



**Escuela Superior
de Ingeniería y Tecnología**
Universidad de La Laguna

Trabajo de Fin de Grado

Ingeniería Informática

Pensamiento Computacional: Sistema de Monitorización de Actividades Desenchufadas

*Computational Thinking: Monitoring System for
Unplugged Activities*

Daniel A. Melián León

La Laguna, 10 de septiembre de 2020

Dña. Coromoto León Hernández, con N.I.F. 78.605.216-W profesora Catedrática de Universidad del área de Lenguajes y Sistemas Informáticos, adscrita al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor,

Dña. Gara Miranda Valladares, con N.I.F. 78.563.584-T profesora Titular de Universidad del área de Lenguajes y Sistemas Informáticos, adscrita al departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor,

C E R T I F I C A (N)

Que la presente memoria titulada:

“Pensamiento Computacional: Sistema de Monitorización de Actividades Desenchufadas”

ha sido realizada bajo su dirección por D. **Daniel Alberto Melián León** con N.I.F. 54060332M.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 10 de septiembre de 2020

Agradecimientos

*A Javier Castro, por acompañarme durante todo este viaje,
apoyándonos desde el primer hasta el último día.*

*A mi familia y amigos, por estar siempre ahí, dándome
fuerzas para continuar.*

A mis tutoras, Dña. Coromoto León Hernández y Dña. Gara
Miranda Valladares, por estar ahí siempre para ayudarme a
sacar adelante este trabajo.

A todos mis compañeros y compañeras del Grado, que han
hecho que este largo viaje no lo parezca tanto.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial 4.0 Internacional.

Resumen

El pensamiento computacional puede definirse como el proceso mediante el cual se utilizan herramientas y técnicas de la informática para entender y razonar sobre todos los procesos que nos rodean, ya sean naturales o artificiales.

Partiendo de esa definición, el objetivo de este trabajo ha sido desde un principio aportar un granito de arena al gran desarrollo que está teniendo la rama del Pensamiento Computacional en estos años. Para ello, y tras un proceso de investigación hemos decidido crear un sistema de evaluación de actividades de pensamiento computacional desenchufado, con el cual facilitar a los profesores la evaluación y el almacenamiento de las calificaciones en todos los eventos relacionados con el pensamiento computacional.

Este sistema explicado en detalle en el siguiente documento permite crear eventos en el ámbito del pensamiento computacional, crear participantes y monitores para dichos eventos y finalmente, asignar actividades de Pensamiento Computacional Desenchufado recopiladas en una base de datos a esos participantes, las cuales serán evaluadas por el monitor previamente especificado.

Todo ello, con el fin de extender estas prácticas y de tener una recopilación de datos que nos refleje como avanza el pensamiento computacional junto al desarrollo tecnológico.

Cabe destacar, que este sistema, diseñado en primera instancia para la gestión de actividades dentro del ámbito del pensamiento computacional, podrá extenderse para la gestión de actividades en cualquier otro ámbito docente.

Este proyecto ha sido desarrollado con Xamp, tecnología que nos ha permitido trabajar con el gestor de bases de datos Mysql, el servidor web apache, y que, además, incluye el interprete de PHP, lenguaje, que junto con JavaScript y HTML, conforman el proyecto.

Palabras clave: Pensamiento Computacional, Actividades Desenchufadas, Educación, Tecnología.

Abstract

Computational thinking can be defined as the process by which computer science tools and techniques are used to understand and reason about all the processes that surround us, whether natural or artificial.

Based on this definition, the objective of this project has been from the outset to contribute to the great development that the Computational Thinking in having in these years. To do this, and after a great deal of research we have decided to create an evaluation system for unplugged computational thinking activities. To make it easier for teachers to assess and preserve grades in this field.

This system explained in detail in the following document allows to create events in the field of computational thinking, add participants and monitors and finally, assign unplugged computational thinking activities collected in a database to those participants, which will be evaluated by the previously specified monitor.

All this, to extend these practices and to have a compilation of data that reflects to us how the technological thinking evolves along with technological development.

It should be noted, that this system, designed in first instance for the management of activities within the field of computational thinking, may be extended to the management of activities in any other teaching field.

This project has been developed with Xamp, a technology that has allowed us to work with the Mysql database manager, the Apache web server, and which also includes de PHP interpreter, language, which make up the project together with HTML and JavaScript.

Keywords: Computational thinking, Unplugged Activities, education, technology.

Índice general

Capítulo 1 Introducción.....	1
1.1 Pensamiento Computacional.....	1
1.2 Antecedentes	2
1.3 Objetivos	2
Capítulo 2 Diseño del Sistema Software.....	3
2.1 Arquitectura del Software.....	3
2.2 Herramientas de Desarrollo de Software	4-7
2.3 Prototipo	7-11
Capítulo 3 Modo de Uso.....	12
3.1 Estructura del Sistema	12-16
3.2 Modo de Uso	17-22
Capítulo 5 Conclusiones y líneas futuras.....	23
Capítulo 6 Summary and Conclusions	24
Capítulo 7 Presupuesto	25
Capítulo 8 Apéndice 1: Código fuente del sistema de gráficas	26-29

Índice de figuras

Figura 2.3.1.1: Vista administrador prototipo.....	8
Figura 2.3.1.2: Vista monitor prototipo	8
Figura 2.3.1.3: Vista actividades prototipo	9
Figura 2.3.3.1: Diagrama de la base de datos	10
Figura 3.3.1: Login de la aplicación.....	12
Figura 3.3.1.1: Vista administrador	13
Figura 3.1.2.1: Vista coordinador	14
Figura 3.1.3.1: Vista monitor.....	15
Figura 3.1.4.1: Gráfica media calificaciones por año, Apéndice 1.....	16
Figura 3.1.4.2: Gráfica media calificaciones de un evento, Apéndice 1.....	16
Figura 3.2.1: Agregando un coordinador.....	17
Figura 3.2.2: Agregando un monitor.....	18
Figura 3.2.3: Agregando un evento.....	19
Figura 3.2.4: Agregando al participante “Equipo_Rojo”	19
Figura 3.2.5: Agregando al participante “Equipo_Azul”	20
Figura 3.2.6: Agregando nueva evaluación.....	20
Figura 3.2.7: Vista evaluaciones asignadas al monitor	21
Figura 3.2.8: Actualizando calificación	21
Figura 3.2.9: Actividades evaluadas.....	22
Figura 3.2.10: Gráfica evento ejemplo	22
Figura 7.1.1: Código fuente 1	26
Figura 7.1.2: Código fuente 2	27
Figura 7.1.3: Código fuente 3	27
Figura 7.1.4: Código fuente 4	28
Figura 7.1.5: Código fuente 5	28
Figura 7.1.6: Código fuente 6	29

Índice de tablas

Tabla 6.1: Presupuesto.....	25
-----------------------------	----

Capítulo 1

Introducción

1.1 Pensamiento Computacional

El Pensamiento Computacional se define como la capacidad de una persona para, a través de habilidades propias de la computación, hacer frente a problemas de diversa índole. Entre estas habilidades se incluyen la descomposición de un problema, la creación de algoritmos y el procesamiento de datos, así como la comprensión de nociones de pensamiento divergente, heurística, y el pensamiento abstracto.

Teniendo esto en cuenta, esta metodología que permite estudiar y trabajar de manera organizada a través de procesos y del aprendizaje de sus errores, es una aptitud indispensable en un siglo XXI cada vez más tecnológico.

Para fomentar el desarrollo del Pensamiento Computacional existen dos metodologías educativas complementarias denominadas actividades ‘Desenchufadas’ (*Unplugged*) y actividades ‘Enchufadas’ (*Plugged*). Las actividades enchufadas son el método más común del entrenamiento del Pensamiento Computacional. Son, en resumen, actividades orientadas a la programación o programación en sí misma realizada en un ordenador, tableta o similar. Las actividades desenchufadas por su parte hacen uso de diversos objetos como cartas, cuerdas o de juegos de lógica para entender los diferentes conceptos relacionados con el desarrollo de software mencionados anteriormente, todo ello sin necesidad de utilizar un ordenador.

