

# PROYECTO INTEGRADOR

Desarrollo De Aplicaciones Web

Pequeños Navegantes



**Autores:**

Rigoberto Araujo Coronado, Adrián Arcones Gomez y  
Daniel Clavel Vega

**Tutores:**

Sara Villanueva Rosa, Irene del Rincón Bello Y Raquel  
Cerdá Losa

## Introducción

En el presente proyecto de desarrollo de aplicaciones web tiene como finalidad principal la integración y aplicación de los conocimientos adquiridos en los módulos formativos de segundo año, los cuales son: **Desarrollo Web en Entorno Cliente, Desarrollo Web en Entorno Servidor, Diseño de Interfaces Web y Despliegue de Aplicaciones Web**. Además de estos, también utilizaremos conocimientos de módulos del primer año para su realización, los cuales son: **Lenguaje de Marcas, Bases de Datos y Entorno de Desarrollo**. Este enfoque permitirá abordar de manera integral el proceso de creación de una aplicación web funcional para un campamento infantil, desde el diseño de interfaces hasta su despliegue en un servidor, siguiendo los conocimientos adquiridos en los módulos formativos anteriores.

Utilizaremos para la realización de la aplicación en el entorno del Front-end lenguajes de marcas como HTML y CSS para aplicarle estilo a la web, para la validación de los formularios y aplicarle interactividad y dinamismo a la aplicación utilizaremos JavaScript. Para la parte del back-end, utilizaremos el lenguaje de PHP para la conexión con la base de datos SQL. Para la conexión asíncrona de PHP y JavaScript utilizaremos AJAX.

Para el desarrollo de nuestro proyecto de una manera más sencilla y llevar un orden de la creación de esta, hemos decidido adoptar la metodología ágil SCRUM mediante sprints. Este enfoque combina la flexibilidad y la adaptabilidad de los sprints con los valores fundamentales del espíritu SCRUM: colaboración, aprendizaje constante y orientación hacia un propósito común. A través de la metodología SCRUM por Sprints, buscamos garantizar un desarrollo ágil, eficiente y con un enfoque centrado en las personas, lo que nos permitirá adaptarnos a las necesidades del proyecto y generar resultados de alta calidad.

La finalidad de este proyecto es realizar una aplicación web funcional para un campamento de niños, en la cual los padres puedan obtener información de precios, actividades y eventos pasados, registrarse o reservar plaza. Los monitores podrán observar las actividades que impartirán ese día y la lista de niños que tienen en esa actividad para pasar lista o ver información de ellos, también los monitores podrán enviar un currículum para optar a conseguir trabajo el campamento.

Los administradores de la página podrán asignar a cada monitor actividades y niños a su cargo además de mirar los niños inscritos y si han pagado la reserva.

Para este proyecto que realizaremos siguiendo las pautas y gustos de nuestro cliente, que en este caso serán nuestros compañeros del ciclo formativo de educación infantil. Además, contaremos con el apoyo del ciclo formativo de marketing para realizar una correcta publicidad de la aplicación.

## Palabras clave

Desarrollo de aplicaciones web, Módulos formativos, Desarrollo Web en entorno Cliente, Desarrollo web en entorno servidor, Diseño de interfaces Web, Despliegue de aplicaciones Web, Front-end, Back-end, HTML, CSS, JavaScript, PHP, SQL, Conexión Asíncrona, AJAX, SCRUM, Sprint, XAMPP.

## Índice

<b>Introducción .....</b>	<b>2</b>
<b>Palabras clave.....</b>	<b>4</b>
<b>Índice .....</b>	<b>5</b>
<b>Módulos formativos aplicados en el trabajo .....</b>	<b>7</b>
<b>Herramientas y lenguajes utilizados .....</b>	<b>8</b>
Herramientas de diseño y prototipado .....	8
Herramientas de desarrollo.....	9
Lenguajes de programación utilizados .....	10
Fuentes de información .....	10
<b>Fases del proyecto .....</b>	<b>11</b>
Estudio de mercado .....	11
Análisis de la aplicación .....	14
Funcionalidades: .....	16
Diagrama de entidad-relación .....	38
Modelo relacional.....	39
Diagrama de casos de uso .....	40
Diseño de interfaces .....	41
Sistema de diseño.....	41
Mokups .....	44
Prototipo interactivo:.....	49
Planificación del desarrollo: .....	49
Diagrama de Gantt:.....	50
<b>Conclusiones y mejoras del proyecto .....</b>	<b>51</b>
Objetivos alcanzados con el proyecto:.....	51
Principales hallazgos: .....	51
Impacto y beneficios:.....	51
Retos enfrentados: .....	52
Exigencias del cliente .....	52

<b>Estado actual .....</b>	<b>52</b>
<b>Trabajos futuros.....</b>	<b>53</b>
<b>Bibliografía.....</b>	<b>54</b>
<b>Anexos.....</b>	<b>55</b>
<b>Despliegue en servidor .....</b>	<b>55</b>
<b>Modelo relacional de la base de datos: .....</b>	<b>57</b>
<b>Pruebas automáticas:.....</b>	<b>62</b>

## Módulos formativos aplicados en el trabajo

**Diseño de interfaces Web**, en este módulo, se aborda el proceso de conceptualización y creación de las interfaces que los usuarios interactuarán directamente. Este módulo se enfoca en cómo hacer que la aplicación sea atractiva, intuitiva y funcional. Para ello realizaremos: Investigación y análisis de usuarios, creación de wireframes y mockups, experiencia de usuario, interacción de usuarios.

**Desarrollo web en entorno cliente**, en este módulo, abarcaremos, la programación en JavaScript, para realizar validaciones de usuario como formularios, tanto en tiempo real como cuando al enviarlo y realizar además interactividad y dinamismo en la aplicación, dando animaciones y alguna funcionalidad inmersiva con la aplicación.

**Desarrollo web en entorno servidor**, en este módulo, abordaremos la parte de programación del servidor, utilizaremos el lenguaje de PHP, para conectar la aplicación con una base de datos SQL y creando las consultas necesarias para la correcta conexión entre el cliente y el servidor, además usaremos AJAX para la conexión asíncrona de cliente y servidor.

**Despliegue de aplicaciones Web**, en este módulo, se abordará la creación de un servidor y el despliegue de la aplicación en uno, este servidor será un servidor web apache, el cual configuraremos para que los usuarios puedan entrar a ciertas páginas de la aplicación si tienen permiso o no. La aplicación será accesible desde la red y no de manera local con localhost, sino con su propia dirección.

**Bases de datos**, en este módulo emplearemos lo aprendido para realizar una base de datos funcional SQL para albergar la información de los usuarios de la aplicación.

**Lenguaje de marcas**, en este módulo abarcaremos toda la parte de creación de la página y estilo de ella, utilizaremos los lenguajes de HTML y CSS para desarrollarla.

**Entorno de desarrollo**, de este módulo utilizaremos la parte aprendida de Git para crear un repositorio en el que albergaremos los ficheros de la aplicación y llevar una buena gestión de versiones de la aplicación.

## Herramientas y lenguajes utilizados

Para el desarrollo de este proyecto, hemos utilizado diversas herramientas y lenguajes de programación que nos han permitido diseñar, estructurar y desarrollar la aplicación web de manera eficiente.

### Herramientas de diseño y prototipado

- **Figma:** Se ha utilizado para el estudio de los usuarios, la elaboración de la guía de estilo, la creación de wireframes y la generación de mockups interactivos. Además, dentro de Figma, hemos empleado herramientas como **Imagen IA Generate**, que nos ha facilitado la creación de imágenes mediante inteligencia artificial, optimizando el proceso de diseño. Enlace a figma:

<https://www.figma.com/design/3aLE6ziIU2TKafIEQMLCPE/Peque%C3%B1os-Navegantes?node-id=0-1&t=ISDdLlcOvuDTqglQ-1>

- **Canva:** Hemos recurrido a esta herramienta para la elaboración de material gráfico complementario, como banners, iconos y elementos visuales utilizados en la interfaz de la aplicación.
- **Draw.io:** Esta aplicación, la hemos utilizado a la hora de realizar los diferentes diagramas utilizados en la memoria.
- **DALL-E:** Hemos utilizado esta herramienta de IA para generar imágenes para ayudarnos a realizar el logo.
- **Freepik:** Se ha empleado como fuente de imágenes libres de derechos de autor para enriquecer la estética de la aplicación.
- **Google Fonts:** Biblioteca en línea de fuentes tipográficas gratuitas de Google, utilizada para mejorar el diseño de sitios web y documentos.
- **Microsoft Word:** Procesador de textos de Microsoft que permite crear, editar y dar formato a documentos con herramientas avanzadas de escritura y diseño.

- **Paleta de colores online:** Herramienta digital que ayuda a seleccionar combinaciones de colores armoniosas para diseño gráfico, web o branding.
- **Combinar Fuentes online:** Herramienta web que sugiere combinaciones de fuentes tipográficas compatibles para mejorar la estética de un diseño.
- **Adobe illustrator:** Lo utilizamos para realizar editar y modificar imágenes y hacer imágenes en vectores.
- **Ganttproject:** herramienta de gestión de proyectos que permite crear diagramas de Gantt para planificar y visualizar el progreso de las tareas a lo largo del tiempo. Facilita la asignación de recursos, la definición de dependencias entre actividades y la exportación de informes en diversos formatos.
- **Trello:** herramienta de gestión de proyectos basada en tableros visuales que permite organizar tareas de manera sencilla y colaborativa. Utiliza listas y tarjetas para representar el flujo de trabajo, facilitando la asignación de responsabilidades, el seguimiento del progreso y la planificación de actividades.

### Herramientas de desarrollo

- **Visual Studio Code:** Este ha sido el editor de código utilizado para la escritura y estructuración del código fuente de la aplicación. Para el desarrollo nos hemos ayudado de extensiones dentro del editor como:

- PHP intelephense
- PHP debug
- Live server
- JavaScript Debugg
- IntelliSense
- HTML format
- CSS Formatter
- Auto rename
- Auto close tag

- **XAMPP:** Se ha empleado como entorno de pruebas, ya que proporciona un servidor Apache y una base de datos MySQL, permitiendo la ejecución y testeo de la aplicación en un entorno local antes de su despliegue.
- **GitHub:** La aplicación se ha gestionado a través de un repositorio en GitHub, lo que ha permitido llevar un control de versiones, realizar copias de seguridad del código y colaborar en equipo de manera eficiente, aquí puedes acceder a nuestro repositorio:  
<https://github.com/Astaarr/PequenosNavegantes.git>

### Lenguajes de programación utilizados

- **HTML:** Se ha utilizado para la estructura y el contenido de la aplicación web.
- **CSS:** Se ha empleado para aplicar estilos, definir el diseño visual y mejorar la experiencia del usuario.
- **JavaScript:** Ha sido clave para dotar de dinamismo a la aplicación, así como para la validación de formularios y la interacción con el usuario.
- **PHP:** Este lenguaje se ha utilizado para la conexión con la base de datos, permitiendo la gestión del servidor y la ejecución de consultas.
- **SQL:** Se ha utilizado para la creación y administración de la base de datos, gestionando la información almacenada en la aplicación.
- **AXIOS:** Es una librería de JavaScript que se usa para hacer solicitudes HTTP desde el navegador o desde Node.js que permite enviar y recibir datos de servidores de forma sencilla, trabajando con promesas para manejar las respuestas. Es comúnmente utilizada para conectarse a APIs y obtener o enviar información en formato JSON.

### Fuentes de información

Además de las herramientas mencionadas, hemos recurrido a diversas fuentes de información en internet para documentarnos sobre buenas prácticas de desarrollo, soluciones a problemas específicos y referencias de diseño y usabilidad.

## Fases del proyecto

### Estudio de mercado

En esta investigación documental sobre los campamentos, hemos estudiado una variedad de campamentos, las características, tendencias y oportunidades de estos. Nuestro objetivo principal es entender qué buscan las familias en estos campamentos y qué factores determinan su elección. También queremos identificar áreas de mejora y oportunidades de innovación en cada tipo. Para este estudio, nos hemos centrado en 4 tipos en concreto de campamentos, Inmersión lingüística, deportivos, educativos y de aventuras.

**Campamentos de inmersión lingüística.** Es una experiencia educativa en la que los participantes se sumergen completamente en un nuevo idioma a través de actividades interactivas y divertidas. Durante el campamento, los niños y jóvenes participan en juegos, talleres, y proyectos en equipo que fomentan el uso constante del idioma objetivo. Además de mejorar sus habilidades lingüísticas, los participantes desarrollan competencias comunicativas, culturales y sociales.

**Campamentos deportivos.** Es un programa en el que los participantes practican y aprenden diferentes disciplinas deportivas de manera activa y divertida. A través de juegos, entrenamientos, y actividades en equipo, los niños desarrollan habilidades físicas, técnicas y sociales, además de promover valores como el trabajo en equipo, la superación personal y el respeto por los demás, todo en un entorno dinámico y saludable.

**Campamentos educativos.** Es un programa diseñado para que los niños aprendan y desarrollen nuevas habilidades de forma entretenida y participativa. A través de actividades temáticas como ciencias, arte, tecnología o idiomas, se fomenta el aprendizaje práctico y la creatividad, complementando la educación tradicional. El objetivo es que los niños adquieran conocimientos de forma divertida, potencien su curiosidad y descubran nuevas áreas de interés en un ambiente social y de apoyo.

**Campamentos de aventura.** Es un programa centrado en actividades al aire libre que fomentan la exploración, el contacto con la naturaleza y el desarrollo de habilidades físicas y de supervivencia. Los participantes disfrutan de experiencias como senderismo, escalada, tirolina, orientación y actividades acuáticas, promoviendo el trabajo en equipo, la autonomía y el espíritu de aventura.

Cada tipo de campamento responde a necesidades y preferencias específicas de las familias. Actualmente, la personalización, la seguridad y la innovación son factores clave para captar la atención del mercado.

