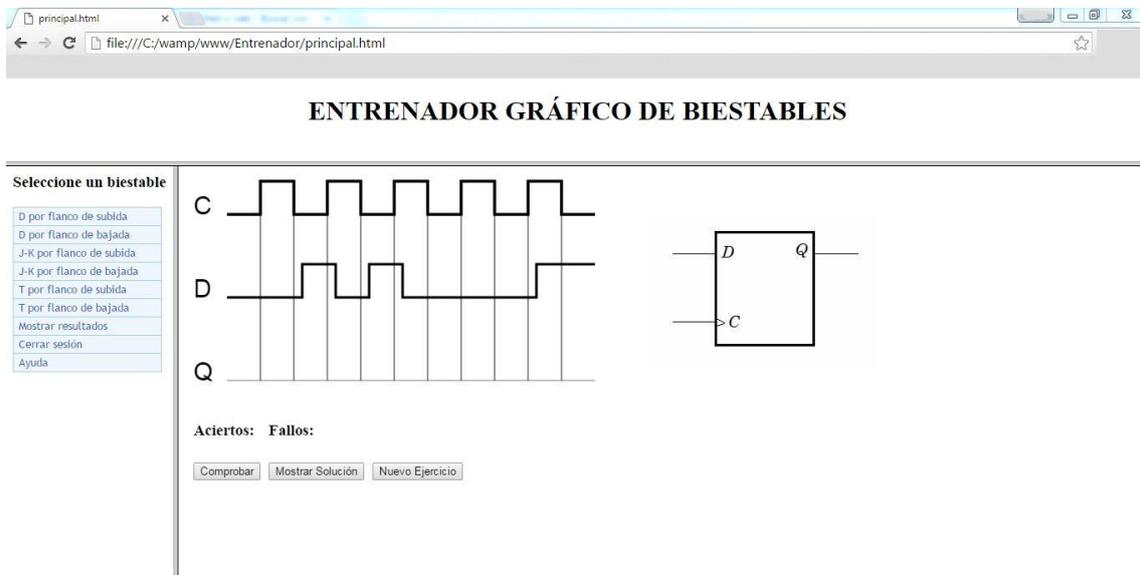


Figura 7.8 Validación en Google Chrome.

7.3 Prueba de seguridad de acceso

En apartados anteriores se ha explicado que para acceder a todas las funcionalidades de la aplicación, un usuario debe registrarse e introducir su nombre de usuario y contraseña en la página de acceso, también conocida como “login.html”. En caso contrario, el usuario no podrá acceder a las funcionalidades principales del entrenador.

Además, si el usuario introduce mal su nombre de usuario y/o contraseña en la página de acceso, se le indicará que su acceso ha fallado tal y como se muestra a continuación.



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Usuario:

Contraseña:

[Registrarme](#)

Figura 7.9 Página de acceso.



Login fallido

Figura 7.10 Error de acceso tras introducir un usuario y contraseña incorrectos.

8 Conclusiones

8.1 Trabajo realizado

El proyecto cumple con la especificación de requisitos expuesta en el apartado correspondiente de este documento y es por ello que se puede afirmar que se ha cumplido con las expectativas puestas en esta aplicación web.

Además, el entrenador es muy sencillo de utilizar, muy intuitivo y ofrece secciones tales como la sección de ayuda que permiten al usuario aprender de forma gradual acerca de los biestables D, T y JK.

Finalmente, y puesto que principalmente el usuario de la aplicación al que va dirigido esta aplicación es el alumno de la asignatura “Fundamentos de Computadores”, se puede concluir que la aplicación cumple su función.

8.2 Valoración personal

El desarrollo de este proyecto ha supuesto todo un reto para mí, puesto que ha sido el momento en el que he tenido que sacar todos mis conocimientos, técnicas de organización, metodologías aprendidas y valentía para sacarlo adelante.

En primer lugar hay que destacar que elegí este proyecto web sin tener apenas idea de programación web y aquí viene el primer reto, porque he tenido que aprender mucho sobre lenguajes como JavaScript, CSS, HTML5, PHP, etc.