Tras un análisis de ambas vertientes metodológicas se ha llegado a la siguiente conclusión, aunque es muy sencillo evaluar las actividades enchufadas gracias a las facilidades que otorgan las diversas aplicaciones orientadas a ello, las actividades desenchufadas no cuentan con un sistema de evaluación que permita organizar las puntuaciones de dichas actividades. Por ello, el objetivo de este trabajo es **diseñar e implementar un sistema software para almacenar y procesar los resultados de actividades desenchufadas.**

1.2 Antecedentes

El concepto de lo que se denomina Pensamiento Computacional surge en la década de 1950, pero gran parte de sus ideas son mucho más antiguas. Sin embargo, quién puso esta idea en la vanguardia de la Informática fue J. Wing [1], en el año 2006, como resultado de la publicación de un ensayo en la revista ‘*ACM Communications*’, en el cual destacaba que el Pensamiento Computacional era una habilidad fundamental para todos, no solo para los informáticos, abogando por integrar ideas computacionales en otras materias de los estudios primarios.

La utilidad del pensamiento computacional en otras ramas de estudio es algo demostrado y que cada vez está cogiendo más fuerza. Podemos observar un ejemplo en este artículo [2], en el cual utilizan este concepto para que estudiantes de biología entiendan mejor el proceso de la selección natural. Sin embargo, el problema principal de integrar este concepto en la educación es probablemente, la formación del profesorado, ya que como se señala en este artículo de la Universidad de Castellón [3], “*Es importante que los docentes recorran el mismo camino que sus futuros alumno (Lee y Resnick)*”. Y para ello es necesario que una generación de docentes sea formada en pensamiento computacional.

Asimismo, como concluye M. Zapata-Ros [4], hay mucho trabajo por delante en el diseño de actividades y en enlazar estas con las capacidades que desarrollan. Por este motivo, queremos aportar nuestro granito de arena facilitando la evaluación de estas con la aplicación propuesta.

1.3 Objetivos

Los objetivos de este trabajo de fin de grado se pueden resumir en los siguientes puntos:

- Realizar una documentación del pensamiento computacional enchufado y desenchufado comparando ambas vertientes.
- Hacer una búsqueda de aplicaciones orientadas a evaluar actividades desenchufadas.
- Desarrollar un sistema para almacenar y procesar actividades de pensamiento computacional desenchufado. Un sistema mediante el cual los evaluadores puedan seleccionar entre las actividades almacenadas, crear su propia combinación de estas y seguidamente evaluar a sus alumnos.
- Como objetivo adicional se pretende que el sistema permita hacer comparativas entre alumnos y grupos de alumnos, es decir, que tras realizar un mismo conjunto de actividades, el usuario tenga la posibilidad de cotejar los resultados, a partir de un sistema de gráficas o algo similar.

Capítulo 2

Diseño del Sistema Software

En este capítulo hablaremos del diseño del sistema. Lo dividiremos en tres apartados: “Arquitectura del software” “Herramientas de Desarrollo de Software” y “Prototipo”.

En el primer apartado hablaremos de la arquitectura del software[5] haciendo referencia al modelo vista controlador.

En el segundo apartado hablaremos de las herramientas utilizadas para el desarrollo del sistema, desde el editor de código fuente hasta los lenguajes de programación utilizados, sin olvidarnos de otras herramientas importantes como la base de datos utilizada y el servidor web.

En el tercer apartado os mostraremos el prototipo creado antes de comenzar con el desarrollo de la aplicación.

2.1 Arquitectura del software

La arquitectura del software será una arquitectura en 3 capas siguiendo el modelo vista-controlador. Las tres capas en cuestión son:

El Modelo

En esta capa se gestiona la lógica, la funcionalidad del sistema, aspectos como que evaluaciones puede hacer un usuario y a que servicios tiene acceso. Lleva un registro de las vistas y controladores del sistema.

EL Controlador

Esta capa recibe los eventos de entrada y contiene las reglas de gestión de eventos. Estas acciones pueden suponer peticiones al modelo o a las vistas. Una petición a las vistas puede ser por ejemplo la llamada a la función `actualizarEvaluacion()`, disponible para evaluar las actividades.

Las Vistas

Son responsables de recibir datos del modelo y mostrarlos al usuario. En nuestro caso, todas las vistas se engloban en tres grupos principales. Las del administrador, las del coordinador y las del monitor.

2.2 Herramientas de Desarrollo de Software

A continuación, describiremos las herramientas utilizadas para la realización de este proyecto.

2.2.1 HTML

Se trata de un lenguaje marcado que se utiliza para el desarrollo de páginas web. Su nombre se corresponde con las siglas *HyperText Markup Lenguaje*, es decir, lenguaje de marcas de hipertexto. La estructura de los documentos desarrollados con este lenguaje se define mediante etiquetas. A parte de esta estructuración lógica, HTML nos ofrece una gran adaptabilidad y una facilidad de interpretación muy alta tanto para humanos como para máquinas.

2.2.2 CSS

Es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje marcado. Es usado para crear el diseño visual de documentos web e interfaces de usuario escritas en HTML o XHTML. Se puede aplicar a cualquier documento XML. Podemos considerarlo como el segundo lenguaje más esencial para crear páginas web.

CSS consiste en una serie de elementos mediante los cuales se declaran los estilos, los más importantes son:

- Selectores: mediante los cuales podemos especificar a que elemento de la página nos referimos.
- Atributos: para elegir qué cosas queremos estilizar en los selectores indicados.
- Valores: que indican que estilo se debe aplicar a cada atributo sobre cada selector. Los valores se especifican en unidades CSS que sirven para cuantificar los valores (píxeles, puntos, porcentajes...).

2.2.3 JavaScript

Se trata de un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, dinámico, imperativo y débilmente tipado. Este lenguaje se diseñó con una sintaxis similar a C, aunque adopta cierta sintaxis de Java.

Todos los navegadores modernos interpretan el código JS integrado en las páginas web. Tradicionalmente este lenguaje se utilizaba para realizar operaciones en el marco del cliente, pero actualmente se utiliza también para enviar y recibir información del servidor apoyándose en otras tecnologías.

Sus ventajas principales son que es un lenguaje sencillo, rápido y muy versátil, además es multiplataforma, permite crear interfaces dinámicas y al estar tan extendido tiene actualizaciones frecuentes.

Para la realización de este sistema, además del desarrollo habitual con JavaScript, hemos utilizado una librería externa:

- Plotly[6]: Ofrece varias librerías en diferentes lenguajes para crear gráficos de todo tipo de una manera mucho más sencilla.

2.2.4 Bootstrap

Bootstrap [7] es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con todo tipo de elementos para añadir a tu web basados en HTML y CSS, así como en extensiones de JavaScript adicionales. Es un framework que solo se encarga de front-end. Es compatible con la mayoría de los navegadores web.

2.2.5 JQuery

JQuery [8] es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

Además del uso de AJAX, JQuery me ha servido para poder utilizar SELECT2, que es un plugin de JQuery que nos permite tener un selector de elementos con un cuadro de búsqueda poniendo así más sencillo encontrar lo que desea buscar el usuario. Es

muy útil sobre todo en selectores donde se ofrecen infinidad de opciones, como por ejemplo, en nuestro caso, al navegar entre las evaluaciones.

2.2.6 PHP

Es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web. Fue creado inicialmente por el programador danés Rasmus Lerdorf en 1994. En la actualidad la implementación de referencia de PHP es producida por The PHP Group. Sus siglas significan *Hypertext Preprocessor*.

Lo que distingue a PHP de lenguajes del lado de cliente como JavaScript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente.

PHP tiene la ventaja de ser muy simple para principiantes y a su vez tener características avanzadas para los más profesionales.

2.2.7 SQL

Es un lenguaje de dominio específico utilizado en programación, diseñado para administrar y recuperar información de sistemas de gestión de bases de datos relacionales. Una de sus características es el manejo de álgebra y de cálculo relacional para realizar consultas con el fin de recuperar o modificar información de la base de datos.

2.2.8 XAMPP

XAMPP [9] es un paquete de software libre que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y PERL.

- Apache: El servidor HTTP Apache es un servidor web HTTP de código abierto que implementa el protocolo HTTP/1.1 y la noción de sitio virtual según la normativa RFC 2616. Apache presenta entre otras características altamente configurables bases de datos de autenticación y negociado de contenido.

- Mysql: Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual y está considerada la base de datos de código abierto más popular del mundo.

2.2.9 Visual Studio Code

Visual Studio Code [10] es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

2.2.10 Balsamiq

Balsamiq [11] es una herramienta que permite diseñar de forma rápida y sencilla maquetas de interfaz para webs y aplicaciones móviles. Ha sido usada en el prototipo de la aplicación.

2.3 Prototipo

La aplicación contará con sistema de evaluación mediante el cual un coordinador podrá crear eventos y programar actividades de pensamiento computacional desenchufado en dichos eventos para que posteriormente el monitor asignado a las mismas las califique.