Los campamentos urbanos son programas diseñados para niños y adolescentes que se desarrollan dentro del entorno urbano, permitiendo a los participantes regresar a casa cada día. En estos los niños pueden tener experiencias únicas y enriquecedoras, estar en un entorno seguro y profesional fomentar sus habilidades sociales, realizar nuevas amistades y fomentar su diversión y aprendizaje.

La temática elegida para nuestro proyecto es un campamento marino.

los principales usuarios para nuestra aplicación son Padres, niños y monitores. Los padres, principalmente entre 30 y 45 años, que buscan actividades para sus hijos durante las vacaciones escolares. Muchos de ellos trabajan y necesitan soluciones de confianza para cuidar a sus hijos durante el día. Niños entre 6 y 7 años, que son los destinatarios finales del campamento. Aunque no suelen tomar la decisión de asistir, es importante que las actividades les resulten atractivas. Profesionales de la educación interesados en colaborar con el campamento o en buscar oportunidades laborales temporales.

Para el estudio de la competencia, hemos realizado un benchmark con dos aplicaciones de campamentos de las cuales hemos sacado las siguientes conclusiones.

La primera aplicación es <https://aulajoven.com> de la cual sacamos conclusiones positivas como el slider con imágenes de calidad e indicado como moverte a través de el botón de buscar para un acceso más rápido a la información que el usuario quiere header estático, para que el usuario pueda ir directamente a la opción que quiera sin tener que subir hasta arriba de la página web, fotos de buena calidad y con efecto cuando pasas por encima el ratón, incluye información necesaria como ubicación, número de teléfono y correo electrónico, es muy minimalista y simple, lo necesario para una versión para el móvil mantiene el logo en pequeño arriba y con una cruz bastante intuitiva para el usuario pueda cerrarlo y El sistema para buscar está muy bien implementado.

Además de conclusiones negativas como, difícil de leer el logo ya que tiene casi el mismo color que el fondo el texto es difícil de leer por el poco contraste que hace, no tiene CTA, lo que complica al usuario la experiencia de apuntarse, mucho texto que puede aburrir a los clientes y difícil de leer por el color del texto y del fondo, el texto debería de estar justificado para que sea algo más fácil de leer y lleve una simetría, el texto de algunas imágenes en color blanco no se puede leer, mala elección de tonalidad de color para usar texto de color blanco, footer muy cargante y muy grande, se podría optimizar mucho más y que el usuario no se agobie con tantas cosas, los elementos no están bien alineados.

La segunda aplicación es [aventurarte.es](http://aventurarte.es) la cual se sacan las siguientes conclusiones positivas, La página web presenta un diseño moderno y atractivo que destaca desde el primer momento. El video interactivo en el header capta la atención del usuario de manera dinámica, mientras que la imagen con relieve sobre el video añade profundidad y dinamismo visual. El contraste de colores está muy bien logrado, y se aprecia una paleta vibrante y armónica que mejora respecto a los tonos de la cabecera. La estructura es simple, correcta y funcional, lo que facilita la navegación y ofrece una experiencia intuitiva.

Además de conclusiones negativas como, mala elección paleta de colores, demasiada cantidad de colores, mala distribución de los elementos de la cabecera poco contraste entre el video y los textos, demasiados colores, el fondo amarillo produce un mal contraste con el botón CTA por ejemplo, y desentonan con el azul del cuadro de imágenes, demasiado texto y elección errónea del color de fondo.

Además del estudio de aplicaciones de otros campamentos, se ha estudiado el perfil de futuros usuarios de la aplicación, hemos creado el perfil de un padre, un niño y un monitor para estudiar un poco sobre cómo podrían ser y las necesidades que estos tendrían con respecto a la aplicación, sus objetivos, frustraciones y motivaciones de esta.

Después se realizó un estudio del punto de vista de cada usuario ideando posibles soluciones para sus preocupaciones, se realizó mediante una lluvia de ideas sobre cómo se podría ayudar con nuestra aplicación a los futuros usuarios y mediante un gráfico en escala al esfuerzo y el impacto de cada posible solución se pudo estudiar que sería bueno implementar y que no para que los usuarios estén a gusto a la hora de usar nuestra aplicación web.

### Análisis de la aplicación

En este apartado se describe cómo se llevó a cabo la implementación de la aplicación web, detallando sus funcionalidades principales y el proceso de ejecución. En esta sección, se explica el funcionamiento de la aplicación, las herramientas y tecnologías utilizadas, y los pasos seguidos para su desarrollo. Además, se incluyen aspectos como la arquitectura del sistema, la interacción del usuario y las pruebas realizadas para asegurar su correcto desempeño.

La aplicación tiene como objetivo ofrecer un campamento para niños de 6 a 7 años a empresas, colegios, ayuntamientos y padres con el fin de organizarlos.

Los usuarios y roles para esta aplicación son padres, monitores, niños y empresas que quieran tanto trabajar en el campamento o contratar el campamento.

Los requisitos impuestos para el desarrollo de la aplicación son, la creación de una aplicación web con temática acuática, marina y de playa. Este campamento será en la ciudad de Madrid, usar unos colores relacionados con el agua, mar, arena playa y sol. Los requisitos técnicos de la aplicación son, implementación de un apartado inicio en el cual se pueda observar un poco la descripción del campamento, las tarifas del campamento, las instalaciones, antiguas ediciones y poder inscribirse.

Además, una barra de menú en la que podamos iniciar sesión, e ir a apartados de la aplicación. También se necesita que la aplicación se pueda acceder desde un ordenador como un dispositivo móvil, se necesitará que sea responsive.

Uno de los requisitos más importantes es que la aplicación sea accesible a todas las personas. Los requisitos de la UI son, una gama de colores que combinen, una buena ambientación con el tema, un apartado de contacto, y un hilo conductor que se basara en “Los pequeños navegantes, han quedado atrapados en una isla en la que para poder salir de ella y ser rescatados tendrán que pasar una serie de retos y pruebas. ¿Serán los navegantes capaces de lograrlo?”.

Los requisitos de seguridad son, poder crear una cuenta cifrada para usuarios de la aplicación, usar un buen protocolo de seguridad, el tratamiento de datos, sin derechos de autor y no exponer imágenes ni datos sensibles. En conclusión, se necesitará una página que sea llamativa, accesible para todos los públicos, bilingüe y responsive.

En ella se podrán realizar diversas consultas como padre o empresa se podrá: registrarse, contactar con el personal, reservar un campamento, tener información de las actividades destacadas, antiguas ediciones del campamento, ver reseñas de persona. Como monitor podremos unirnos como trabajadores, observar el funcionamiento del campamento, actividades que se realizan o la gestión que se utiliza para organizar a los niños.

La funcionalidad de la aplicación debe contener una página principal o home en la que podremos observar información acerca del campamento, alguna información destacada sobre las actividades, precios y reseñas.

Además de esto también contara con un menú donde podremos dirigirnos a secciones de ella en específico como, por ejemplo, más información sobre el campamento, contacto, información del personal, algunas antiguas ediciones que se realizaron del campamento. Debe tener un inicio de sesión para cada miembro, Administrador, padres, colegios, empresas, o miembros del personal. En las cuales iniciarían sesión para observar información respecto a su cuenta y lo que el pude hacer en la aplicación.

Esta debe ser muy fácil de entender para las personas y no muy confusa con su funcionalidad, debe ser clara y no liosa de entender para todo tipo de personas.

La aplicación debe ser segura de usar, puesto que en ella se realizarán inicios de sesión, para introducir datos importantes y sensible sobre menores y mayores de edad. Los usuarios deben sentirse seguros de que sus datos no estén a disposición de todo el mundo.

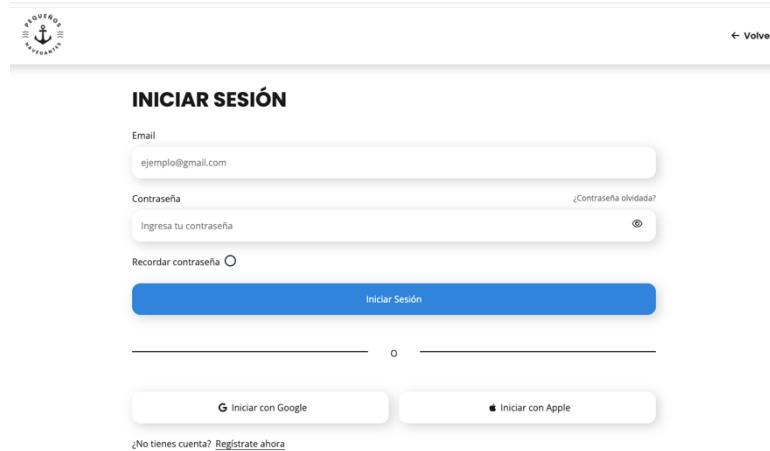
El rendimiento debe ser optimo, lo importante es que la aplicación no cuente con cuelgues o deficiencia a la hora de su desarrollo, debe ser estable y no confuso, no nos lleva a otro apartado si seleccionamos uno en concreto.

En la aplicación, para el inicio de sesión como usuario también se dispondrá de la opción de entrar y registrarse con la cuenta de Google y Apple ID pero esta quedara incluida en la opción de futuros inclusiones a la aplicación.

Las funcionalidades técnicas de formularios, en ellas se explicará su funcionamiento y como están programadas.

### Funcionalidades:

#### Página del login:



The screenshot shows the 'INICIAR SESIÓN' (Login) page. At the top right is a 'Volver' (Back) button. The main form fields are 'Email' (with placeholder 'ejemplo@gmail.com') and 'Contraseña' (with placeholder 'Ingresa tu contraseña'). To the right of the password field is a link '¿Contraseña olvidada?'. Below the fields is a 'Recordar contraseña' checkbox. A large blue 'Iniciar Sesión' (Start Session) button is centered below the inputs. At the bottom of the form are two links: 'G Iniciar con Google' and 'A Iniciar con Apple'.

#### Recuperación de contraseña:

- Abre un popup cuando el usuario hace clic en “Olvidé mi contraseña”.
- Envía el correo de recuperación al servidor.
- Muestra un mensaje de confirmación si el envío es exitoso o una alerta si falla.

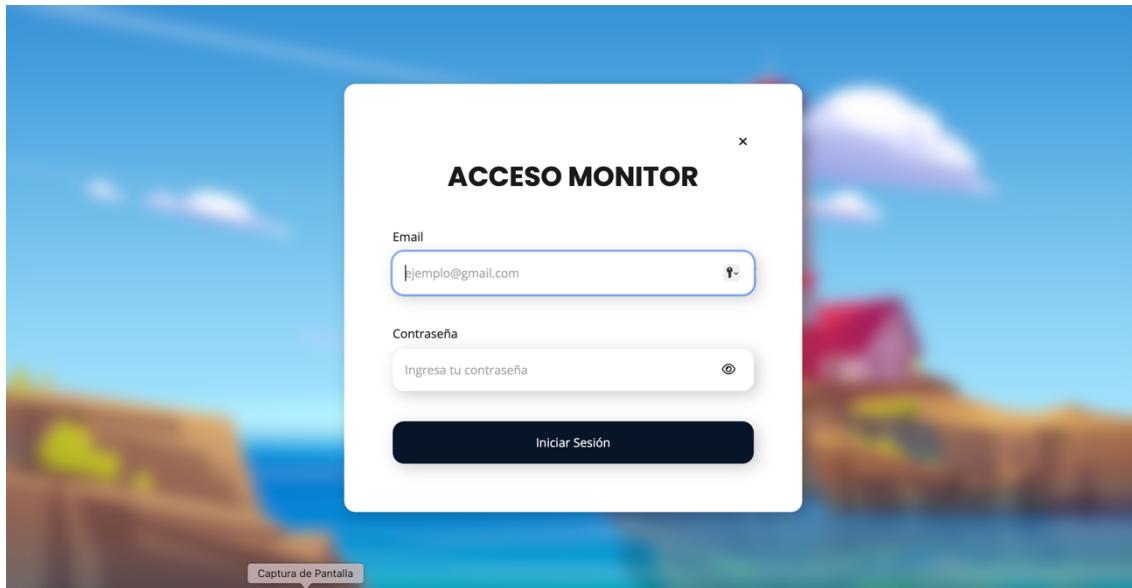
#### Inicio de sesión:

- Captura los datos ingresados en el formulario.
- Envía la información al servidor para validarla.
- Si el inicio de sesión es exitoso, muestra un popup de confirmación y redirige a la página principal.
- Si falla, muestra un mensaje de error.

#### Interacción con el usuario:

- Muestra y oculta popups según las acciones realizadas.
- Valida que los campos no estén vacíos antes de enviarlos.
- Maneja errores y notifica al usuario cuando algo sale mal.

Página del login monitor:



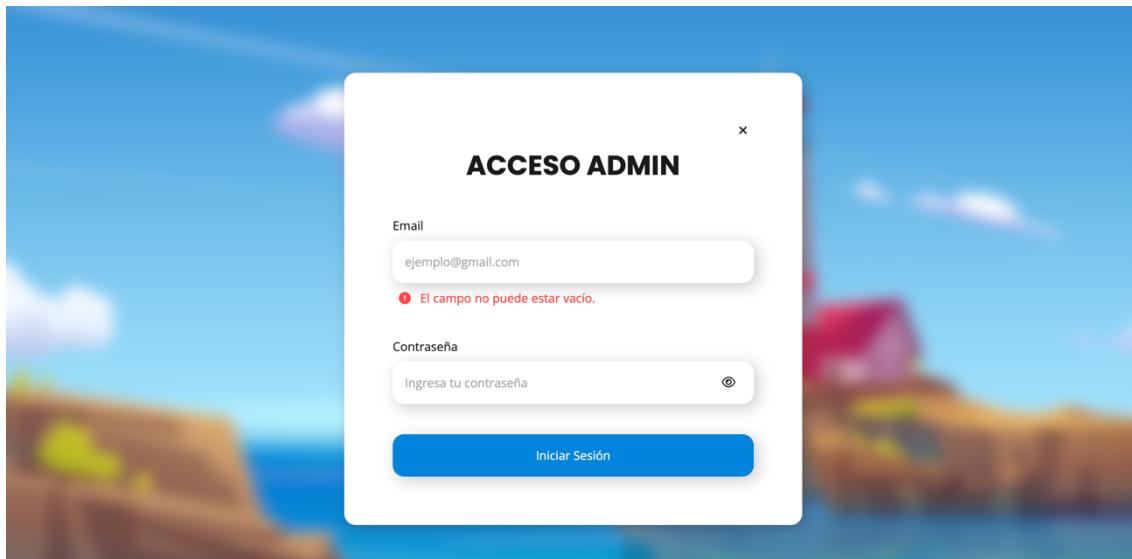
### Inicio de sesión del monitor:

- Captura los datos ingresados en el formulario (correo y contraseña).
- Envía la información al backend en formato JSON.
- Si el inicio de sesión es exitoso, muestra un popup de confirmación.
- Si falla, muestra un mensaje de error indicando que los datos son incorrectos.

