

PROYECTO DE FIN DE GRADO – CORÉ RUIZ DÍAZ

Tabletop Finder

1 - Descripción y objetivos generales del proyecto desarrollado	
- Objetivos del Proyecto	2
- Motivación	3
- Descripción de la Aplicación	4
2. Nivel empresarial	
- Introducción	6
- Análisis del entorno	7
- Solución propuesta	12
3 – Base de datos	
- Diseño de la base de datos	13
- Porqué MariaDB y un modelo de datos relacional	14
- Diagrama de clases	15
4 – Documentación de la aplicación	
- Introducción	16
- Tecnologías utilizadas	17
- Despliegue de la aplicación	20
5 - Manual del usuario	
- Registro e inicio de sesión	21
- Publicación y gestión de anuncios	23
- Búsqueda de partidas	25
- Sistema de chat	27
- Gestión de perfil	29
6 – Despliegue en internet	
- Creación de la instancia	33
- Asignación de una IP estática (elástica)	35
- Conexión a la máquina	36
7 – Mock-up	
- Decisiones de diseño	38
- Elementos interactivos	39
8 – Integración con Google y RRSS	
- Google Analytics	41
- Google ads	42
- Redes sociales	43
8 – Documentación del código	
- Estructura de datos	44
9 – Bibliografía y documentación	51

Descripción y objetivos generales del proyecto desarrollado

Objetivos del proyecto



El proyecto **Tabletop Finder** surge como una plataforma web pensada para la comunidad de jugadores de mesa en Almería, buscando cubrir la necesidad de un espacio centralizado donde se puedan reunir tanto aficionados experimentados como novatos. El objetivo principal es **facilitar la comunicación** y la **organización de partidas**, ofreciendo un lugar donde anunciar nuevas sesiones de juego, buscar compañeros de mesa y **promocionar eventos y torneos** de las tiendas locales o de cualquier otra entidad interesada en la difusión de los juegos de mesa.

Hoy en día, cada vez más personas se inician en el mundo de los juegos de mesa, ya sea por curiosidad, recomendación de amigos o por el atractivo que supone la gran diversidad de títulos y mecánicas disponibles. Sin embargo, muchos se encuentran con dificultades a la hora de **encontrar grupos** de juego adecuados o eventos en los que participar. Con Tabletop Finder, se pretende **romper esta barrera** de acceso, ofreciendo un portal intuitivo y de uso sencillo para que los jugadores puedan descubrir partidas activas en Almería, informarse de torneos y competiciones, así como conectarse con otros usuarios de manera rápida y eficaz.

Otro de los objetivos esenciales es **fortalecer la comunidad** de jugadores locales en Almería, dando visibilidad a las tiendas y locales que organizan o acogen eventos. Al contar con una plataforma que centraliza la información de diferentes actividades, estas tiendas pueden promocionarse de forma

más efectiva, y los jugadores pueden mantenerse al tanto de las novedades. Además, el uso de **herramientas digitales** facilita la creación de un registro detallado de los participantes, la comunicación de fechas y lugares, e incluso la gestión de las inscripciones.

Finalmente, **Tabletop Finder** aspira a ser un **sistema escalable** que pueda adaptarse a las necesidades de distintas comunidades. En un futuro, se podrían incorporar nuevas funcionalidades que permitan reseñar juegos, compartir experiencias o incluso incluir módulos para otro tipo de actividades relacionadas con el ocio alternativo (como rol en vivo, wargames o juegos de cartas coleccionables). Con ello, se busca que la plataforma sea un punto de referencia para todos aquellos que quieran iniciarse o profundizar en el apasionante mundo de los juegos de mesa, fomentando la interacción y el crecimiento de la escena lúdica almeriense.

Motivación



Mi motivación principal a la hora de hacer Tabletop Finder es el hecho de que he sido miembro de la comunidad de juegos de mesa durante varios años, sea tanto de juegos de mesa clásicos como de juegos de cartas en varios ámbitos, y a pesar de que ya no soy tan activo, soy consciente de que la comunidad de juegos de mesa, a pesar de ser grande, tiene un crecimiento muy pequeño.



Incluso con herramientas como Tabletop Simulator, que permite que se puedan jugar juegos de mesa por internet con gente de otros lugares, sigue habiendo una barrera a la hora de encontrar grupos o gente para poder jugar.

Tabletop Finder nace como una idea de poder reforzar la escena de juegos de mesa en Almería, tanto poniendo en contacto a la gente, como ayudando a las tiendas a promocionar sus locales, partidas y torneos

Descripción de la aplicación



La aplicación está diseñada para facilitar la búsqueda, creación y gestión de partidas de juegos de mesa, así como para fomentar la interacción entre sus usuarios. A continuación, se detallan las funcionalidades principales:

Publicación de anuncios

- Los usuarios pueden **crear anuncios** en los que detallen la información de la partida que desean organizar, indicando el título, descripción, tipo de juego (online o físico), número de jugadores, y juego a jugar.
- La interfaz de publicación se ha diseñado para que sea **intuitiva** y permita a los organizadores describir claramente la experiencia que ofrecen.

Inscripción a partidas

- Cualquier usuario interesado en un anuncio puede **apuntarse** de forma sencilla, indicando su interés y confirmando su asistencia.
- El sistema de inscripción **notifica** al creador de la partida acerca de las nuevas peticiones pendientes, facilitando así la coordinación y el control de los cupos disponibles.