Vamos a resumir el prototipo en cinco apartados fundamentales:

- El diseño y distribución del sistema de evaluación
- Los tipos de usuarios
- Las clases u objetos que almacenar en la base de datos
- Las operaciones realizables a partir de las calificaciones
- Validación y pruebas

2.3.1 Diseño y distribución del sistema de evaluación

La vista del **administrador** y la del **coordinador** tendrán una estructura similar a una tabla, su fila superior será el campo “actividades” y la columna izquierda tendrá el campo “participantes”. También contendrá una fila sobre el campo “actividades” que indicará el monitor de cada una de estas. Adicionalmente, dispondrá de un botón añadir que permitirá añadir actividades y otro similar para añadir participantes.

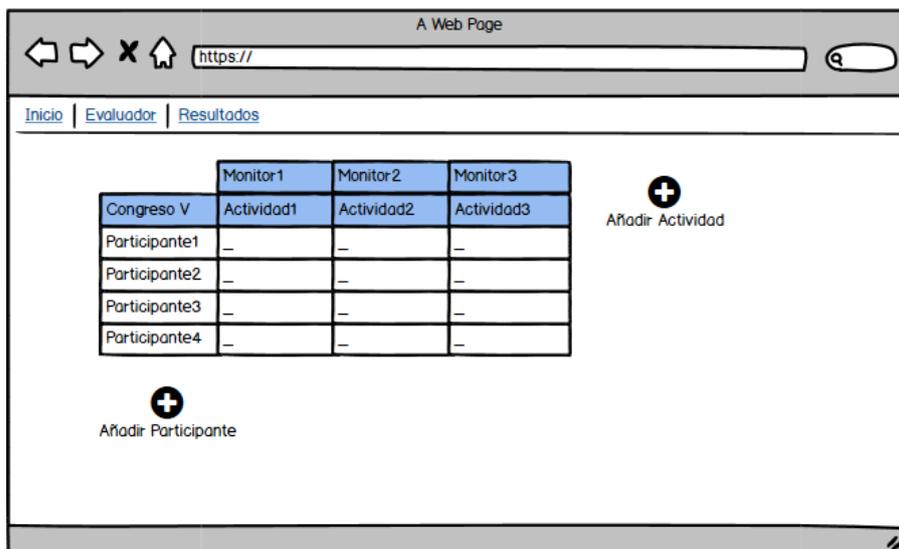


Figura 2.3.1.1: Vista administrador prototipo

La vista del **monitor** mantendrá la estructura anterior, pero el monitor no podrá añadir actividades ni participantes. Además, a dicho usuario solo se le mostrarán las actividades que él evalúe.

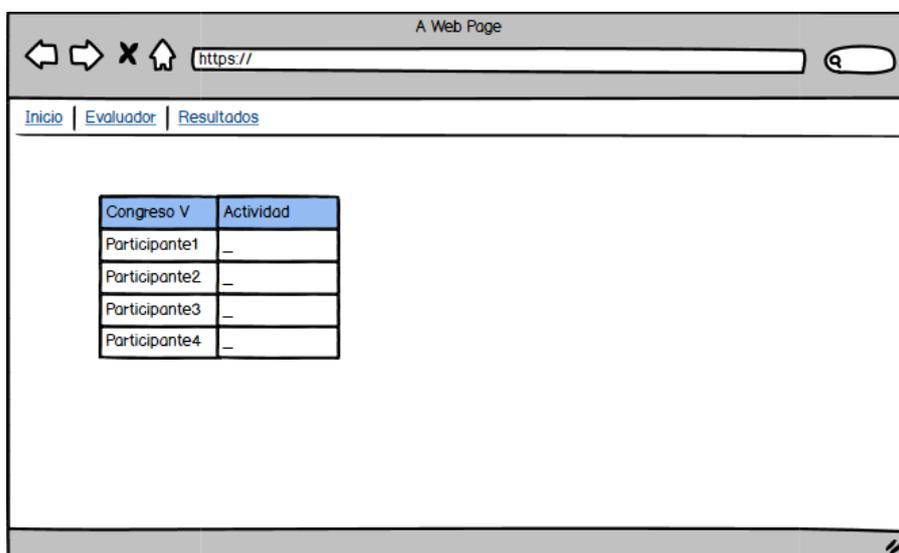
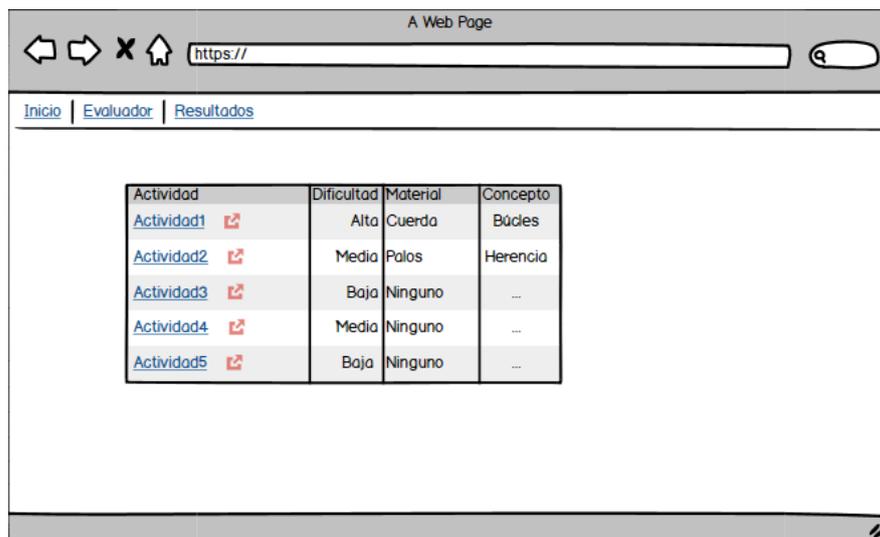


Figura 2.3.1.2: Vista monitor prototipo

Las diferentes **actividades** se mostrarán en un listado con diferentes parámetros, desde el nombre de la actividad, hasta una descripción de la misma, pasando por el material necesario para realizarla.



The screenshot shows a web browser window titled "A Web Page" with a search bar containing "https://". Below the browser window, there is a navigation menu with links for "Inicio", "Evaluador", and "Resultados". The main content area displays a table with the following data:

Actividad	Dificultad	Material	Concepto
Actividad1 	Alta	Cuerda	Báscles
Actividad2 	Media	Palos	Herencia
Actividad3 	Baja	Ninguno	...
Actividad4 	Media	Ninguno	...
Actividad5 	Baja	Ninguno	...

Figura 2.3.1.3: Vista actividades prototipo

2.3.2 Tipos de usuarios

En el sistema distinguiremos 3 tipos de usuarios:

- Monitores: usuarios que califican las actividades
- Coordinadores: usuario intermedio, dirigen los eventos que les han sido asignados, escogiendo las actividades que se llevarán a cabo y asignando a cada una un monitor.
- Administrador General: usuario de más alto nivel, es el encargado de generar los nuevos eventos e introducir las actividades en el sistema. Además, tiene acceso a todo el contenido del sistema.