### Interacción con el usuario:

- Evita el envío del formulario por defecto para manejar la petición manualmente.
- Muestra y oculta popups según la respuesta del servidor.
- Redirige a la página principal del monitor tras la confirmación.
- Maneja errores y notifica al usuario en caso de problemas con la solicitud.

Página login admin:



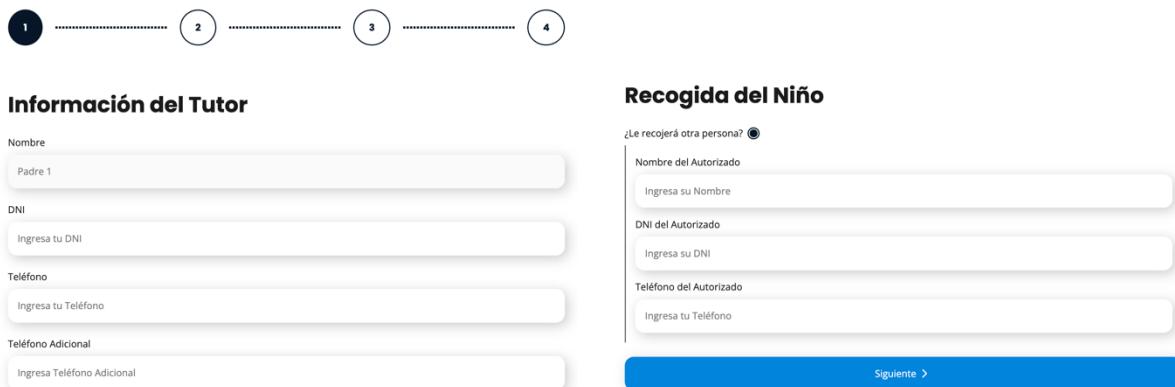
### **Inicio de sesión del administrador:**

- Captura los datos ingresados en el formulario (correo y contraseña).
- Envía la información al backend en formato JSON.
- Si el inicio de sesión es exitoso, muestra un popup de confirmación.
- Si falla, muestra un mensaje de error indicando que los datos son incorrectos.

### **Interacción con el usuario:**

- Evita el envío del formulario por defecto para manejar la petición manualmente.
- Muestra y oculta popups según la respuesta del servidor.
- Redirige a la página de administración tras la confirmación.
- Maneja errores y notifica al usuario en caso de problemas con la solicitud.

Inscribir a un niño:



**Información del Tutor**

Nombre  
Padre 1

DNI  
Ingresa tu DNI

Teléfono  
Ingresa tu Teléfono

Teléfono Adicional  
Ingresa Teléfono Adicional

**Recogida del Niño**

¿Le recogerá otra persona?

Nombre del Autorizado  
Ingresa su Nombre

DNI del Autorizado  
Ingresa su DNI

Teléfono del Autorizado  
Ingresa tu Teléfono

**Siguiente >**

### Autocompletado de datos:

- Al cargar la página, recupera información previamente guardada y la rellena en el formulario.
- Obtiene el nombre del tutor desde el servidor y lo muestra en el campo correspondiente.

### Validación y envío del formulario:

- Verifica que todos los campos sean válidos antes de enviarlos.
- Guarda los datos ingresados en el almacenamiento del navegador.
- Si todo es correcto, redirige a otra página para completar más información.

### Experiencia de usuario optimizada:

- Permite que el usuario no tenga que llenar datos nuevamente si ya los ingresó antes.
- Evita perder información en caso de recarga accidental de la página.
- Proporciona retroalimentación en la consola sobre el estado del formulario.

### Navegación entre páginas:

- Permite retroceder a la sección anterior del formulario.
- Si el formulario es válido, guarda los datos y avanza a la siguiente sección.



**Información del Niño**

Nombre  
Ingrresa nombre del hijo

Apellidos  
Ingrresa apellidos del hijo

Fecha nacimiento  
dd/mm/aaaa

**Información Médica**

¿Presenta alguna alergia?

¿Medicación actual?

Datos adicionales  
Ingrresa datos adicionales hijo

< Anterior      **Siguiente >**

### Autocompletado de datos:

- Recupera y rellena automáticamente los datos previamente guardados en el navegador.

### Validación y almacenamiento de información:

- Verifica que los campos sean correctos antes de enviarlos.
- Guarda los datos ingresados en el almacenamiento del navegador para evitar pérdidas.

### Optimización de la experiencia del usuario:

- Facilita la navegación en el formulario sin perder información.
- Evita que el usuario tenga que volver a escribir datos en caso de recarga de la página.
- Proporciona retroalimentación en la consola sobre el estado del formulario.

**Grumete**  
**Plan Básico**

- ✓ Actividades náuticas y creativas.
- ✓ Monitores especializados.
- ✓ Materiales para talleres.
- ✓ Seguro de accidentes.

**Seleccionar**

**Navegante**  
**Plan Intermedio**  
**+5€**

- ✓ Todo lo del Plan Básico.
- ✓ Desayuno saludable
- ✓ 1 día gratis por cada 10 días

**Seleccionar**

**Capitán**  
**Plan Avanzado**  
**+15€**

- ✓ Todo lo del Plan Medio.
- ✓ Desayuno saludable y comida completa.
- ✓ 1 día gratis por cada 5 días

**Seleccionar**

### Funciones principales:

- Selección de días: Los usuarios pueden seleccionar y deseleccionar días en el calendario.
- Guardado de selección: Los días seleccionados se almacenan en el navegador para que no se pierdan al recargar la página.
- Cambio de mes: Se puede navegar entre meses para seleccionar fechas futuras o revisar selecciones pasadas.
- Visualización del precio: Se calcula el precio total en función del número de días.
- Opciones de planes: Los usuarios pueden elegir entre tres planes diferentes.

### Flujo de inscripción:

1. El usuario selecciona los días en el calendario.
2. El sistema muestra el número de días y el precio total.
3. El usuario elige un plan de inscripción.
4. Si no ha seleccionado días, aparece un mensaje de error.
5. Si todo es correcto, la información se guarda y el usuario avanza a la página de resumen de inscripción.

### Optimización de experiencia:

- Se guarda la selección para evitar pérdida de datos.
- El usuario recibe retroalimentación visual en tiempo real.
- Se maneja la navegación entre meses de manera fluida.



## Método de Pago

 Transferencia Bancaria


**18.00€** incl. impo.  
2 días + Plan basico

 Bizum

 Pago efectivo

Finalizar Inscripción

## Principales funcionalidades

### 1. Selección del método de pago:

- Permite cambiar la opción de pago haciendo clic en un div, asegurando que solo uno esté seleccionado a la vez.

### 2. Cálculo del precio total:

- Obtiene los datos almacenados en sessionStorage.
- Aplica descuentos según el plan elegido:
- Plan Intermedio:** 1 día gratis por cada 10.
- Plan Avanzado:** 1 día gratis por cada 5.
- Muestra los datos en una tarjeta resumen.
- Cambia la imagen según el plan seleccionado.

### 3. Validación y envío de datos al backend:

- Verifica si se seleccionó un método de pago.
- Consolida los datos de tarifa, tutor e hijo.
- Envía la información al servidor usando **Axios**.
- Limpia los datos en sessionStorage tras una respuesta exitosa.
- Muestra un popup de confirmación.

Detalle Actividad:

**Detalles de la Actividad**

Modificar detalles de la actividad.

**Grupos**

+ Grupo 1 X

---

**Detalles Actividad**

Hora de inicio  Hora de fin

Lugar

Descripción

**Guardar**

## Principales funcionalidades

### 1. Popup para agregar o editar tarjetas (grupo o monitor)

- Al abrir el popup, si es edición, se llenan los campos con datos existentes.
- Se pueden agregar nuevas tarjetas o editar las existentes.
- Se pueden eliminar tarjetas con un ícono de basura.

### 2. Carga de actividad desde el backend

- Si en la URL hay un idActividad válido (diferente de “nueva”), se hace una petición con **Axios** a getActividad.php para recuperar los datos.
- Se llenan automáticamente los campos del formulario con la información recuperada.

### 3. Guardar cambios en la actividad

- Al enviar el formulario, se recopilan los datos y se envían con **Axios** a guardarActividad.php.
- Si la actividad es nueva, se crea; si es una existente, se actualiza.
- Tras la operación, redirige a ListadoActividades.html

Pantalla de editar mi información:



### Editar Información Personal

Nombre

DNI  
  
● El campo no puede estar vacío.

Teléfono  
  
● El campo no puede estar vacío.

Teléfono Adicional

**Guardar**

#### **Gestión de datos del padre:**

- Obtiene y muestra los datos del padre desde el backend al cargar la página.
- Rellena automáticamente los campos del formulario con la información recibida.

#### **Actualización de datos:**

- Captura los datos ingresados en el formulario.
- Filtra los campos vacíos para no enviarlos.
- Envía la información actualizada al backend.
- Si la actualización es exitosa, muestra un popup de confirmación.
- Si falla, muestra un mensaje de error.

#### **Interacción con el usuario:**

- Evita el envío del formulario por defecto para manejar la petición manualmente.
- Muestra y oculta popups según la respuesta del servidor.
- Redirige a la página de la cuenta del padre tras la confirmación.

Pantalla de cambiar contraseña:



**Cambiar Contraseña**

Contraseña Actual  
Ingrresa tu contraseña actual

Contraseña Nueva  
Ingresa la nueva contraseña

¿Contraseña olvidada?

**Cambiar**

### Cambio de contraseña:

- Captura la contraseña actual y la nueva ingresada en el formulario.
- Valida que los campos no estén vacíos y que la nueva contraseña tenga al menos 6 caracteres.
- Envía la solicitud al backend para cambiar la contraseña.
- Si la contraseña es correcta, muestra un popup de confirmación y limpia el formulario.
- Si es incorrecta, muestra un mensaje de error.

### Recuperación de contraseña:

- Abre un popup cuando el usuario hace clic en “Olvidé mi contraseña”.
- Captura el correo ingresado para la recuperación.
- Envía el correo al backend para generar un código de recuperación.
- Si el envío es exitoso, muestra un popup de confirmación.
- Si falla, muestra un mensaje de error.

### Interacción con el usuario:

- Muestra y oculta popups según la acción realizada.
- Evita que los enlaces recarguen la página.
- Redirige a la página de cuenta del padre tras confirmar el cambio de contraseña.

Pantalla registro:

## CREAR CUENTA

Nombre

Ingresu su Nombre

Email

ejemplo@gmail.com

Contraseña

Ingresu tu contraseña



Crear Cuenta



G Iniciar con Google

A Iniciar con Apple

¿Ya tienes cuenta? [Accede aquí](#)

## Registro de usuario:

- Captura los datos ingresados en el formulario (nombre, correo y contraseña).
- Valida cada campo antes de enviarlo.
- Envía la información al backend para registrar al usuario.
- Si el registro es exitoso, muestra un popup de confirmación.
- Si hay un error, muestra un mensaje de alerta con la causa del fallo.
- Redirige a la página de inicio de sesión tras confirmar el registro.

## Inicio Monitor:

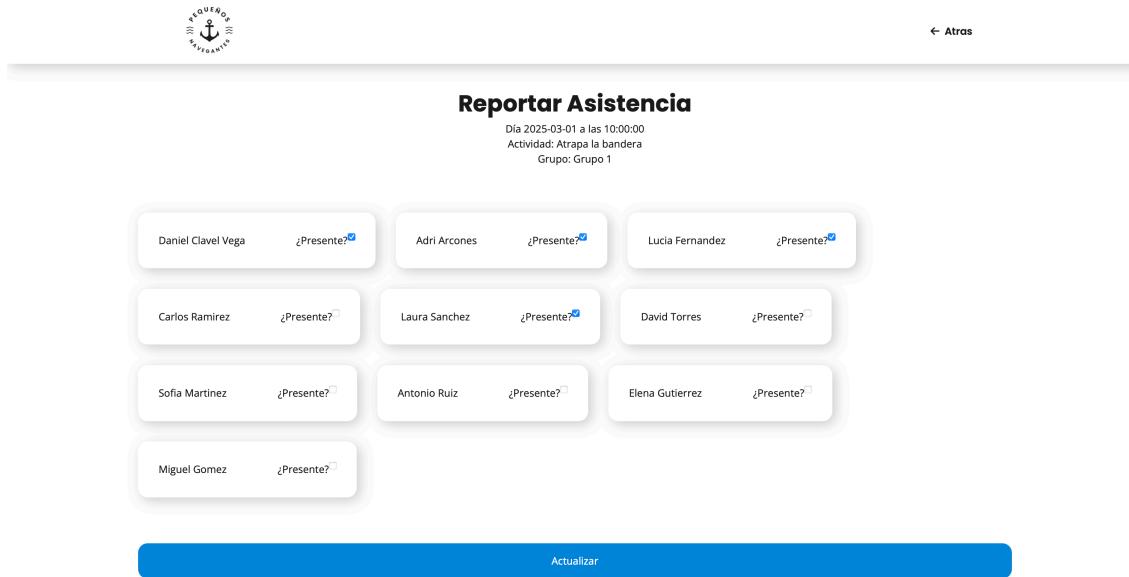
The screenshot shows a monthly calendar for March 2025. The days of the week are labeled L, M, X, J, V, S, D. The dates range from 1 to 31. An event is listed for Saturday, March 1st, at 10:00:00: "Atrapa la bandera". There are also other dates like 10, 11, 12, etc., listed. At the top right, there are buttons for "MIS ACTIVIDADES" and "Juan".