En segundo lugar, nunca me he enfrentado a un proyecto software tan grande y que ha supuesto tanto tiempo y dedicación para mí. Esto es algo muy positivo, puesto que he podido sobrepasar mis límites y he descubierto que, a pesar de las adversidades, si le pongo ganas puedo conseguirlo.

Finalmente, me siento muy orgulloso del producto final que he conseguido y que me gustaría que fuera una ayuda y una buena herramienta para aquellos alumnos que tienen dificultades, así como para todos aquellos cuyas ganas de aprender nunca cesan.

8.3 Futuras mejoras

Este proyecto, aunque totalmente funcional, puede incluir una serie de mejoras en el futuro. A continuación se expone un listado con las mismas:

- Inclusión de ejercicios de biestables con entradas *CLEAR* y *RESET*.
- Aumento del tamaño de las gráficas de los ejercicios de biestables.

- Sistema de *cookies* para recordar el nombre de usuario y la contraseña en la aplicación y evitar que el usuario tenga que introducir sus datos cada vez que entre a la página.
- Permitir al usuario dibujar también la señal o señales de entrada creando así ejercicios personalizados.

9 Bibliografía

Biestable. *Biestable* - *Wikipedia, La Enciclopedia Libre*. Retrieved from <https://es.wikipedia.org/wiki/Biestable>

BIESTABLES. Retrieved from <http://www.profesormolina.com.ar/electronica/componentes/int/biest.htm>

1.2. *Biestable D*. Retrieved from http://educativa.catedu.es/44700165/aula/archivos/repositorio//4750/4922/html/12_biestable_d.html

Caso de uso. *Caso De Uso* - *Wikipedia, La Enciclopedia Libre*. Retrieved from https://es.wikipedia.org/wiki/Caso_de_uso

Calle, F., Aguilar, J., Florez, Y., Inéz, M., Ruiz, D. Rodríguez, & Lútz, M. *Arquitectura 3 Capas. Arquitectura 3 Capas*. Retrieved from <http://es.slideshare.net/Decimo/arquitectura-3-capas>

¿Qué Es Y Para Qué Sirve HTML? El Lenguaje Más Importante Para Crear Páginas Webs. *HTML Tags (CU00704B)*. Retrieved from http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=435:que-es-y-para-que-sirve-html-el-lenguaje-mas-importante-para-crear-paginas-webs-html-tags-cu00704b&catid=69:tutorial-basico-programador-web-html-desde-cero&Itemid=192

HTML. *HTML* - *Wikipedia, La Enciclopedia Libre*. Retrieved from <https://es.wikipedia.org/wiki/HTML>

SGML. *SGML* - *Wikipedia, La Enciclopedia Libre*. Retrieved from <https://es.wikipedia.org/wiki/SGML>

Lenguaje de marcado. *Lenguaje De Marcado* - *Wikipedia, La Enciclopedia Libre*. Retrieved from https://es.wikipedia.org/wiki/Lenguaje_de_marcado

1.2. Breve historia de HTML. *1.2. Breve Historia De HTML (Introducción a XHTML)*. Retrieved from http://librosweb.es/libro/xhtml/capitulo_1/breve_historia_de_html.html

Hoja de estilos en cascada. *Hoja De Estilos En Cascada* - *Wikipedia, La Enciclopedia Libre*. Retrieved from https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada

10 razones para usar CSS. *10 Razones Para Usar CSS*. Retrieved from <http://www.maestrosdelweb.com/usarcss/>

JavaScript. *JavaScript* - *Wikipedia, La Enciclopedia Libre*. Retrieved from <https://es.wikipedia.org/wiki/JavaScript>

PHP. *PHP* - *Wikipedia, La Enciclopedia Libre*. Retrieved from <https://es.wikipedia.org/wiki/PHP>

MySQL. *MySQL* - *Wikipedia, La Enciclopedia Libre*. Retrieved from <https://es.wikipedia.org/wiki/MySQL>



IEEE Recommended Practice for Software Requirements Specifications, IEEE Std 830-1998.