2.3.3 Base de datos

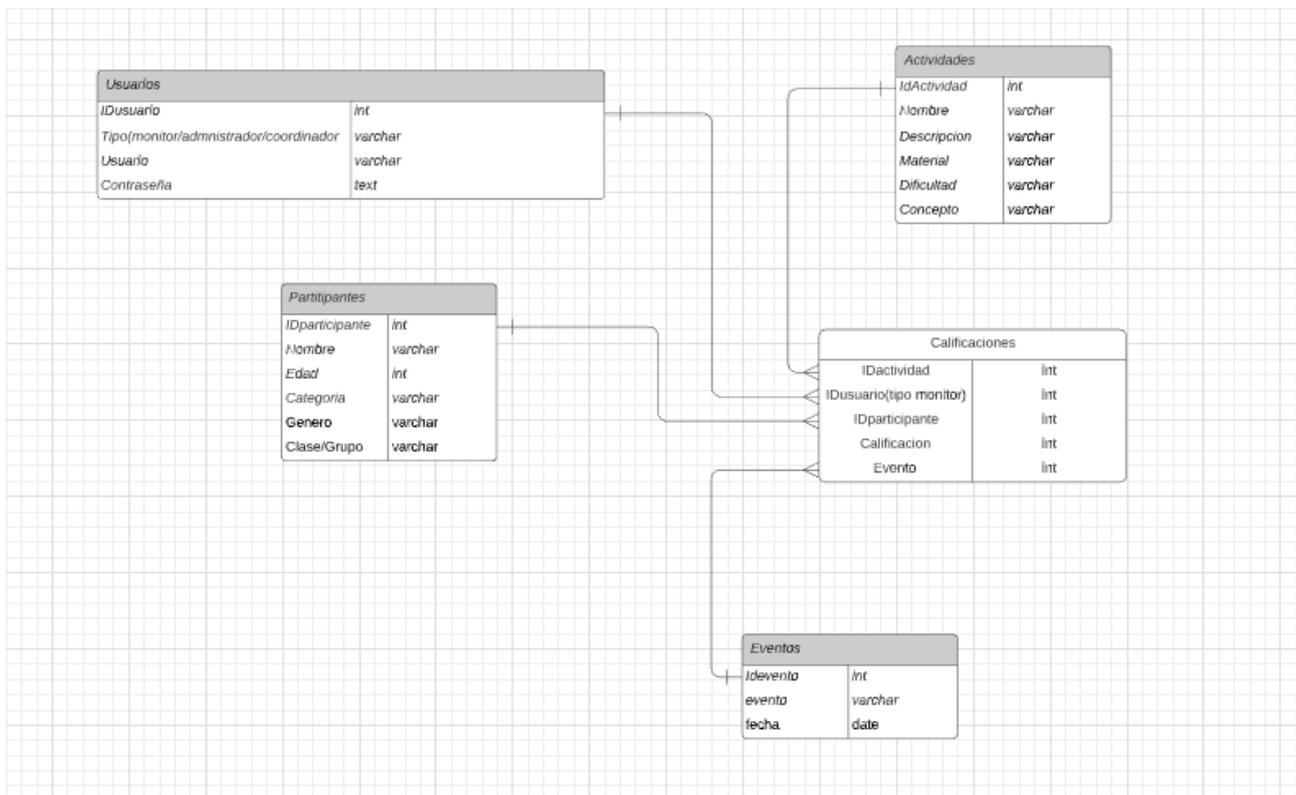
En la base de datos encontraremos las siguientes tablas:

- Actividades: tendrá los siguientes campos: *IDactividad*, *nombre*, *descripción*, *material necesario*, *nivel de dificultad* (por edades), y *concepto computacional*

que desarrolla (bucles, herencia...). Su clave primaria será el *IDactividad* así como su clave foránea.

- Participantes: tendrá los siguientes campos: *IDparticipante*, *nombre*, *edad*, *categoría* (individual/grupo), *género*, *clase* (aula etc). Su clave primaria será el *IDParticipante* así como su clave foránea.
- Usuarios: tendrá los siguientes campos: *IDusuario*, *tipo* (monitor, coordinador, administrador), *nombre*. Su clave primaria será el *IDusuario* así como su clave foránea.
- Eventos: tendrá los siguientes campos: *IDevento*, *NombreEvento*, *FechaEvento*. Su clave primaria será el *IDevento* así como su clave foránea.
- Calificaciones: será la tabla que relaciona a todas las demás, tendrá los siguientes campos: *IDparticipante*, *IDactividad*, *IDmonitor*, *IDevento*, *calificación*. Su clave primaria estará compuesta por los campos *IDparticipante*, *IDactividad*, *IDmonitor* e *IDevento*.

Diagrama de la base de datos:



[12] Figura 2.3.3.1: Diagrama de la Base de Datos

2.3.4 Operaciones

El Sistema hará las siguientes operaciones con las calificaciones aportadas:

- Media de las calificaciones de todas las actividades.
- Ordenar los participantes según la calificación de una actividad.
- Comparar media de actividades en un evento.
- Comparar media de actividades por año.

2.3.5 Validación y Pruebas.

La validación y las pruebas las dividiremos en dos apartados:

- 1) Despliegue: El despliegue de la aplicación se hará en local. Comprobando la correcta conexión entre el sistema y la base de datos.
- 2) Pruebas de Usuario: Se procederá a crear diversos usuarios de cada tipo (Administrador, Coordinador, Monitor) y se comprobará que dichos usuarios puedan acceder al sistema. Además, se revisará que dichos usuarios solo puedan acceder a las páginas en las que tienen permisos para evitar problemas de seguridad.
- 3) Pruebas de Actividades: Se comprobará que todos los usuarios previamente creados puedan realizar todas las funcionalidades que están disponibles en el sistema. Para ello, se realizarán varios test en los cuales se simulará el transcurso de un evento, donde se crearán nuevos participantes y actividades, se añadirán y asignarán nuevas evaluaciones y además estas serán evaluadas.

Capítulo 3

Modo de Uso

En este capítulo procedemos a explicar cómo funciona el sistema. En primer lugar, hablando de la estructura del sistema para posteriormente demostrar como funciona dicho sistema mediante un ejemplo práctico.

3.1 Estructura del Sistema

La estructura del sistema web está dividida en tres vistas, cada una acorde a un tipo de usuario (Administrador, Coordinador y Monitor). En cada una de estas vistas el usuario en cuestión tiene acceso solo a las funcionalidades que tiene permitidas.

En primer lugar, para poder acceder a la vista de tu usuario, debes pasar por el inicio de sesión de la página, que es lo primero que ves al acceder al sistema. En esta página el usuario debe introducir su usuario y contraseña, y tras autenticarse, entrará a la web. Cabe señalar que la sesión se mantendrá abierta hasta que dicho usuario pulse el botón “cerrar sesión” ubicado en la barra de navegación.

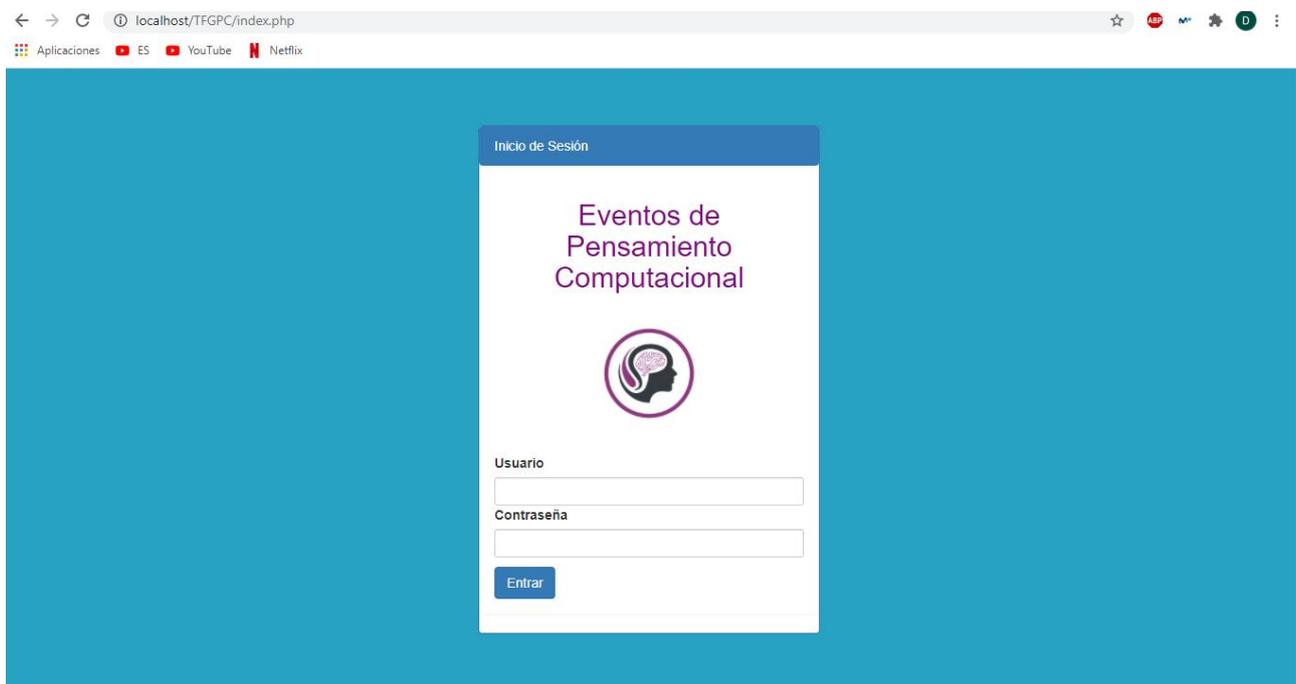


Figura 3.3.1: Login de la aplicación

3.1.1 Administrador

Es el usuario con nivel más alto en el sistema, puede hacerlo todo, desde funcionalidades exclusivas como crear y eliminar usuarios, hasta otras que pueden realizar otros usuarios como crear y editar actividades, añadir o eliminar participantes hasta incluso modificar evaluaciones si así lo ve conveniente.