The screenshot shows a detailed view of an activity for March 1, 2025. It includes a back arrow labeled "← Atras", the date "Actividades 2025-03-01", and a list of details: "10:00:00", "Atrapa la bandera", "Grupo 1", "120 mins", and "Patio".

## Carga de Actividades del Día:

- Obtiene la fecha de la URL y la muestra en la interfaz.
- Llama al backend para obtener las actividades programadas en esa fecha.
- Si hay actividades, las muestra en tarjetas interactivas con información detallada.
- Si no hay actividades, muestra un mensaje informativo.
- Permite redirigir a la página de asistencia al hacer clic en una actividad.

Página de asistencia:



The screenshot shows a 'Reportar Asistencia' (Report Attendance) page. At the top right is a back arrow labeled 'Atras'. Below the title 'Reportar Asistencia' are details: 'Día 2025-03-01 a las 10:00:00', 'Actividad: Atrapa la bandera', and 'Grupo: Grupo 1'. The main area contains a 4x2 grid of student names and checkboxes:

Daniel Clavel Vega	<input type="checkbox"/> ¿Presente?	Adri Arcones	<input type="checkbox"/> ¿Presente?	Lucia Fernandez	<input type="checkbox"/> ¿Presente?
Carlos Ramirez	<input type="checkbox"/> ¿Presente?	Laura Sanchez	<input type="checkbox"/> ¿Presente?	David Torres	<input type="checkbox"/> ¿Presente?
Sofia Martinez	<input type="checkbox"/> ¿Presente?	Antonio Ruiz	<input type="checkbox"/> ¿Presente?	Elena Gutierrez	<input type="checkbox"/> ¿Presente?
Miguel Gomez	<input type="checkbox"/> ¿Presente?				

A large blue 'Actualizar' (Update) button is at the bottom.

## Carga de Datos

- Obtiene la lista de niños del grupo desde el backend al cargar la página.
- Muestra la fecha, hora, actividad y grupo en la vista.
- Genera dinámicamente tarjetas con los nombres de los niños y un checkbox para marcar asistencia.

## Guardado de Asistencia

- Al hacer clic en “Guardar”, recopila el estado de los checkboxes.
- Envía los datos al backend para actualizar la asistencia.
- Muestra un popup de confirmación si la actualización es exitosa.

## Cierre del Popup

- Al cerrarlo, recarga la página para actualizar los datos.

Página principal administrador:

The screenshot shows the administrator's main dashboard. At the top, there is a logo with the letters 'UE' and the text 'Universidad Europea'. Below the logo, there are tabs for 'PROGRAMACIÓN', 'ACTIVIDADES', 'GRUPOS', 'MONITORES', and a blue button labeled 'Administrar'. A small anchor icon is visible above the tabs.

**Administrador**

Cerrar Sesión

**Actividades**  
Modifica, añade o elimina actividades para garantizar que cada día sea una experiencia inolvidable.  
[Ver más >](#)

**Grupos**  
Gestiona los grupos de niños de manera sencilla, asegurando que cada uno disfrute al máximo de las actividades.  
[Ver más >](#)

**Monitores**  
Actualiza y organiza los monitores asignados a cada grupo o actividad, garantizando el mejor acompañamiento y supervisión en cada momento.  
[Ver más >](#)

**Marzo 2025**

L	M	X	J	V	S	D
					1 10:00:00 - Atrapa la bandera 10:00:00 - Decorar las espadas	2

## Generación del Calendario

- Al cargar la página, se crea dinámicamente el calendario del mes actual.
- Se muestran los días del mes y se resaltan los espacios vacíos antes del primer día.
- El día actual se marca con una clase especial "hoy".

## Navegación entre Meses

- Funciones `prevMonth()` y `nextMonth()` permiten cambiar de mes.

## Carga de Actividades

- `cargarActividadesDia()` hace una solicitud al backend para obtener las actividades programadas para cada día.
- Si hay eventos, se añaden al día correspondiente en el calendario.

## Interacción del Usuario

- Al hacer clic en un día, se redirige a la vista de programación de ese día.
- Cerrar sesión del administrador.

Actividades admin:



**Listado de Actividades**

Editar actividades

- ▶ Atrapa la bandera
- ▶ Decorar las espadas
- ▶ Barcos de papel
- ▶ Lanzar las anillas
- ▶ Encajar las llaves

## Carga de Actividades

- Al abrir la página, se cargan automáticamente todas las actividades desde el servidor.
- Se muestra cada actividad en una tarjeta con su nombre y detalles ocultos.
- Se incluye un botón para agregar nuevas actividades.

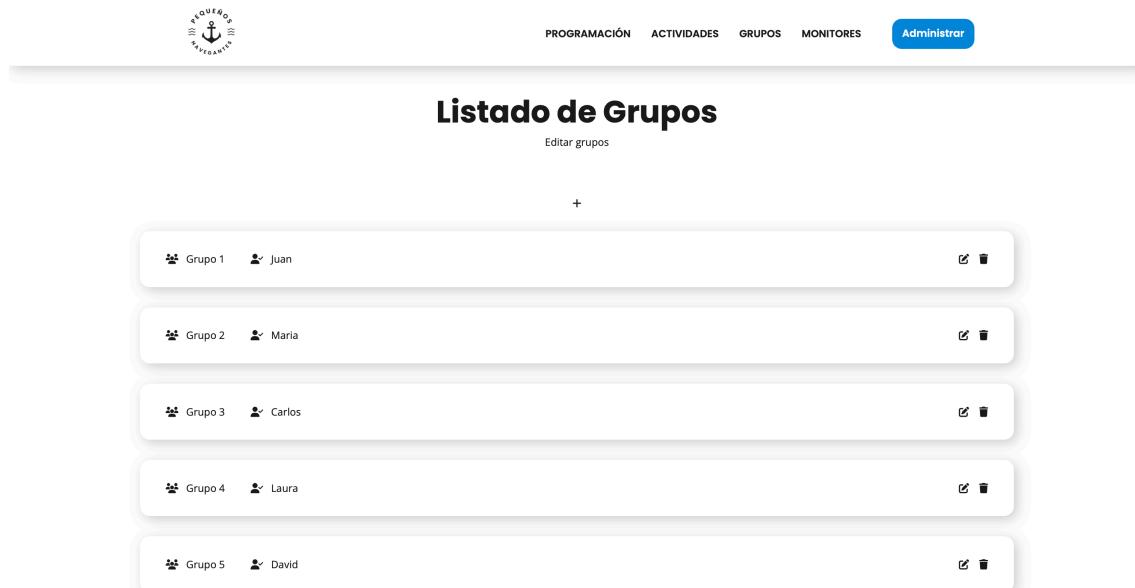
## Interacción con las Tarjetas

- Al hacer clic en una tarjeta, se despliegan los detalles de la actividad.
- Se pueden editar o eliminar actividades mediante los botones correspondientes.

## Agregar, editar o eliminar actividades

- Al presionar el botón “Agregar”, se abre un formulario para ingresar el nombre y la descripción.
- Si se selecciona “Editar”, los datos de la actividad se cargan automáticamente en el formulario.
- Al guardar, los cambios se envían al servidor y la lista se actualiza.
- Al presionar el botón de eliminar, aparece una ventana de confirmación.
- Si se confirma, la actividad se borra del sistema y la lista se actualiza.

Listado grupos administrador:



Grupo	Monitor
Grupo 1	Juan
Grupo 2	Maria
Grupo 3	Carlos
Grupo 4	Laura
Grupo 5	David

## Carga de Grupos

- Al abrir la página, se cargan todos los grupos desde el servidor.
- Se muestran en tarjetas con el nombre del grupo y el monitor asignado.
- Se incluye un botón para agregar un nuevo grupo.

## Interacción con las Tarjetas

- Cada tarjeta tiene botones para editar o eliminar el grupo.
- Al hacer clic en el botón de edición, se abre la página de detalles del grupo.
- Se puede eliminar un grupo tras confirmar la acción en un popup.

## Agregar, editar o eliminar actividades

- Al presionar “Agregar”, se abre la página de detalles para un nuevo grupo.
- Si se edita un grupo, se abre la misma página con los datos precargados.
- Para nuevos grupos, se genera un identificador temporal.
- Al presionar el botón de eliminar, aparece una ventana de confirmación.
- Si se confirma, el grupo se borra del sistema y la lista se actualiza.

Solicitud monitores admin:



The screenshot shows a web application interface for managing monitor requests. At the top, there's a navigation bar with links: PROGRAMACIÓN, ACTIVIDADES, GRUPOS, MONITORES, and Administrar (which is highlighted). Below the navigation is a section titled "Solicitudes de Monitores" with the sub-instruction "Acepta nuevos integrantes para formar grupos:". Two cards are displayed, each representing a monitor request:

- Carlos Gómez Pérez**: Includes an email icon and address (carlos.gomez@example.com), a phone icon and number (644123456), a "Descargar CV" button, and a "Acepto" (Accept) and "Rechazo" (Reject) button.
- Laura Martínez Sánchez**: Includes an email icon and address (laura.martinez@example.com), a phone icon and number (655987654), a "Descargar CV" button, and a "Acepto" (Accept) and "Rechazo" (Reject) button.

## Carga de Monitores

- Al abrir la página, se cargan todos los monitores desde el servidor.
- Se muestran en tarjetas con su nombre, correo y teléfono.
- Cada tarjeta incluye botones para aceptar o eliminar la solicitud del monitor.

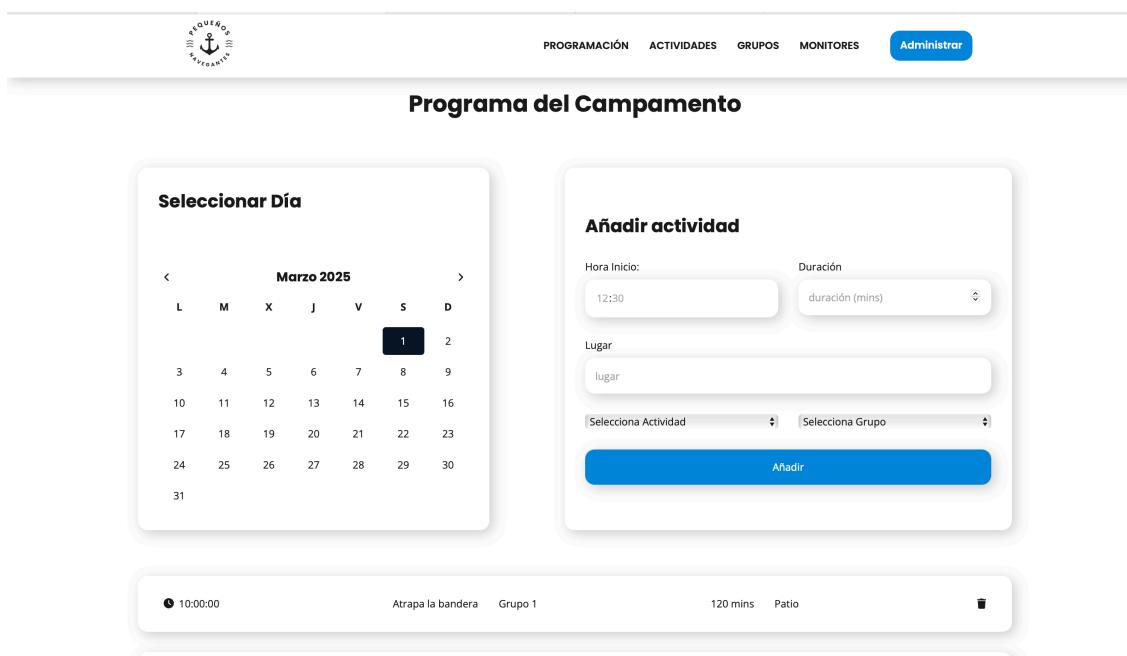
## Interacción con las Tarjetas

- Cada tarjeta permite descargar el currículum del monitor con un clic.
- Al hacer clic en el botón de aceptar, se abre un popup para asignar DNI y contraseña.
- Al hacer clic en el botón de eliminar, se abre un popup de confirmación antes de borrar la solicitud.

## Aceptar o Eliminar Monitores

- Si se acepta un monitor, su tarjeta se elimina de la lista y se envían los datos al servidor.
- Si se elimina una solicitud, la tarjeta desaparece tras confirmar la acción.
- Si ocurre un error en el proceso, se muestra un mensaje de alerta.

Añadir actividad:



The screenshot shows a user interface for managing camp activities. At the top, there's a navigation bar with links to 'PROGRAMACIÓN', 'ACTIVIDADES', 'GRUPOS', 'MONITORES', and a blue 'Administrar' button. Below the navigation is a title 'Programa del Campamento'. On the left, a 'Seleccionar Día' (Select Day) panel displays a calendar for March 2025, with March 1st highlighted. On the right, an 'Añadir actividad' (Add activity) form is shown, containing fields for 'Hora Inicio' (Start Time), 'Duración' (Duration), 'Lugar' (Place), and dropdown menus for 'Selección Actividad' (Select Activity) and 'Selección Grupo' (Select Group). A large blue 'Añadir' (Add) button is at the bottom of the form. Below the form, a summary row shows details for an activity: '10:00:00', 'Atrapa la bandera', 'Grupo 1', '120 mins', 'Patio', and a small trash can icon.

## Carga del Calendario

- Se obtiene la fecha actual o la proporcionada en la URL.
- Se genera el calendario con los días correctamente alineados.
- Los días sin actividad se marcan con una clase especial.