Gulbransen, D. y Rawlings, K. (1998). *HTML Dinámico*. Madrid: Prentice-Hall.

Welling, L. y Thomson, L. (2008). *PHP and MySQL Web development* (4a. ed.). Addison-Wesley Professional.

Fulton, S. y Fulton, J. (2011). *HTML 5 Canvas*. Madrid: Anaya Multimedia.

Colomé Fornós, X. (2007). *CSS*. Madrid: Anaya Multimedia.

Flanagan, D. (2011). *JavaScript: the definitive guide* (6a. ed.). Sebastopol: O'Reilly Media.

Anexos

A. Herramientas utilizadas

a. ArgoUML

ArgoUML es una aplicación programada en lenguaje Java que permite la creación de diagramas UML. Algunos de los motivos y ventajas por los que se ha decidido utilizar esta herramienta para la creación de diagramas en el proyecto han sido los siguientes:

- Proporciona soporte para la generación de código PHP, Java, C++, C# y Python.
- Consume pocos recursos.
- Genera ficheros gráficos desde los diagramas creados por el usuario.
- Es multilinguaje.



Figura A Logotipo de ArgoUML.

b. Moqups

Es una aplicación web que permite al usuario crear prototipos para las interfaces gráficas del proyecto.

Se ha elegido esta aplicación por su extensa librería gráfica, así como su facilidad de uso y por otros aspectos como el parecido de sus elementos gráficos con las interfaces deseadas para el entrenador gráfico.

Finalmente, cabe destacar que esta aplicación es totalmente gratuita y que no requiere de registro.



Figura B Logotipo de Moqups

c. Sublime Text

Sublime Text es una herramienta orientada a editar textos y también código fuente.

Para el desarrollo de este proyecto se ha utilizado esta herramienta debido a los siguientes motivos:

- Soporta una gran cantidad de lenguajes de programación entre los que se encuentran los utilizados para el desarrollo del entrenador.
- Dispone de una versión de evaluación plenamente funcional y sin fecha de caducidad.
- Contiene una barra lateral con la previsualización de la estructura del código, lo que hace más fácil encontrar un determinado fragmento del mismo, especialmente cuando este es muy extenso.
- El editor resalta las expresiones propias de la sintaxis del código, cosa que facilita su lectura.

Finalmente hay que destacar que se ha utilizado la versión 2.0 de esta herramienta para el desarrollo de este proyecto.



Figura C Logotipo de Sublime Text

d. WampServer

WampServer es un entorno de desarrollo web que nos permite disponer de un servidor en nuestro host local, es decir, un servidor local.

La aplicación descrita en el presente documento está basada en una arquitectura de tres capas y utiliza archivos PHP para el acceso a una base de datos, de modo que es necesario ejecutarla en un entorno como WampServer puesto que los archivos PHP se ejecutan en el servidor y la base de datos se aloja en el mismo.

Finalmente, este entorno cuenta con una serie de características por las que se ha decidido su uso:

- Utiliza Windows como sistema operativo, necesario puesto que la aplicación se ha desarrollado en Windows 7.
- Se utiliza PHP como lenguaje de programación entre otros.
- Utiliza MySQL como sistema gestor de bases de datos.
- Utiliza Apache como servidor web.

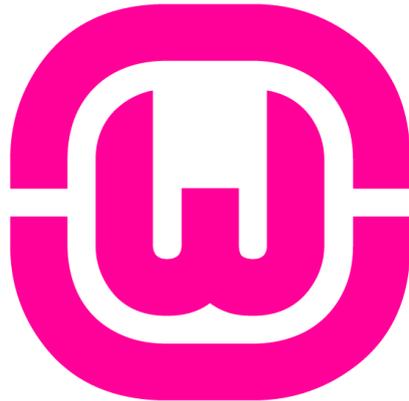


Figura D Logotipo de WampServer