The screenshot shows a web browser at localhost/TFGPC/Actividades.php. The navigation menu includes Inicio, Actividades, Participantes, Usuarios, Eventos, Evaluaciones, Estadística, and Cerrar Sesión. A search bar labeled 'Buscador' is present. The main content area is titled 'Actividades de Pensamiento Computacional' and features a table of activities with columns for Nombre, Descripción, Material, Dificultad, Concepto que Desarrolla, Editar, and Eliminar. A 'Agregar nuevo +' button is located above the table.

Nombre	Descripción	Material	Dificultad	Concepto que Desarrolla	Editar	Eliminar
Descomposición	Los participantes tienen que analizar diversos problemas (por ejemplo, plantar un árbol) identificando todos los pasos necesarios para resolverlo.	Folios	4-8 años	Descomposición y Algoritmia		
Mapa de Mónica	Un mapa con muchos personajes es mostrado a los estudiantes y tienen que encontrar la ruta más corta entre ellos usando solo flechas arriba, abajo, izquierda y derecha (→, ←, ↑ y ↓). Para mayor dificultad, deberán usar multiplicadores (es decir, → → → → → = 5x →) para escribir las soluciones.	Folios	4-8 años	Reconocimiento de patrones y Algoritmia		
Tetris	Se presentan algunos dibujos de piezas de Tetris a uno de los estudiantes que da instrucciones a su compañero. El estudiante que obtuvo la parte superior del papel tiene que ocultar la imágenes del compañero para que solo sea posible escuchar las instrucciones sin buscar las respuestas. Las instrucciones se limitan a "inicio", "arriba", "abajo", "izquierda", "derecha" y "detener". No se pueden usar otras palabras para describir cómo se dibuja la figura.	Folios	4-8 años	Reconocimiento de patrones y Algoritmia		
Autómatas de Mónica	Esta actividad es un remake más simple del juego de mesa Code Master desarrollado por la empresa ThinkFun. En esta actividad se supone que el alumno	Colores y folios	4-8 años	Descomposición, Algoritmia v		

Figura 3.3.1.1: Vista administrador

3.1.2 Coordinador

Es el usuario de nivel medio, tiene la capacidad de añadir nuevos participantes, crear nuevos eventos y acceder a los datos estadísticos. Pero su funcionalidad principal es la de asignar evaluaciones a los monitores, es decir, programar una actividad para un conjunto de participantes en un evento previamente creado y asignarle a esta un monitor.

Actividad	Monitor	Participante	Evento	Calificación	Editar	Eliminar
Descomposición	mo001	Javier Gonzales	Congreso_IV	9		
Mapa de Mónica	mo001	Irina Suarez	Congreso_IV	No evaluada		
Mapa de Mónica	mo001	María Medina	Congreso_IV	No evaluada		
Mapa de Mónica	mo001	Javier Gonzales	Congreso_IV	No evaluada		
Mapa de Mónica	mo001	Joshua Sanchez	Congreso_IV	No evaluada		
Autómatas de Mónica	mo001	María Medina	Congreso_V	No evaluada		

Figura 3.1.2.1: Vista Coordinador

3.1.3 Monitor

Es el usuario de nivel más bajo, su única funcionalidad es la de evaluar las actividades que le han sido asignadas.

© Daniel Alberto Melián León

Figura 3.1.3.1: Vista Monitor

3.1.4 Estadística

A parte de las funcionalidades previamente mencionadas, las cuales desarrollaremos con un ejemplo en el siguiente punto. El sistema cuenta con una pestaña accesible a todos los usuarios. Esta pestaña es la denominada “estadística” y cuenta con unas gráficas en la que se muestran la media de calificaciones de cada año, así como un buscador que permite acceder a las medias concretas de los diferentes eventos.

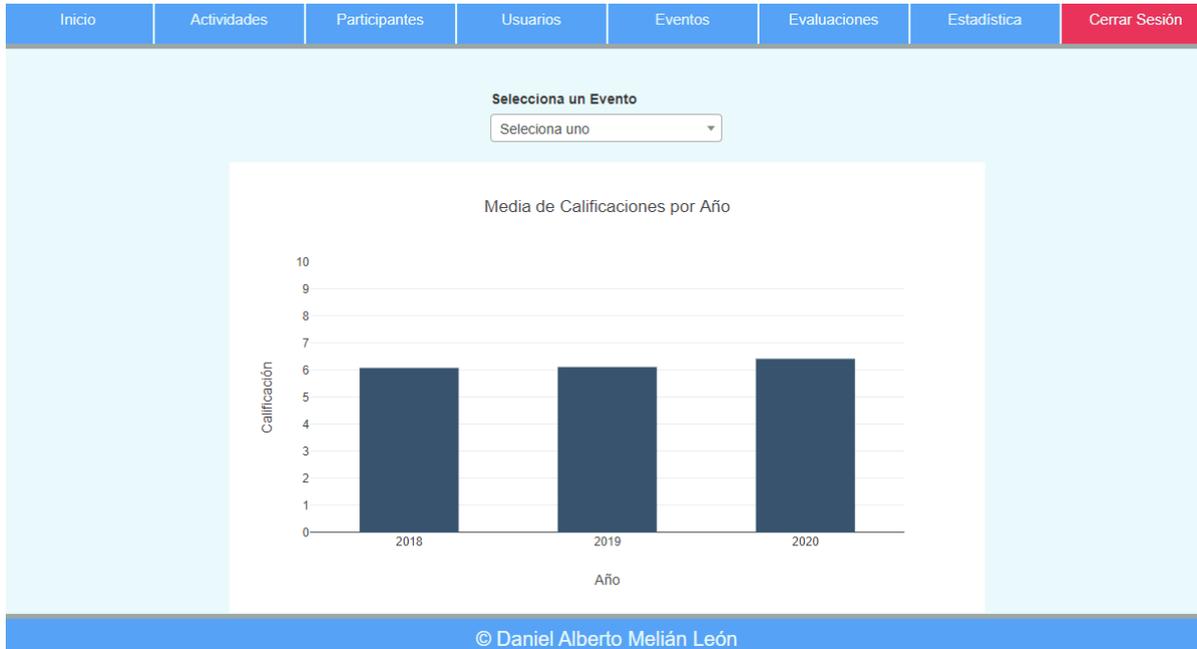


Figura 3.1.4.1: Gráfica media calificaciones por año, Apéndice 1



Figura 3.1.4.2: Gráfica media calificaciones de un evento, Apéndice 1

3.2 Ejemplo Práctico

En este punto se explicará mediante un ejemplo, el funcionamiento del sistema. En el ejemplo en cuestión se procederá a crear el evento para las Olimpiadas de Pensamiento Computacional 2020, se añadirán los participantes, se les asignará una actividad con un monitor determinado, y finalmente se evaluarán.

En **primer lugar**, procedemos desde una cuenta de administrador a crear un coordinador para dicho evento, así como un monitor para evaluar la actividad (podrían usarse los usuarios ya creados, pero procedemos como si empezáramos de cero).