## Interacción con el Calendario

- Los días se pueden seleccionar y marcar como elegidos.
- Se puede navegar entre meses.
- Si no hay días seleccionados, se muestra un mensaje informativo.

## Gestión de Actividades

- Se consultan y muestran las actividades programadas en el día seleccionado.
- Si no hay actividades, se indica con un mensaje.
- Las actividades se listan con detalles como hora, duración y ubicación.
- Cada actividad tiene un botón para eliminarla.
- El formulario permite ingresar una nueva actividad con sus detalles.
- Si hay errores en los campos, se muestra un mensaje de validación.
- Tras guardarse, la lista de actividades del día se actualiza.

### Modelo de datos utilizado

Para la aplicación, se utilizará una base de datos MySQL, para realizar dicha base, tenemos que seguir unos pasos concretos y hacerla de modo que corresponda con la funcionalidad de nuestra aplicación. Primero debemos realizar el modelo Vista controlador, después haremos el modelo relacional. Además de estos dos modelos también realizaremos el diagrama de casos de uso.

Las tablas para usar en la base de datos son:

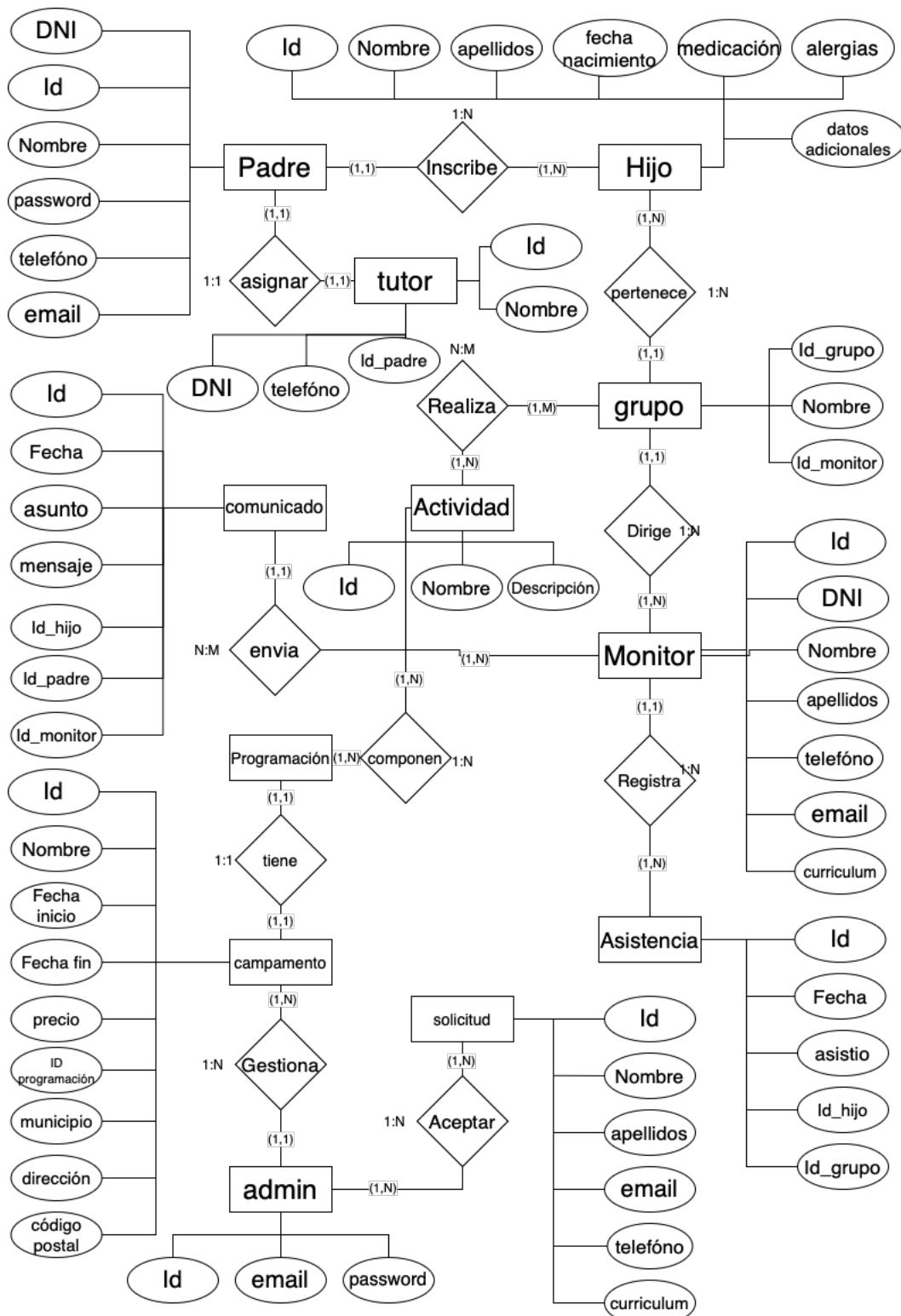
- Actividad: consta de 3 columnas, ID, nombre y descripción. ID es un campo de tipo INT con autoincremento con 11 caracteres de longitud siendo la clave primaria. Nombre es un campo de tipo VARCHAR con 100 caracteres de longitud. Descripción es un campo de tipo TEXT.
- Admin: consta de 3 columnas id\_admin, email y password. Id es un campo de tipo INT de 11 caracteres, con autoincremento además de ser la clave primaria. Email es de tipo VARCHAR con 100 caracteres de longitud. Password es de tipo VARCHAR con 255 caracteres de longitud, cada contraseña esta cifrada.
- Asistencia: consta de 5 columnas, ID\_asistencia, fecha, asistió, ID\_hijo y ID\_grupo. ID\_asistencia es un campo de tipo INT de 11 caracteres de longitud con autoincremento además de ser la clave primaria, fecha es un campo de tipo DATE, asistió es un campo TINYINT con 1 carácter de longitud, ID\_hijo un campo de tipo INT de 11 caracteres es una clave foránea de la tabla hijo y la columna ID\_grupo es de tipo INT de 11 caracteres de longitud es una clave foránea de la tabla grupo.
- Grupo: es una tabla con 3 columnas. Id\_grupo, nombre y ID\_monitor. La columna ID\_grupo es la clave primaria cuenta con 11 caracteres de longitud y es autoincrementada. Nombre es de tipo VARCHAR con 100 caracteres de longitud. ID\_monitor es una clave foránea de la tabla monitor, es un campo INT de 11 caracteres de longitud.

- Campamento: es una tabla que cuenta con 9 columnas, ID\_campamento, nombre, fecha\_inicio, fecha\_fin, precio, ID\_programacion, municipio, dirección y código\_postal. ID\_campamento es la clave primaria es un campo de tipo INT con 11 caracteres de longitud. Nombre es un campo de tipo VARCHAR con 100 caracteres de longitud. Fecha\_inicio y fecha\_fin son campos de tipo DATE. Precio es un campo de tipo DECIMAL (6,2). ID\_programacion es una clave foránea de la tabla programación\_actividad de tipo INT con 11 caracteres de longitud. Municipio es un campo de tipo TEXT, dirección es un campo de tipo TEXT. Código\_postal es un campo de tipo VARCHAR de 5 caracteres de longitud.
- Comunicación: tabla de 7 columnas, ID\_comunicacion, fecha, asunto, mensaje, ID\_padre, ID\_hijo y ID\_hijo. ID\_comunicacion es la clave primaria, de tipo INT de 11 caracteres de longitud. Fecha es un campo de tipo DATE. Asunto es un campo de tipo VARCHAR de 100 caracteres de longitud. Mensaje es un campo de tipo TEXT. ID\_padre es una clave foránea de la tabla padre, es un campo INT de 11 caracteres de longitud. ID\_hijo es una clave foránea de la tabla hijo, es un campo INT de 11 caracteres de longitud. ID\_monitor es una clave foránea de la tabla monitor, es un campo INT de 11 caracteres de longitud.
- Hijo: esta tabla cuenta con 9 columnas, ID\_hijo, nombre, apellidos, fecha\_nacimiento, medicación, alergias, datos\_adicionales, ID\_padre y ID\_grupo. ID\_hijo es la clave primaria cuenta con 11 caracteres de longitud y es autoincrementada. Nombre es una columna de tipo VARCHAR de 100 caracteres de longitud. Apellidos es una columna de tipo VARCHAR de 100 caracteres de longitud. Fecha\_nacimiento es un campo de tipo DATE. Medicación es un campo de tipo TEXT. Alergias es un campo de tipo TEXT. Datos\_adicionales es un campo de tipo TEXT. ID\_padre es una clave foránea de la tabla padre, es un campo INT de 11 caracteres de longitud. ID\_grupo es una clave foránea de la tabla grupo, es un campo INT de 11 caracteres de longitud.

- Inscripción: cuenta con 5 columnas, ID\_inscripción, fecha\_inscripcion, id\_hijo, ID\_campamento, estado\_pago. ID\_inscripción es la clave primaria cuenta con 11 caracteres de longitud y es autoincrementada. Fecha\_inscripcion es de tipo DATE. ID\_hijo es una clave foránea de la tabla hijo, es un campo INT de 11 caracteres de longitud. ID\_campamento es una clave foránea de la tabla campamento, es un campo INT de 11 caracteres de longitud. Estado\_pago es un campo de tipo ENUM y puede estar 'pendiente', 'pagado' o 'no pagado'.
- Monitor: esta tabla cuenta con 8 columnas, id\_monitor, DNI, nombre, apellidos, password, email, teléfono, curriculum. ID\_monitor es la clave primaria cuenta con 11 caracteres de longitud y es autoincrementada. DNI es un campo de tipo VARCHAR con 9 caracteres de longitud. Nombre es una columna de tipo VARCHAR de 100 caracteres de longitud. Apellidos es una columna de tipo VARCHAR de 100 caracteres de longitud. Password es un campo de tipo VARCHAR con 255 caracteres y la contraseña está cifrada. email es una columna de tipo VARCHAR de 100 caracteres de longitud. teléfono es una columna de tipo VARCHAR de 16 caracteres de longitud. Curriculum es un campo de tipo TEXT.
- Padre: cuenta con 9 columnas, ID\_padre, DNI, nombre, password, email, telefono, token\_login, código\_restablecer, token\_restablecer. ID\_padre es la clave primaria cuenta con 11 caracteres de longitud y es autoincrementada. DNI es un campo de tipo VARCHAR con 9 caracteres de longitud. Nombre es una columna de tipo VARCHAR de 100 caracteres de longitud. Apellidos es una columna de tipo VARCHAR de 100 caracteres de longitud. Password es un campo de tipo VARCHAR con 255 caracteres y la contraseña está cifrada. email es una columna de tipo VARCHAR de 100 caracteres de longitud. teléfono es una columna de tipo VARCHAR de 16 caracteres de longitud. Token\_login en esta columna se guardará el token que se creará para el guardado de la contraseña al darle al recordar esta es de tipo VARCHAR y cuenta con 255 caracteres de longitud. Codigo\_restablecer, en esta columna se guardará el código para restablecer la contraseña esta es de tipo Varchar con 255 caracteres de longitud. Token\_restablecer, esta guardara el token necesario para poder restablecer la contraseña, esta es de tipo VARCHAR y cuenta con 255 caracteres de longitud.

- Programación\_actividades: esta tabla cuenta con 6 columnas, id\_programacion, id\_actividad, id\_grupo, hora\_inicio, hora\_fin y lugar. Id\_Programacion es la clave primaria, esta es de tipo INT con 11 caracteres de longitud y cuenta con autoincremento. Id\_actividad es una clave foránea de la tabla actividad, cuenta con 11 caracteres de longitud y es de tipo INT. Id\_grupo es una clave foránea de la tabla grupo, cuenta con 11 caracteres de longitud y es de tipo INT. Hora\_inicio y hora\_fin son dos campos de tipo DATE. Lugar es de tipo VARCHAR cuenta con 255 caracteres de longitud.
- Solicitud\_monitor: esta tabla cuenta con 6 columnas, id\_solicitud, nombre, apellidos, email, teléfono y curriculum. Siendo id\_solicitud la clave primaria un campo de tipo INT con 11 caracteres de longitud y contando con autoincremento, nombre un campo de tipo VARCHAR de 100 caracteres de longitud, apellidos es un campo de tipo VARCHAR de 100 caracteres de longitud, email es un campo de tipo EMAIL, teléfono es un campo de tipo VARCHAR de 16 caracteres de longitud y curriculum es un campo de almacenamiento de ruta de ficheros.
- Tutor\_adicional: esta tabla cuenta con 5 columnas, id\_tutor, DNI, nombre, Telefono\_autorizado y id\_padre. Id\_tutor es la clave primaria un campo de tipo INT con 11 caracteres de longitud y contando con autoincremento. Id\_padre es la clave foránea de la tabla padres un campo INT de 11 caracteres con autoincremento. DNI es un tipo VARCHAR de 10 Caracteres de longitud, nombre es una de tipo VARCHAR de 100 caracteres y apellidos es un Varchar de 100 caracteres.