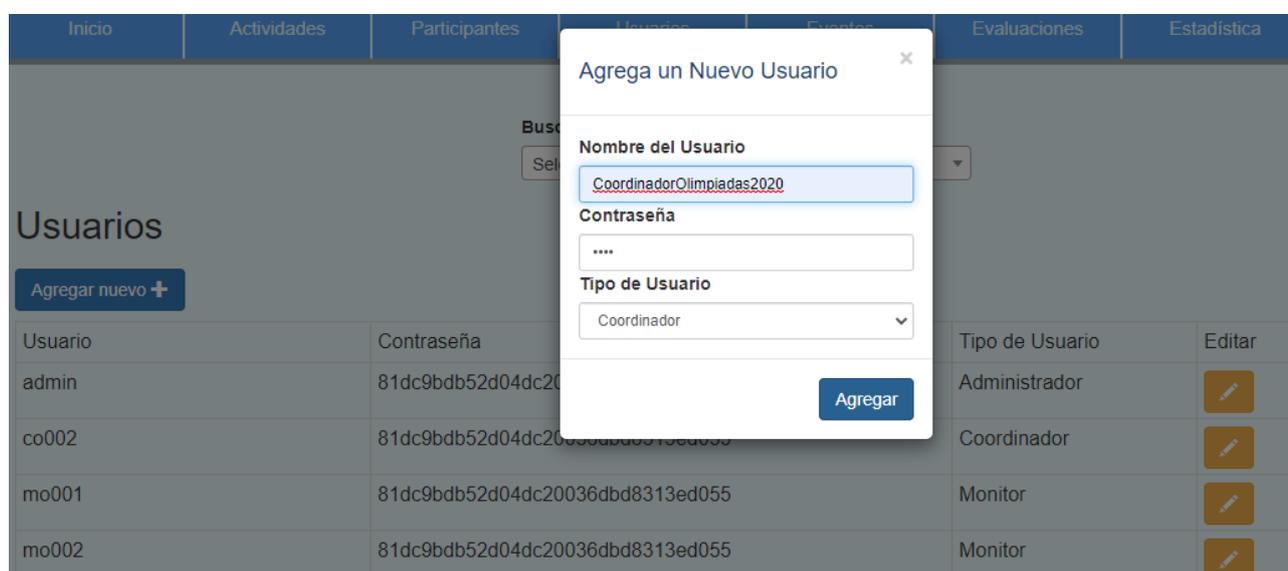


Figura 3.2.1: Agregando un coordinador

Para ello, iniciamos sesión como administrador, seleccionamos la pestaña “usuarios” para movernos a dicha página, y una vez ahí debemos pulsar el botón “Agregar Nuevo” situado arriba a la izquierda para luego introducir el nombre del usuario, la contraseña y seleccionar el tipo (en este caso, coordinador). Para finalizar este paso pulsamos el botón “Agregar” y con esto ya tenemos al coordinador agregado.

El **segundo paso** sería repetir el paso anterior para añadir al monitor. La única diferencia es que en el tipo de usuario debemos seleccionar la opción “Monitor” (Este paso se debe repetir tantas veces como monitores se quiera crear)

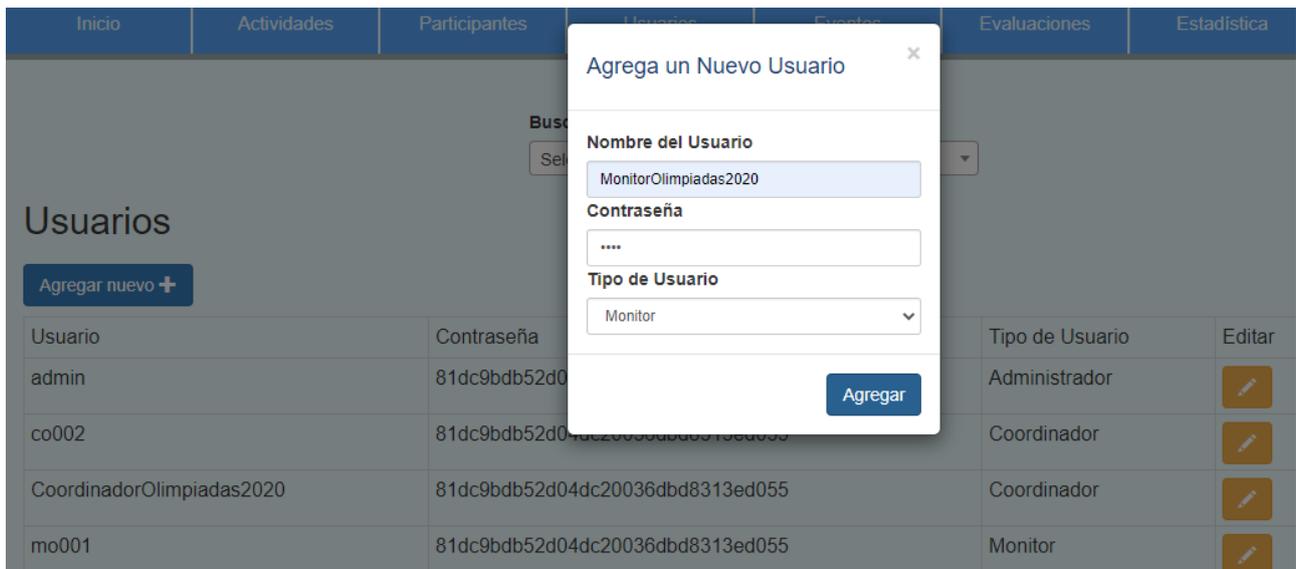


Figura 3.2.2: Agregando un monitor

En **tercer lugar**, procedemos a acceder a la cuenta del coordinador, desde la cual crearemos el evento. Para ello nos movemos a la pestaña eventos, y seguidamente pulsamos en “Agregar nuevo” y rellenamos los datos como en casos anteriores (en este caso solo es necesario introducir el nombre del evento y la fecha de este).

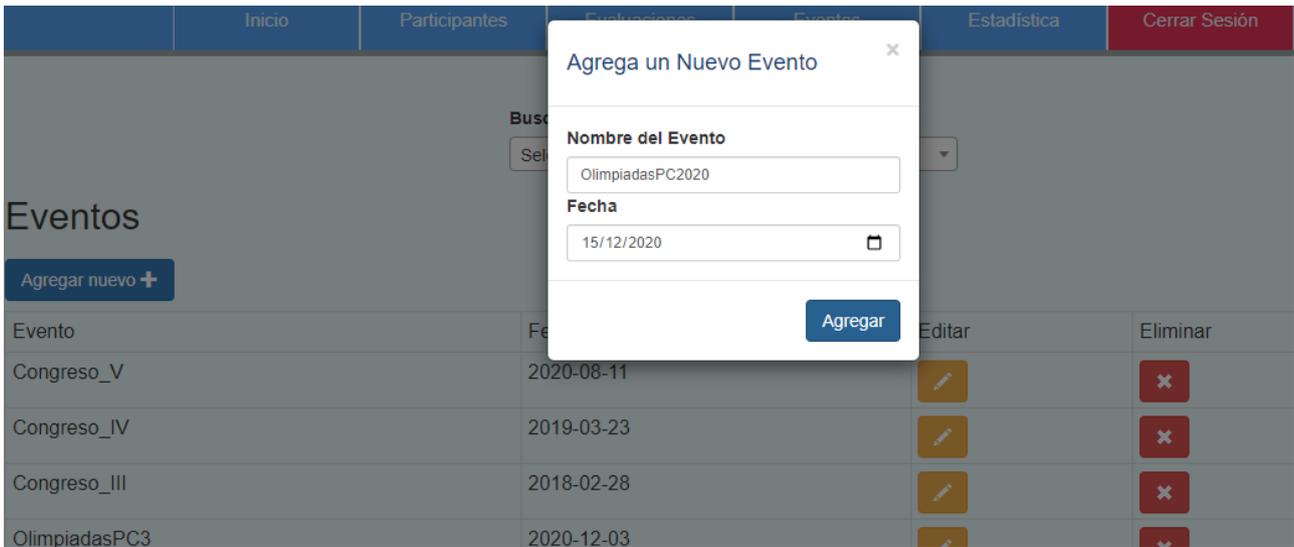


Figura 3.2.3: Agregando un evento

El **Cuarto Paso** sería añadir a los participantes del evento, para ello nos movemos a la pestaña “participantes”, seleccionamos agregar nuevo y rellenamos los campos (en este supuesto creamos dos grupos denominados “EquipoRojo” y “EquipoAzul”, suponemos que lo componen niños de 8 años, además, como en el campo “categoría” indicamos que son un grupo, no introducimos género. Para acabar, en el campo Grupo indicamos que pertenecen al Colegio1 y procedemos a pulsar el botón “agregar”.



Figura 3.2.4: Agregando al participante “Equipo_Rojo”

Repetimos el cuarto paso y añadimos también al Equipo Azul.

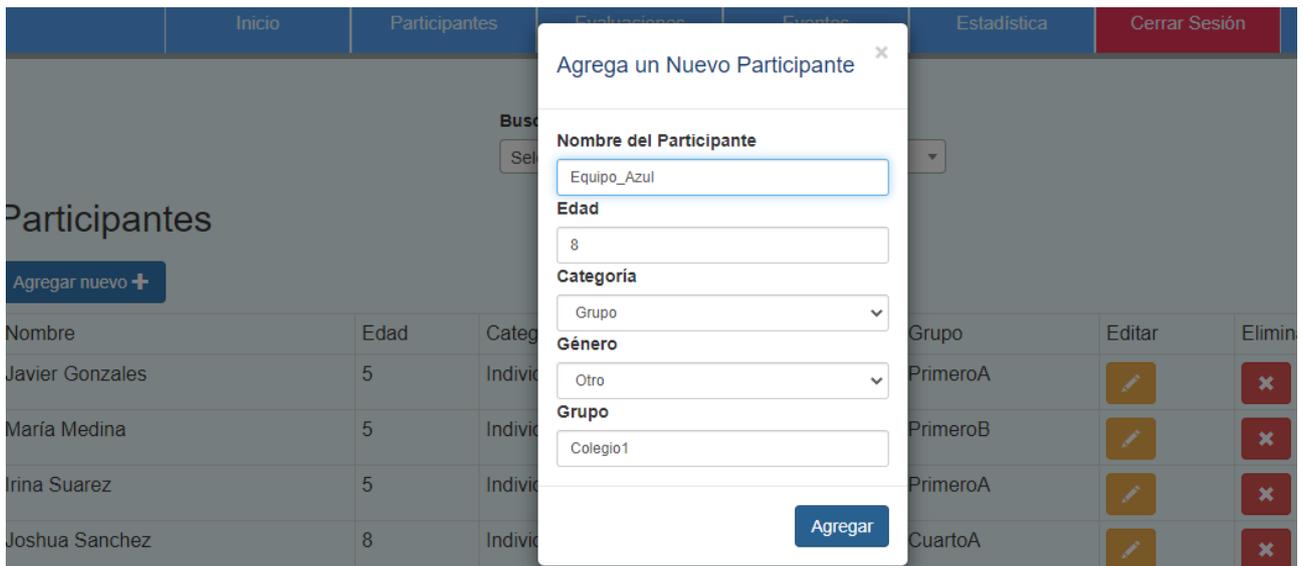


Figura 3.2.5: Agregando al participante “Equipo_Azul”

Ya con los dos equipos agregados, procedemos al **quinto paso**, en el cual el coordinador debe asignar una actividad a ambos equipos y asignarle un monitor a la misma. Para ello, el coordinador debe moverse a la pestaña evaluaciones y allí pulsar en “agregar una evaluación grupal” (Podrían añadirse de uno en uno si fuera necesario pulsando en “agregar una evaluación individual”) posteriormente debe seleccionar la actividad, el monitor, el grupo a evaluar y el evento al que pertenece la actividad.

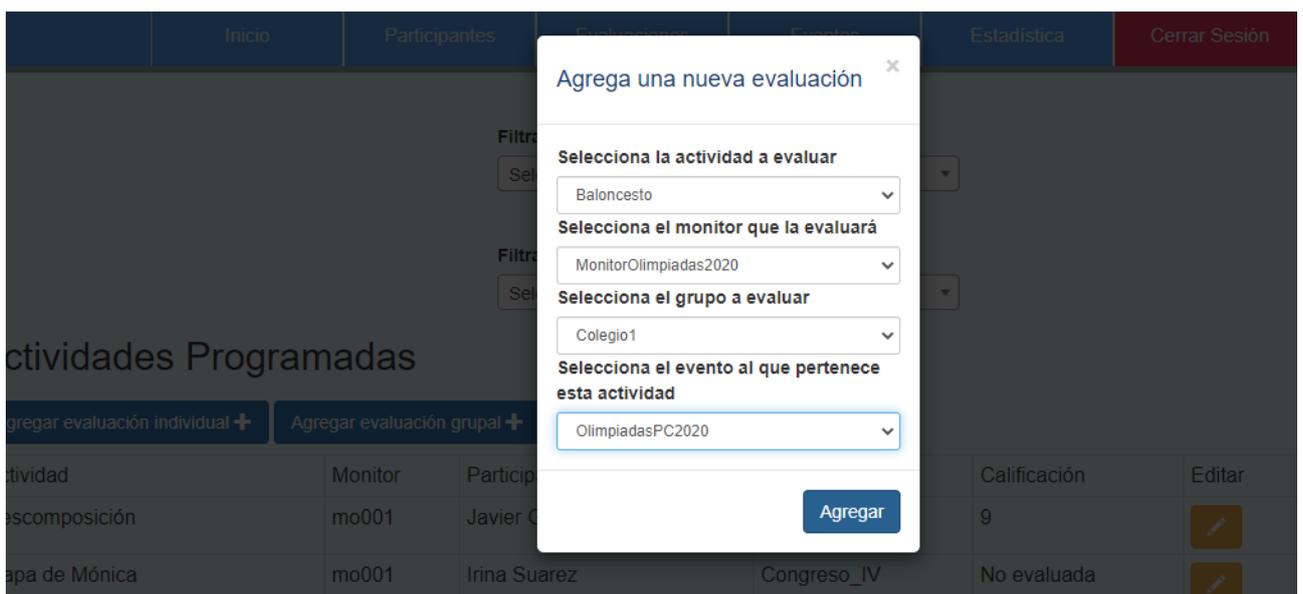


Figura 3.2.6: Agregando nueva evaluación

Al pulsar en agregar, automáticamente se genera la actividad “Baloncesto”, en el evento “OlimpiadasPC2020”, evaluada por el “MonitorOlimpiadas2020”, y se incluye en ella a todos los participantes del “Colegio1”

Y para finalizar nos queda el **sexto** y último paso, es decir, que el monitor acceda a su cuenta, y desde ahí se mueva a la pestaña evaluaciones, y evalúe.

Actividad	Monitor	Participante	Evento	Calificación	Editar	Eliminar
Baloncesto	MonitorOlimpiadas2020	Equipo_Rojo	OlimpiadasPC2020	No evaluada		
Baloncesto	MonitorOlimpiadas2020	Equipo_Azul	OlimpiadasPC2020	No evaluada		

Figura 3.2.7: vista evaluaciones asignadas al monitor

Para evaluar la actividad debe pulsar el botón editar que se encuentra en cada fila de la tabla y agregar la calificación (0-10).

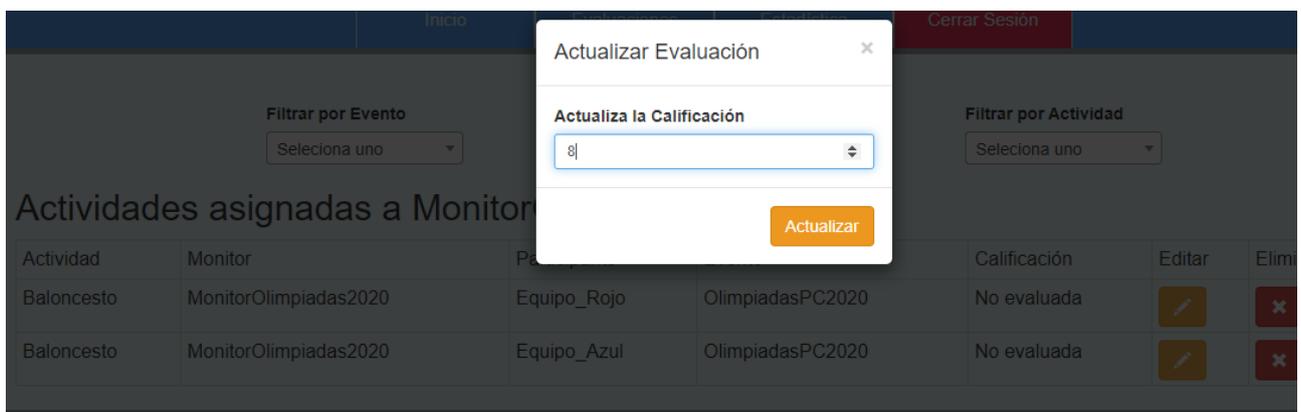


Figura 3.2.8: actualizando calificación

Después de evaluar el resultado sería el siguiente:

Actividad	Monitor	Participante	Evento	Calificación	Editar	Eliminar
Baloncesto	MonitorOlimpiadas2020	Equipo_Azul	OlimpiadasPC2020	5		
Baloncesto	MonitorOlimpiadas2020	Equipo_Rojo	OlimpiadasPC2020	8		

Figura 3.2.9: Actividades evaluadas

Adicionalmente, he añadido otra actividad al evento siguiendo los mismos pasos, y la he evaluado para poder observar en la pestaña estadística la gráfica de la media de evaluaciones en el evento.



Figura 3.2.10: Gráfica evento ejemplo

Capítulo 4

Conclusiones y líneas futuras

El pensamiento computacional es algo que cada vez está cogiendo más fuerza en nuestra sociedad. Hay muchas iniciativas en diversos países como “PlayMaker”[4] en Singapur, “Cs unplugged”[4] en Nueva Zelanda y otras muchas en España respaldadas por el INTEF[13]. Este cambio, nos muestra, que por fin se está haciendo eco del valor innegable que tiene el pensamiento computacional en la educación de los más jóvenes.

Por todo ello, como ya reflejé al inicio, pienso en este proyecto como un grano de arena más en el campo del pensamiento computacional. Un acercamiento a este campo que espero que en el futuro haga más fácil la labor de todos los profesores que deseen llevar a cabo evaluaciones dentro del ámbito del pensamiento computacional desenchufado.