## Diagrama de entidad-relación



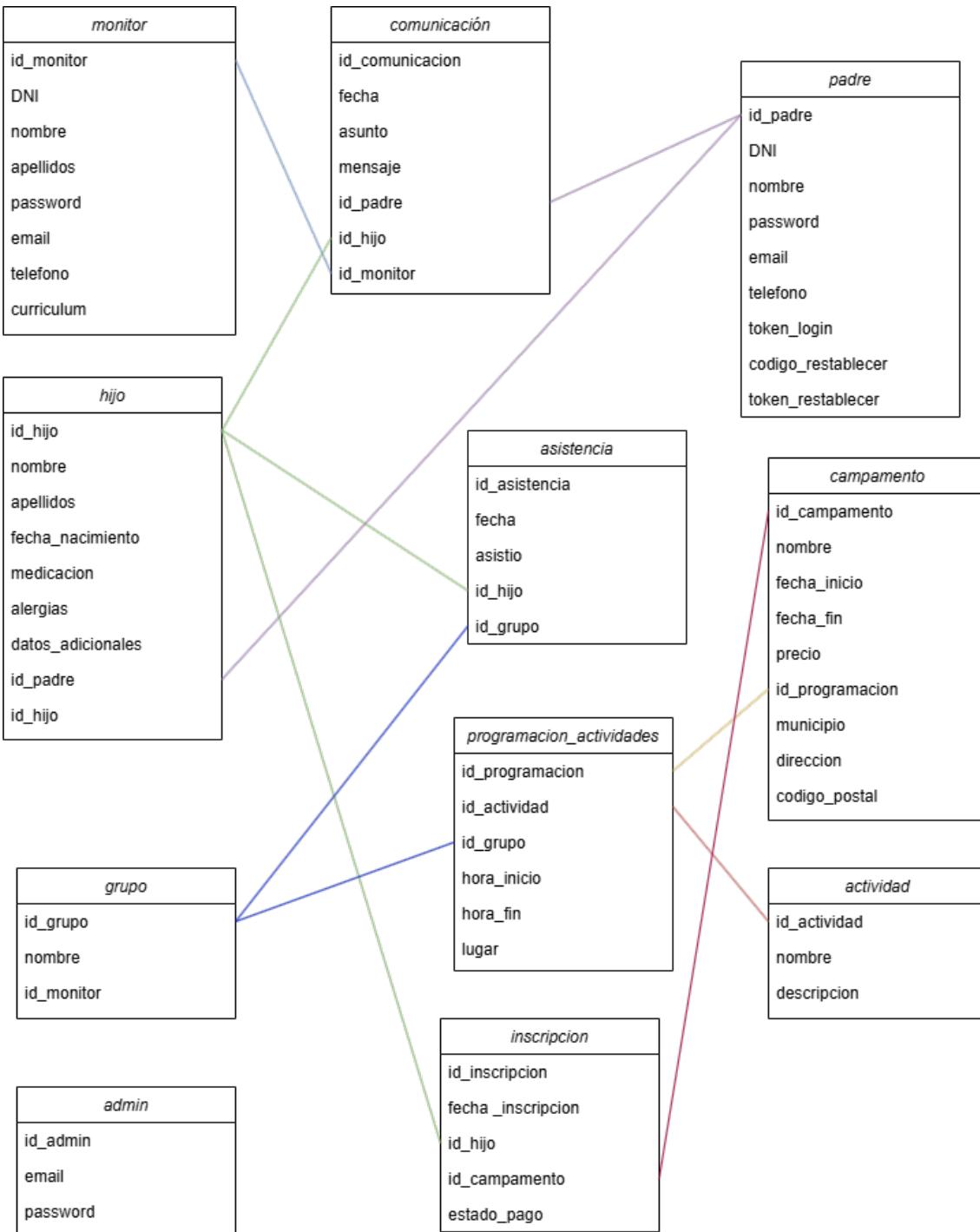
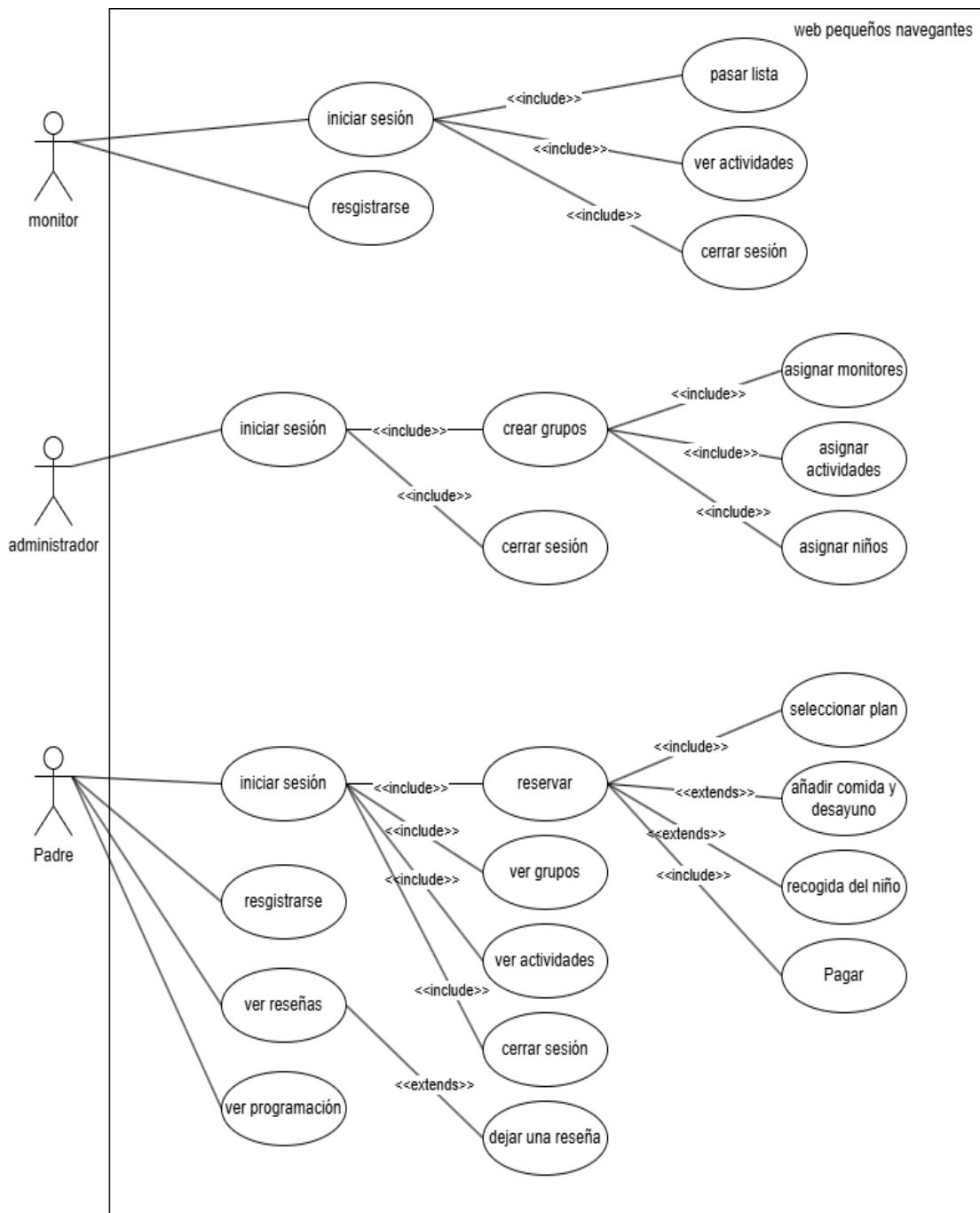
Modelo relacional

Diagrama de casos de uso



### Diseño de interfaces

En este apartado, exploraremos los principios y buenas prácticas del diseño de interfaces, centrándonos en aspectos como la usabilidad, la coherencia visual y la optimización de la experiencia del usuario. Además, se abordará la importancia de los sistemas de diseño para garantizar una identidad sólida y una navegación fluida.

Sistema de diseño: en este apartado abordaremos todo lo referido a la guía y componentes que aseguran la coherencia visual y funcional en la creación de interfaces. Su implementación facilita el trabajo en equipo, optimiza los procesos de diseño y desarrollo, y mejora la experiencia del usuario al mantener una identidad unificada en todos los productos.

- Logo de la aplicación: el ancla central es el elemento principal del diseño, simbolizando estabilidad, aventura y el espíritu marítimo que define al campamento. El texto curvo que se ubica en la parte tanto superior como inferior creando un efecto circular, aporta dinamismo y equilibrio visual. Las olas actúan como separadores son tres ondas a cada lado que refuerzan la conexión con el agua y la tipografía moderna y en mayúsculas que transiten confianza y claridad.



- Paleta de colores: La paleta de colores recoge los principales colores que usa la aplicación, estos colores tienen que combinar de manera que no sea una carga visual para los usuarios de la aplicación, para ello se utilizó una paleta con colores relacionados con el mar, marineros, islas, etc.
  - El color principal es el #0184DC
  - El color secundario es el #071528
  - El color blanco utilizado es el #fffff
  - El color negro utilizado es el #1a1a1a
  - El color gris utilizado es el #222
  - El color de error es el #FF4444
  - Color de disable es el #666666
- Tipografía: Para la aplicación se utilizó dos tipos de fuentes diferentes. En primer lugar, se utilizó la tipografía poppins bold para títulos principales en 56px, para títulos de sección en 48px y para los subtítulos de sección en 24px. Para los énfasis utilizaremos open sans bold a 16px y para el párrafo open sans regular a 24px.

# TÍTULO PRINCIPAL (H1)

Poppins bold 56px

## TÍTULO DE SECCIÓN (H2)

Poppins bold 56px

### SUBTÍTULOS EN LA SECCIÓN (H3)

Poppins bold 24px

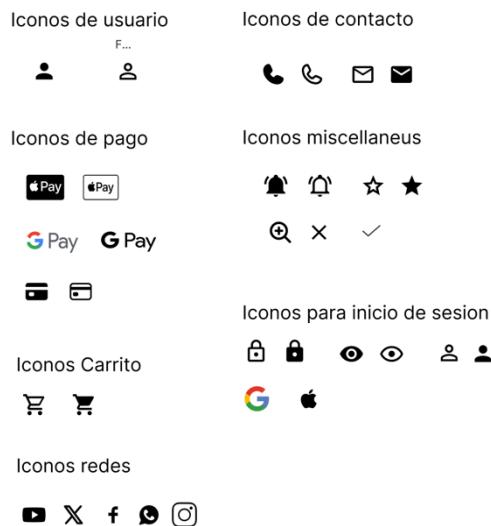
#### Énfasis (b, span)

Open Sans bold 16px

Esto es un párrafo (p)

Open Sans regular 16px

- **Iconografía:** Para la aplicación se utilizó distintos iconos para ayudar a la comprensión de varias secciones o para cambiarlos por frases o palabras que las personas interpretan mejor. Estos deben ser fáciles de leer tanto para niños o como para adultos.



Mokups: En este apartado se mostrará el diseño de las páginas importantes de la aplicación.

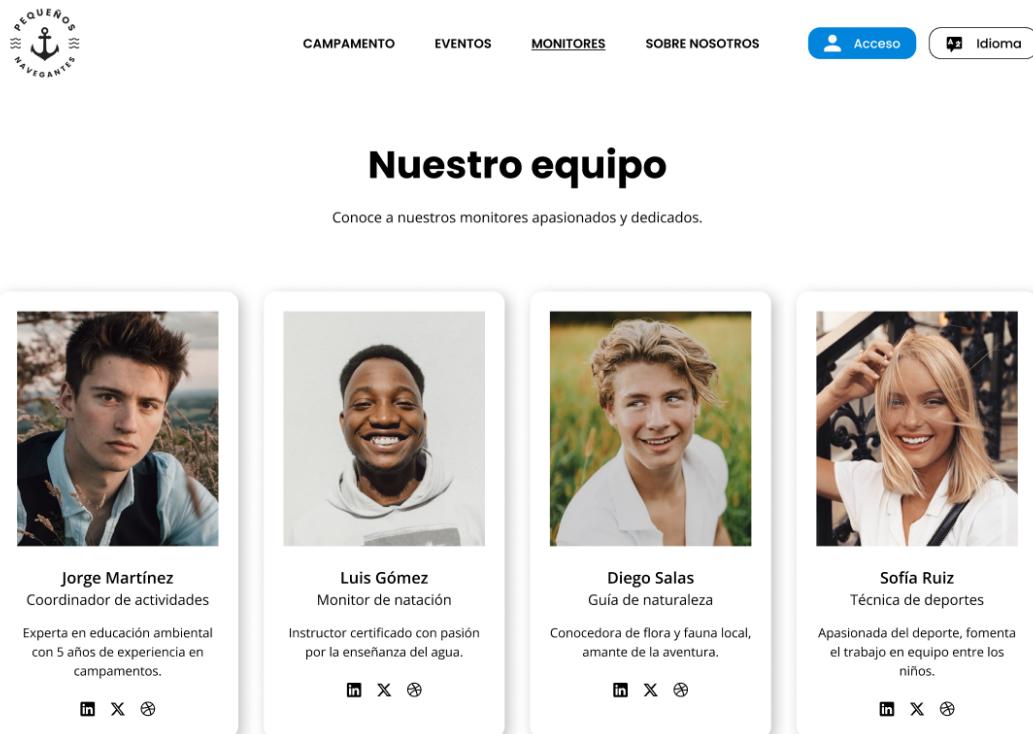
- Página principal:

The main page features a large illustration of a lighthouse on a rocky island in the ocean under a blue sky with white clouds. To the left, there is a logo with the text "PEQUEÑOS NAVEGANTES" and an anchor. Below the logo, a poem reads:  
Lejos de la costa, pero con espíritu náutico:  
Tus pequeños navegantes vivirán aventuras  
acuáticas, diversión y trabajo en equipo  
inolvidables.  
A blue button labeled "¡Zarpa ya!" is positioned below the poem.

- Página eventos:

The events page has a header with the "PEQUEÑOS NAVEGANTES" logo and navigation links for CAMPAMENTO, EVENTOS (underlined), MONITORES, and SOBRE NOSOTROS. It also includes "Acceso" and "Idioma" buttons. A section titled "Galería de Imágenes" with the sub-instruction "Revive los momentos más emocionantes de nuestros campamentos pasados." displays six images: a group of children on a sailboat, children interacting with a large shark, children setting up a blue sailboard on the beach, a group of children standing together, and children standing near a white tent.

- Página monitores:



**Nuestro equipo**

Conoce a nuestros monitores apasionados y dedicados.



**Jorge Martínez**  
Coordinador de actividades

Experta en educación ambiental con 5 años de experiencia en campamentos.