Mirando hacia el futuro, el siguiente paso sería conseguir que este sistema sea utilizado en futuros eventos de pensamiento computacional. Sin embargo, es innegable que esta muy lejos de ser un sistema perfecto, y sería ideal optimizarlo para adaptarlo a las necesidades reales de cada evento.

Por otro lado, recopilar una gran cantidad de datos a lo largo de los años utilizando este sistema, nos daría una visión gráfica y real del desarrollo de esta capacidad en los más jóvenes.

Para terminar, hay que recordar que este trabajo sigue la filosofía del código abierto, por lo que se anima a cualquiera que lo desee a utilizarlo, modificarlo y mejorarlo como desee siguiendo esta misma filosofía.

Capítulo 5

Summary and Conclusions

The computational thinking is something that is gaining more visibility in our society. There are many initiatives in various countries such as “PlayMaker”[4] in Singapore, “Cs unplugged”[4] in New Zealand and many others, for example, in Spain, supported by INTEF[13]. This change, show us, that it is finally echoing the undeniable value that computational thinking has in the education of the young people.

For all of this, as I already showed at the beginning of this document, the objective of this project has been from the outset to contribute to the great development that the Computational Thinking in having in these years. An approach to this field that I hope will make easier the work of all teachers who wish to carry out evaluations within the field of unplugged computational thinking in the future.

On the first hand, the next step would be to get this system used in future computational thinking events. However, it’s undeniable that it’s far from being a perfect system, and it would be ideal to optimize it to adapt it to the real needs of each event.

On the other hand, collecting a large amount of data over the years using this system would give us a graphic and real vision of the development of capacity in the young people.

Finally, it must be remembered that this project follows the philosophy of open source, so anyone who wishes to use it in encouraged to use, modify and improve it as they wish following this same philosophy.

Capítulo 6

Presupuesto

Tareas	Horas	Presupuesto
Revisión bibliográfica	20h	5€/h
Estudio de antecedentes	30h	5€/h
Desarrollo de la aplicación	120h	20€/h
Pruebas de la aplicación	30h	10€/h
TOTAL	200h	2950€

Tabla 6.1: Presupuesto

Como se puede observar en la tabla el presupuesto total asciende a 2950€ por 200h de trabajo.

Capítulo 7

Apéndice 1

7.1 Código fuente del sistema de gráficas.

```
1  <?php
2
3  session_start();
4  require_once "../php/conexion.php";
5  $conexion=conexion();
6  $valoresX=array();
7  $valoresY=array();
8  $x=0;
9  $y=0;
10
11
12
13  if($_SESSION['consulta8'] > 0){
14
15      $idcom=$_SESSION['consulta8'];
16
17      $sql1="SELECT idact, AVG(calificacion)
18      FROM t_evaluacion WHERE ideve='$idcom' && calificacion!=-1 group by idact";
19      $result1=mysqli_query($conexion,$sql1);
20      while($ver1=mysqli_fetch_row($result1)){
21
22          $sql2="SELECT nombre
23          FROM t_actividad where id='$ver1[0]'";
24          $result2=mysqli_query($conexion,$sql2);
25          while($ver2=mysqli_fetch_row($result2)){
26
27              $valoresX[$x]=$ver2[0];
28              $valoresY[$y]=$ver1[1];
29              $x++;
30              $y++;
31          }
32      }
33
34  ?>
```

Figura 7.1.1: Código fuente 1

```
35 <script>
36
37     var layout = {
38         title: 'Media de calificaciones por actividad en el evento',
39
40         font:{
41             family: 'Raleway, sans-serif'
42         },
43         showlegend: false,
44         xaxis: {
45             dtick: 1,
46             title: 'Actividad',
47
48         },
49         yaxis: {
50             dtick: 1,
51             gridwidth: 1,
52             title: 'Calificación',
53             range: [0, 10],
54         },
55         bargap :0.02
56     };
57
58 </script>
59 <?php
60
61
62
```

Figura 7.1.2: Código fuente 2

```
64 }else{
65
66
67     $fechainicio=2018;
68     $añoactual=date("Y");
69     while($fechainicio < ($añoactual + 1)){
70         $sql="SELECT AVG(calificacion)
71             from t_evaluacion where calificacion!='-1' && ideve IN (SELECT idevento
72                                     from t_evento where YEAR(fecha)='$fechainicio')";
73         $result=mysqli_query($conexion,$sql);
74         while($ver=mysqli_fetch_row($result)){
75
76             $valoresX[$x]=$fechainicio;
77             $valoresY[$y]=$ver[0];
78
79         }
80
81         $x++;
82         $y++;
83         $fechainicio++;
84     }
85
86 }>
```

Figura 7.1.3: Código fuente 3

```

87     <script>
88         var layout = {
89             title: 'Media de Calificaciones por Año',
90
91             font:{
92                 family: 'Raleway, sans-serif'
93             },
94             showlegend: false,
95             xaxis: {
96                 dtick: 1,
97                 title: 'Año',
98             },
99         },
100         yaxis: {
101             dtick: 1,
102             gridwidth: 1,
103             title: 'Calificación',
104             range: [0, 10],
105         },
106         bargap :0.02
107     };
108     </script>
109     <?php
110
111
112     }
113

```

Figura 7.1.4: Código fuente 4

```

114
115
116     $datosX=json_encode($valoresX);
117     $datosY=json_encode($valoresY);
118
119
120     ?>
121
122     <div id="graficaBarras"></div>
123
124     <script type="text/javascript">
125
126         function crearCadenaBarras(json){
127             var parsed = JSON.parse(json);
128             var arr = [];
129             for (var x in parsed){
130                 arr.push(parsed[x]);
131             }
132             return arr;
133         }
134
135     </script>
136

```

Figura 7.1.5: Código fuente 5

```
137 <script>
138
139 datosX=crearCadenaBarras('<?php echo $datosX ?>');
140 datosY=crearCadenaBarras('<?php echo $datosY ?>');
141 var data = [
142     {
143         x: datosX,
144         y: datosY,
145         type: 'bar',
146         width:0.5,
147         marker: {color: 'rgb(55, 83, 109)'},
148     }
149 ];
150 ];
151
152 Plotly.newPlot('graficaBarras', data, layout);
153 </script>
```

Figura 7.1.6: Código fuente 6

Bibliografía

- [1] J. M. Wing, “Computational Thinking,” *Communications of the ACM*, vol. 49, no. 3, pp. 33-35, 2006.
- [2] A. Peel, T. D. Sadler, and P. Friedrichsen, “Learning natural selection through computational thinking: Unplugged design of algorithmic explanations,” *Journal of Research in Science Teaching*, vol. 56, no. 7, pp. 23-25, February 2019.
- [3] J. Adell Segura, F. Esteve Mon, M. Á. Llopis Nebot, M. G. Valdeolivas Novella, “EL PENSAMIENTO COMPUTACIONAL EN LA FORMACIÓN INICIAL DEL PROFESORADO DE INFANTIL Y PRIMARIA” *XXV Jornadas Universitarias de Tecnología Educativa*, 2017, Burgos, pp. 151-158.
- [4] M. Zapata-Ros, “Pensamiento computacional desenchufado,” *Education in the Knowledge Society*, vol. 20, 2019.
- [5] Cardacci, Darío G. (2015): “Arquitectura de software académica para la comprensión del desarrollo de software en capas”, Serie Documentos de Trabajo, No. 574, Universidad del Centro de Estudios Macroeconómicos de Argentina (UCEMA), Buenos Aires.
- [6] Plotly, “Plotly Open Source Graphing Libraries”, [Online]. Available: <https://plotly.com/graphing-libraries/>.
- [7] Twitter, “Bootstrap”, [Online]. Available: <https://getbootstrap.com/>.
- [8] The JQuery Foundation, “JQuery”, [Online]. Available: <https://jquery.com/>.
- [9] Apache Friends, “XAMPP”, [Online]. Available: <https://www.apachefriends.org/es/>.
- [10] Microsoft, “Visual Studio Code”, [Online]. Available: <https://code.visualstudio.com/>.
- [11] Balsamiq Studios LLC, “Balsamiq”, [Online]. Available: <https://balsamiq.com/>.
- [12] Lucid Software Inc., “Lucidchart”, [Online]. Available: <https://www.lucidchart.com/>.
- [13] INTEF. La escuela de pensamiento computacional. Intef.es. <https://intef.es/tecnologia-educativa/pensamiento-computacional>.