[LinkedIn](#) [X](#) [Instagram](#)



**Luis Gómez**  
Monitor de natación

Instructor certificado con pasión por la enseñanza del agua.

[LinkedIn](#) [X](#) [Instagram](#)



**Diego Salas**  
Guía de naturaleza

Conocedora de flora y fauna local, amante de la aventura.

[LinkedIn](#) [X](#) [Instagram](#)

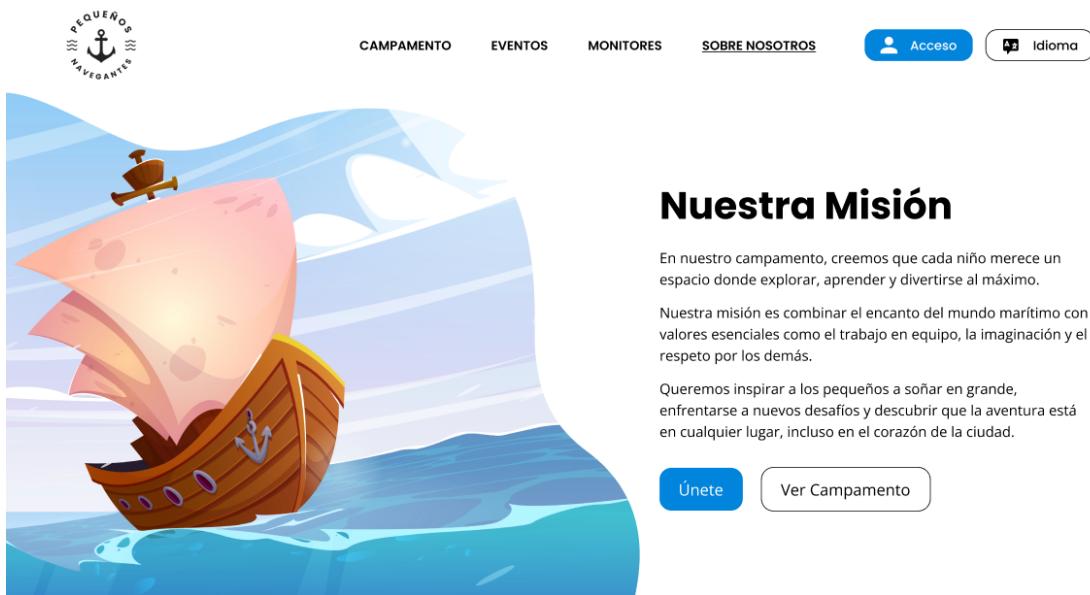


**Sofía Ruiz**  
Técnica de deportes

Apasionada del deporte, fomenta el trabajo en equipo entre los niños.

[LinkedIn](#) [X](#) [Instagram](#)

- Página sobre nosotros:



**Nuestra Misión**

En nuestro campamento, creemos que cada niño merece un espacio donde explorar, aprender y divertirse al máximo.

Nuestra misión es combinar el encanto del mundo marítimo con valores esenciales como el trabajo en equipo, la imaginación y el respeto por los demás.

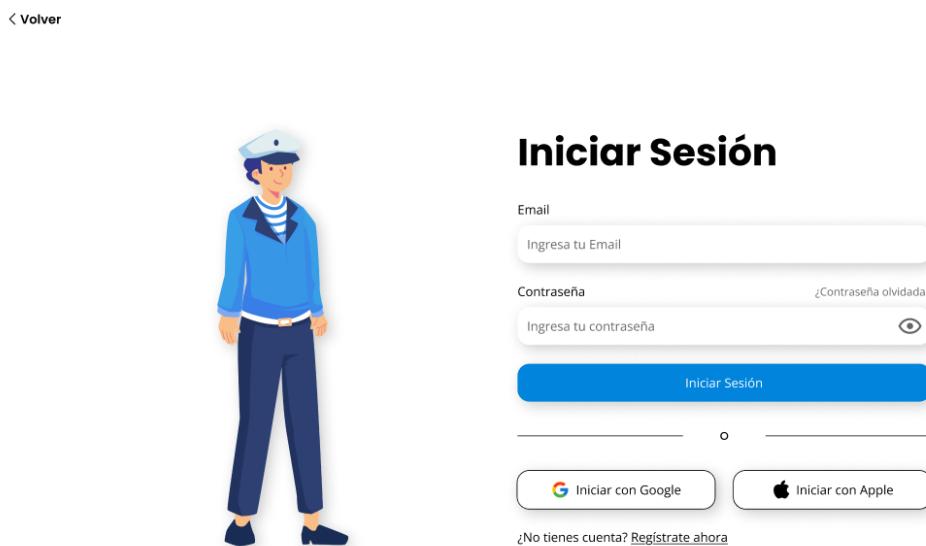
Queremos inspirar a los pequeños a soñar en grande, enfrentarse a nuevos desafíos y descubrir que la aventura está en cualquier lugar, incluso en el corazón de la ciudad.

[Únete](#) [Ver Campamento](#)

- Página de login:



- Página de inicio sesión:



- Página de registro:

< Volver

## Crear Cuenta

Nombre

Ingresá tu nombre

Email

Ingresá tu Email

Contraseña

Ingresá tu contraseña

**Crear Cuenta**



Iniciar con Google

Iniciar con Apple

¿Ya tienes cuenta? [Accede aquí](#)

- Página mi perfil:



CAMPAMENTO

EVENTOS

MONITORES

SOBRE NOSOTROS

Acceso

Idioma

## Mi cuenta



### Información personal

Mantén tus datos actualizados para una experiencia sin contratiempos.

Editar >



### Cambiar contraseña

Tu cuenta es la clave para disfrutar de todas nuestras actividades.

Cambiar >



### Preferencias de comunicación

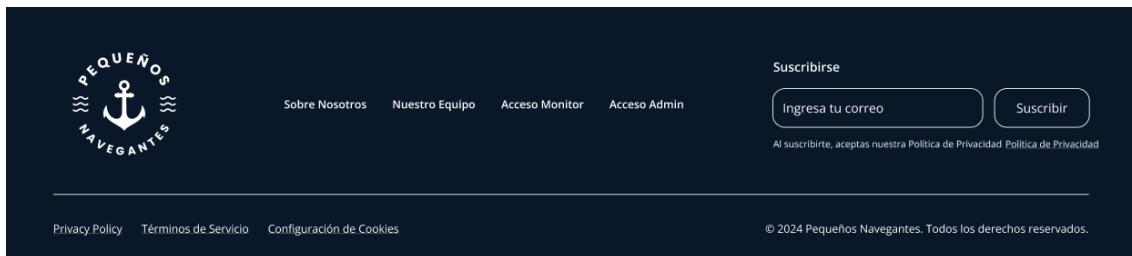
Elegí cómo deseas comunicarte con el equipo y recibir novedades.

Actualizar >

- Header:



- Footer:



- Página de cambio de contraseña:

[◀ Volver](#)



- Página de editar información:

### Información Personal

Nombre	Pablo Gómez	
Correo	pablogomez@gmail.com	
DNI	12345678A	
Teléfono	123 45 67 89	

[Guardar](#)



Prototipo interactivo: Para poder observar el prototipado de la aplicación se deberá acceder a Figma mediante el siguiente enlace. en este podremos observar y probar el funcionamiento del prototipo de nuestra aplicación, y observar alguna de sus funciones.

[Enlace para acceder al prototipo web.](#)

Planificación del desarrollo: En este apartado se explicará la planificación del desarrollo, este es un aspecto fundamental para asegurar la correcta ejecución del proyecto. A través de una estructura organizada, se busca definir las fases del desarrollo, establecer los recursos necesarios y optimizar los tiempos de ejecución. Este apartado tiene como objetivo describir la metodología empleada, las etapas del desarrollo y los plazos estimados, garantizando así un proceso eficiente y alineado con los requisitos del proyecto.

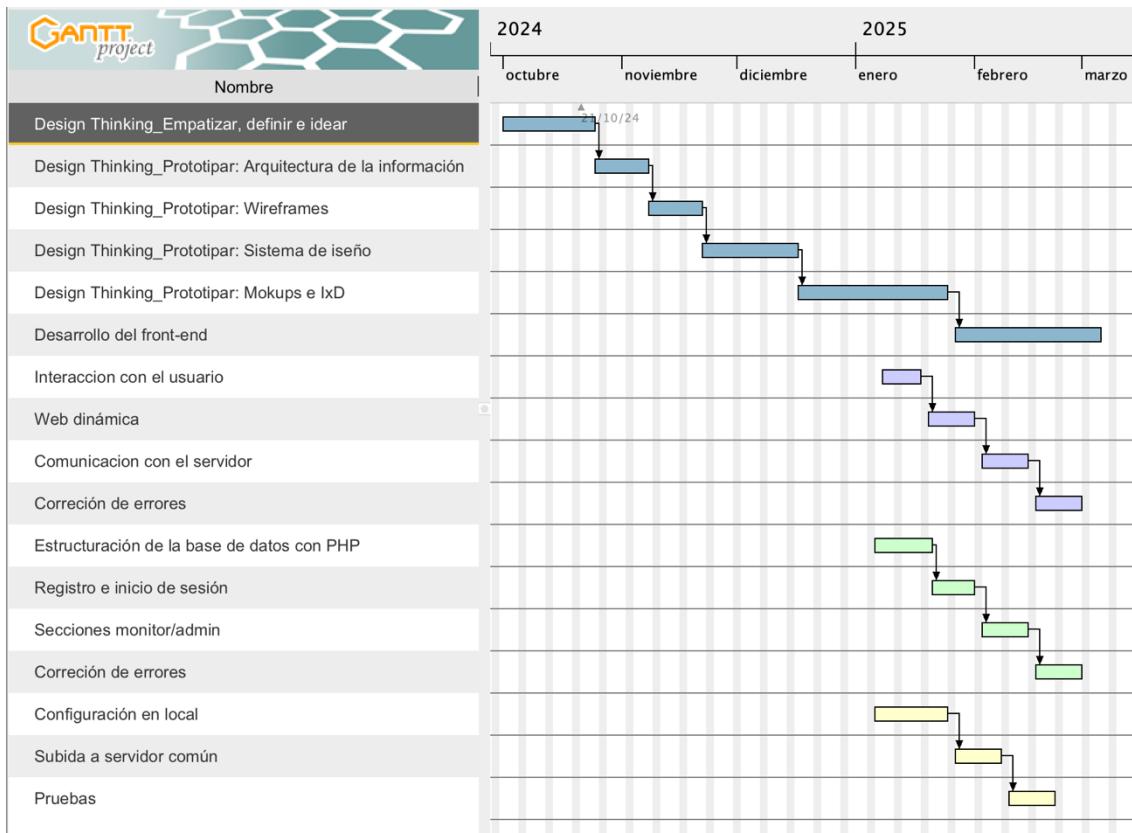
En primer lugar, se explicará la metodología ágil utilizada. En este caso se ha utilizado la metodología SCRUM mediante SPRINTS, teniendo así entregas de las partes a realizar del proyecto a dicha fecha de entrega.

Para ayudarnos a llevar una buena planificación del proyecto hemos utilizado algunas herramientas para realizar diagramas de Gantt o llevar una planificación de las tareas a realizar.

Para realizar un seguimiento de las tareas se ha utilizado trello, una aplicación en la que hemos creado una tabla en la que organizamos las tareas por hacer, las que están en proceso y las que ya están finalizadas.

Para ver el tablero entra en el siguiente [enlace](#).

Además, para llevar un seguimiento del tiempo y la duración de cada aplicación hemos utilizado Gantt Project para realizar un diagrama de Gantt. En él se puede observar detalladamente cuando empezamos cada entrega, cuando se finaliza y además la duración de cada una en días.

Diagrama de Gantt:

## Conclusiones y mejoras del proyecto

### Objetivos alcanzados con el proyecto:

El desarrollo de la aplicación para el campamento ha permitido mejorar la experiencia tanto de los padres como de los niños y monitores. A través de una plataforma intuitiva y funcional, se ha logrado optimizar el acceso a la información, facilitar la inscripción y fortalecer la comunicación entre todas las partes involucradas. A continuación, se detallan los principales objetivos alcanzados durante el proyecto.

Se implementó un inicio de sesión fácil y sencillo para todos los usuarios, solo se pide nombre, email y contraseña para ello, y más adelante cuando quieran inscribir a un hijo se solicitarán más información sobre el parentesco y el propio hijo.

El portal para inscribir a un hijo es intuitivo, consta de 4 pasos, el primero es la información del parentesco o tutor del niño, además podemos incluir un tutor en caso de que el parentesco no pueda recogerlo, el segundo paso es la información del niño, si tiene enfermedades o alergias, el tercer paso es el número de días que asistirá al campamento y el plan para ello y por último paso es el método de pago y confirmación de reserva.

### Principales hallazgos:

Para la realización de este proyecto se ha aprendido, a utilizar lenguajes de programación no vistos como axios, el cual se utiliza para la conexión asíncrona entre lenguajes como PHP y JavaScript. Además de este, se aprendió a utilizar más y a entender y comprender más los lenguajes de programación vistos en clase para realizar el proyecto.

### Impacto y beneficios:

La implementación de la aplicación ha generado un impacto positivo en la gestión del campamento, beneficiando tanto a la organización como a los monitores y a las familias.

Por un lado, la optimización de procesos ha permitido reducir el tiempo dedicado a la gestión de inscripciones y seguimiento de los niños, facilitando el trabajo del equipo organizador. Además, la posibilidad de actualizar la información en tiempo real ha mejorado la seguridad y el control dentro del campamento, permitiendo una respuesta más rápida ante cualquier eventualidad.

Asimismo, la aplicación ha fortalecido la comunicación con los padres, brindándoles información detallada y constante sobre la experiencia de sus hijos. Esto ha contribuido a generar mayor confianza en el servicio del campamento, mejorando la percepción y satisfacción de las familias.

Por último, la digitalización de la información ha reducido la carga administrativa para los monitores, quienes ahora pueden gestionar las actividades y el bienestar de los niños de manera más eficiente. Además, la mejora en la experiencia tanto de los padres como de los niños puede fomentar una mayor fidelización y participación en futuras ediciones del campamento.

#### Retos enfrentados:

Los principales retos que se enfrentaron fueron estructurar y realizar correctamente la estructura de la propia página, la creación correcta de la base de datos, y como configurarla para su correcta inclusión de datos desde la página. Otro reto el cual no se pudo conseguir fue la inclusión de un inicio de sesión funcionar con Google o Apple, después de hacer varias pruebas no se consiguió llegar a la configuración correcta.

#### Exigencias del cliente

Para realizar este proyecto, se necesitaron los principales requisitos que el cliente demandaba para su realización. Entre ellos se destacan, que se realizara una aplicación web para un campamento, con temática de playa. El cliente demandó funcionalidades como una web que fuera accesible para todos, que sea responsive, para poder verla tanto en ordenador como en el móvil, sea compatible con todos los navegadores web.

#### Estado actual

Hoy en día el proyecto es funcional se consiguió llegar a los requisitos y exigencias tanto del cliente como de los propuestos por los profesores. Lo cual ha llevado a realizar una aplicación web funcional para los usuarios que se destina usarán en un futuro.

### Trabajos futuros

Para futuras inclusiones al proyecto, se podrían añadir nuevas funcionalidades que no se tuvo tiempo a añadir. Se podría añadir el inicio de sesión con terceros como Google o Apple, implementar una pasarela de pagos funcional, añadir la funcionalidad de contacto en tiempo real de los padres con los monitores.

## Bibliografía

Stack Overflow. (s.f.). *Stack Overflow*. <https://stackoverflow.com>

The PHP Group. (s.f.). *PHP Manual*. <https://www.php.net/manual/en/index.php>

Axios. (s.f.). *Axios: Promise based HTTP client for the browser and node.js*. <https://axios-http.com/>

Sebastian Bergmann. (s.f.). *PHPUnit Manual*. PHPUnit. <https://phpunit.de/documentation.html>

Coloros. (s.f.). *Coloros: The super fast color palettes generator*. Coloros. <https://coloros.co/>

OpenAI. (s.f.). *DALL·E: AI image generator*. OpenAI. <https://openai.com/dall-e>

YouTube. (s.f.). *YouTube*. <https://www.youtube.com>

Aula Joven. (s.f.). Aula Joven: Campamentos de verano y actividades extraescolares. <https://aulajoven.com/>

Aventurarte. (s.f.). Aventurarte: Campamentos de verano en plena naturaleza. <https://www.aventurarte.es>

## Anexos

### Despliegue en servidor

En este anexo se explicará la parte de despliegue en servidor y la configuración de este. El primer paso, es encender la máquina del servidor donde tenemos instalado apache y en la configuración el primer paso es ir al directorio donde crean los directorios de las aplicaciones web y crearemos el de nuestra aplicación.

```
root@daniserver:/# cd /var/www/html/_
```

Después de dirigirse al directorio donde se alojará el directorio raíz de nuestra aplicación web, creamos el directorio de esta y nos movemos a su interior.

```
root@daniserver:/var/www/html# mkdir pequenosnavegantes
root@daniserver:/var/www/html# cd pequenosnavegantes/
root@daniserver:/var/www/html/pequenosnavegantes# personal/
root@daniserver:/var/www/html/pequenosnavegantes# errores/
root@daniserver:/var/www/html/pequenosnavegantes#
```

Tras crear el directorio y entrar a él, cambiamos la configuración de los sitios disponibles en el servidor.

```
root@daniserver:/# cd /etc/apache2/sites-available/
```

Una vez dentro de los sitios disponibles, creamos el que será el fichero de configuración del sitio web.

```
root@daniserver:/etc/apache2/sites-available# nano PINT_pequenosnavegantes.conf
```

Una vez creado el fichero de la configuración, se procede a crear la configuración.

```
GNU nano 7.2                                         PINT_pequenosnavegantes.conf
<VirtualHost *:80>
    ServerName PINT_pequenosnavegantes
    DocumentRoot "/var/www/html/pequenosnavegantes"
    DirectoryIndex inicio_pequenosnavegantes.html
    #Si no hay directory index no mostrara el listado de archivos
    <Directory "/var/www/html/pequenosnavegantes">
        Options -Indexes
        AllowOverride All
    </Directory>
    #Autenticacion digest
    <Directory "/var/www/html/pequenosnavegantes">
        AuthType Digest
        AuthName "Zona Protegida"
        AuthUserFile "/var/www/html/pequenosnavegantes/.htdigest"
        Require valid-user
    </Directory>
    #Errores
    ErrorDocument 404 /errores/404.html
    ErrorDocument 403 /errores/403.html
    ErrorDocument 500 /errores/500.html
</VirtualHost>
```

Tras configurar el fichero, y guardarse, se procederá a comprobar su correcta creación.

```
root@daniserver:/etc/apache2/sites-available# ls
000-default.conf  asir.conf  daw.conf  default-ssl.conf  externa.conf  interna.conf  mapeo.conf  PINT_pequenosnavegantes.conf
```

Modelo relacional de la base de datos:

Tabla actividad:

Nombre de la columna	Tipo de dato	Restricciones
ID	INT(11)	PK, AI
Nombre	VARCHAR(100)	NOT NULL
Descripción	TEXT	NULL

Tabla admin:

Nombre de la columna	Tipo de dato	Restricciones
ID_admin	INT(11)	PK, AI
Email	VARCHAR(100)	UNIQUE, NOT NULL
Password	VARCHAR(255)	NOT NULL

Tabla asistencia:

Nombre de la columna	Tipo de dato	Restricciones
ID_asistencia	INT(11)	PK, AI
Fecha	DATE	NOT NULL
Asistió	TINYINT(1)	NOT NULL
ID_hijo	INT(11)	FK → Hijo(ID_hijo), NOT NULL
ID_grupo	INT(11)	FK → Grupo(ID_grupo), NOT NULL

Tabla grupo:

Nombre de la columna	Tipo de dato	Restricciones
ID_grupo	INT(11)	PK, AI
Nombre	VARCHAR(100)	NOT NULL
ID_monitor	INT(11)	FK → Monitor(ID_monitor), NOT NULL

Tabla campamento:

Nombre de la columna	Tipo de dato	Restricciones
ID_campamento	INT(11)	PK, AI
Nombre	VARCHAR(100)	NOT NULL
Fecha_inicio	DATE	NOT NULL
Fecha_fin	DATE	NOT NULL
Precio	DECIMAL(6,2)	NOT NULL
ID_programacion	INT(11)	FK → Programación_actividades(ID_programacion), NOT NULL
Municipio	TEXT	NOT NULL
Dirección	TEXT	NOT NULL
Código_postal	VARCHAR(5)	NOT NULL

Tabla comunicación:

Nombre de la columna	Tipo de dato	Restricciones
ID_comunicacion	INT(11)	PK, AI
Fecha	DATE	NOT NULL
Asunto	VARCHAR(100)	NOT NULL
Mensaje	TEXT	NOT NULL
ID_padre	INT(11)	FK → Padre(ID_padre), NOT NULL
ID_hijo	INT(11)	FK → Hijo(ID_hijo), NOT NULL
ID_monitor	INT(11)	FK → Monitor(ID_monitor), NOT NULL

Tabla inscripción:

Nombre de la columna	Tipo de dato	Restricciones
ID_inscripción	INT(11)	PK, AI
Fecha_inscripcion	DATE	NOT NULL
ID_hijo	INT(11)	FK → Hijo(ID_hijo), NOT NULL
ID_campamento	INT(11)	FK → Campamento(ID_campamento), NOT NULL
Estado_pago	ENUM	('pendiente', 'pagado', 'no pagado')

Tabla hijo:

Nombre de la columna	Tipo de dato	Restricciones
ID_hijo	INT(11)	PK, AI
Nombre	VARCHAR(100)	NOT NULL
Apellidos	VARCHAR(100)	NOT NULL
Fecha_nacimiento	DATE	NOT NULL
Medicación	TEXT	NULL
Alergias	TEXT	NULL
Datos_adicionales	TEXT	NULL
ID_padre	INT(11)	FK → Padre(ID_padre), NOT NULL
ID_grupo	INT(11)	FK → Grupo(ID_grupo), NOT NULL

Tabla monitor:

Nombre de la columna	Tipo de dato	Restricciones
ID_monitor	INT(11)	PK, AI
DNI	VARCHAR(9)	UNIQUE, NOT NULL
Nombre	VARCHAR(100)	NOT NULL
Apellidos	VARCHAR(100)	NOT NULL
Password	VARCHAR(255)	NOT NULL
Email	VARCHAR(100)	UNIQUE, NOT NULL
Teléfono	VARCHAR(16)	NOT NULL
Curriculum	TEXT	NULL

Tabla padre:

Nombre de la columna	Tipo de dato	Restricciones
ID_padre	INT(11)	PK, AI
DNI	VARCHAR(9)	UNIQUE, NOT NULL
Nombre	VARCHAR(100)	NOT NULL
Apellidos	VARCHAR(100)	NOT NULL
Password	VARCHAR(255)	NOT NULL
Email	VARCHAR(100)	UNIQUE, NOT NULL
Teléfono	VARCHAR(16)	NOT NULL
Token_login	VARCHAR(255)	NULL
Código_restablecer	VARCHAR(255)	NULL
Token_restablecer	VARCHAR(255)	NULL

Tabla programación\_actividades:

Nombre de la columna	Tipo de dato	Restricciones
ID_programacion	INT(11)	PK, AI
ID_actividad	INT(11)	FK → Actividad(ID), NOT NULL
ID_grupo	INT(11)	FK → Grupo(ID_grupo), NOT NULL
Hora_inicio	DATE	NOT NULL
Hora_fin	DATE	NOT NULL
Lugar	VARCHAR(255)	NOT NULL

Tabla solicitud\_monitor:

Nombre de la columna	Tipo de dato	Restricciones
ID_solicitud	INT(11)	PK, AI
Nombre	VARCHAR(100)	NOT NULL
Apellidos	VARCHAR(100)	NOT NULL
Email	VARCHAR(100)	UNIQUE, NOT NULL
Teléfono	VARCHAR(16)	NOT NULL
Curriculum	TEXT	NULL

Tabla tutor\_adicional:

Nombre de la columna	Tipo de dato	Restricciones
ID_tutor	INT(11)	PK, AI
DNI	VARCHAR(10)	UNIQUE, NOT NULL
Nombre	VARCHAR(100)	NOT NULL
Teléfono_autorizado	VARCHAR(16)	NOT NULL
ID_padre	INT(11)	FK → Padre(ID_padre), NOT NULL

Pruebas automáticas:

En este proyecto se realizaron pruebas manuales durante su desarrollo, así como pruebas automáticas adicionales. Las pruebas manuales permitieron detectar errores en tiempo real y ajustar aspectos funcionales durante la implementación. Por otro lado, las pruebas automáticas se ejecutaron para validar el rendimiento, la estabilidad y la correcta integración de los diferentes módulos del sistema. Gracias a este enfoque combinado, se logró mejorar la calidad de la aplicación web, asegurando su correcto funcionamiento en distintos escenarios.

Al realizar las pruebas automáticas, hubo un problema con phpunit, el cual no ejecutaba las pruebas debido a la falta de un fichero que no permitía la ejecución, para arreglarlo se buscó información en internet para ayudarse a encontrar el problema y la solución encontrada fue incluir los siguientes comandos:

- `sudo rm /tmp/mysql.sock`
- `sudo ln -s /Applications/XAMPP/xamppfiles/var/mysql/mysql.sock /tmp/mysql.sock`
- `ls -l /tmp/mysql.sock`
- `vendor/bin/phpunit tests`

una vez utilizado los comandos para arreglar phpunit, se procedió a ejecutar las pruebas. Un ejemplo de las pruebas realizadas fue a la conexión con la base de datos, al fichero conecta.php.

```
<?php  
  
use PHPUnit\Framework\TestCase;  
  
class ConectaTest extends TestCase  
{  
    private $conexion;  
    private $host = "localhost";  
    private $usuario = "root";  
    private $password = "";  
    private $dbname = "pequenosnavegantes";  
  
    /**
     * Configuración inicial antes de cada prueba.
```

```
 * Intenta establecer una conexión con MySQL.
 */
protected function setUp(): void
{
    // Intentar conectar a MySQL sin seleccionar la base de
    // datos todavía
    $this->conexion = new mysqli($this->host, $this-
    >usuario, $this->password);

    // Verificar si hay un error en la conexión
    if ($this->conexion->connect_error) {
        $this->fail("Error al conectar con MySQL: " . $this-
    >conexion->connect_error);
    }
}

/**
 * Prueba que verifica si la base de datos existe o la crea
 * si no está presente.
 */
public function testBaseDeDatosExisteOCrear()
{
    // Comprobar si la base de datos existe
    $sql = "SHOW DATABASES LIKE '{$this->dbname}'";
    $query = $this->conexion->query($sql);

    if ($query->num_rows <= 0) {
        // Si la base de datos no existe, intentamos crearla
        $sql = "CREATE DATABASE {$this->dbname}";
        $this->assertTrue(
            $this->conexion->query($sql),
            "Error al crear la base de datos: " . $this-
    >conexion->error
        );
    }

    // Verificar nuevamente que la base de datos se haya
    // creado correctamente
    $query = $this->conexion->query("SHOW DATABASES LIKE
    '{$this->dbname}'");
    $this->assertGreaterThanOrEqual(
        0,
        $query->num_rows,
        "La base de datos no fue creada correctamente"
    );
}

/**
```

```
* Prueba que verifica si la base de datos puede ser
seleccionada correctamente.
*/
public function testSeleccionarBaseDeDatos()
{
    $this->assertTrue(
        $this->conexion->select_db($this->dbname),
        "Error al seleccionar la base de datos: " . $this-
>conexion->error
    );
}
}
```