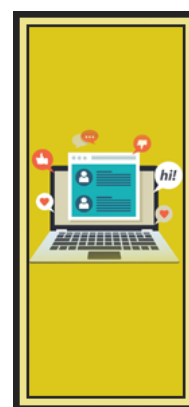


Búsqueda por tipo de partida (online o física)

- Tabletop Finder ofrece un **filtro** que distingue entre partidas que se llevarán a cabo de manera presencial (en tiendas, locales o domicilios particulares) y partidas en línea (vía plataformas virtuales de juegos de mesa o servicios de videoconferencia).
- De esta manera, cada usuario puede encontrar rápidamente **partidas que se ajusten a sus preferencias**, ya sea por comodidad, cercanía geográfica o disponibilidad de medios digitales.

Sistema de chat integrado

- Con el fin de fomentar la comunicación y la **coordinación en tiempo real**, se ha incorporado un sistema de mensajería interno que permite a los usuarios **chatear** entre ellos, avisando mediante el sistema de notificaciones cuando un usuario tiene un mensaje nuevo.
- Este chat puede ser utilizado tanto para **resolver dudas** sobre las partidas como para conversar de manera informal y así **fortalecer la comunidad**.



Nivel empresarial

Introducción



En este apartado se ofrece una visión global del proyecto **Tabletop Finder**, resaltando su finalidad, objetivos, requisitos y restricciones, así como la propuesta de valor y el público al que se dirige.

Breve descripción:

Tabletop Finder es una aplicación web diseñada para facilitar la organización y difusión de partidas, torneos y eventos de juegos de mesa en Almería. La plataforma permitirá a los usuarios publicar anuncios, inscribirse en partidas, filtrar búsquedas según el tipo de encuentro (online o presencial) y comunicarse mediante un sistema de chat integrado.

Propuesta de valor:

La aplicación se presenta como una herramienta centralizada que optimiza la comunicación y coordinación entre jugadores, organizadores de eventos y locales especializados. Su facilidad de uso y accesibilidad digital eliminan barreras tradicionales, permitiendo que tanto aficionados novatos como experimentados encuentren y organicen actividades lúdicas de forma rápida y eficaz.

Necesidad que satisface:

Tabletop Finder responde a la demanda de espacios digitales que unifiquen la información sobre eventos y partidas de juegos de mesa. Facilita la búsqueda de grupos y partidas, reduce la carga administrativa para los organizadores y fomenta el crecimiento de una comunidad local activa y comprometida.

Target:

El público objetivo incluye:

- Jugadores de mesa de todas las edades y niveles.
- Organizadores de partidas y eventos.
- Locales y tiendas especializadas en juegos de mesa.
- Personas interesadas en participar en actividades de ocio y entretenimiento lúdico en la región de Almería.

Análisis del entorno



Este apartado analiza el entorno en el que se desarrollará el proyecto, teniendo en cuenta tanto factores generales (macroentorno) como aspectos más específicos del sector (microentorno).

Macroentorno (PEST)

Político

Normativas y regulaciones: En el mundo digital y en la organización de eventos de ocio, es fundamental tener en cuenta las normativas que se aplican a nivel local, regional y nacional. Por ejemplo, la ley de protección de datos (como el RGPD en Europa) establece cómo se deben manejar los datos personales de los usuarios. Además, las reglas sobre la organización de eventos y actividades de ocio, que incluyen temas de seguridad y accesibilidad, afectan directamente a cómo funcionará la aplicación.

Políticas de apoyo a la innovación y digitalización: Tanto el gobierno central como las autoridades regionales, especialmente en Andalucía, han impulsado iniciativas para fomentar la digitalización y la innovación. Estas políticas se traducen en ayudas, incentivos fiscales y facilidades de financiación, que pueden ser muy útiles para poner en marcha y hacer crecer Tabletop Finder.

Económico

Situación económica en Almería: La economía de Almería sigue basándose en sectores tradicionales como la agricultura y el turismo, pero también hay un creciente interés por diversificar y apostar por la innovación y el emprendimiento digital. Esto abre la puerta a nuevos modelos de negocio, especialmente en el ámbito del ocio y el entretenimiento.

Gasto en ocio y capacidad adquisitiva: Aunque el nivel de ingresos varía en la provincia, se observa que la gente sigue destinando una parte de su presupuesto a actividades de ocio. La demanda de experiencias, como los juegos de mesa y eventos relacionados, es alta. Esto significa que, con una propuesta accesible y bien posicionada, la aplicación puede captar a un público dispuesto a invertir en momentos de socialización y entretenimiento.

Social

Tendencias en el ocio: Los juegos de mesa nunca han abandonado la popularidad a lo largo de los años, aunque siguen siendo un hobby de nicho. Esto ha generado una comunidad en crecimiento, formada tanto por jugadores experimentados como por principiantes, que buscan espacios para organizar partidas y compartir su pasión.

Aceptación de plataformas digitales: La digitalización ha cambiado la forma en que las personas organizan y participan en actividades de ocio. Cada vez es más común confiar en plataformas online para coordinar encuentros, inscribirse en eventos y estar al día de las novedades. Esta tendencia favorece a Tabletop Finder, que ofrece una solución completa y fácil de usar.

Cambio en los hábitos de consumo de ocio: Hoy en día, la gente busca experiencias personalizadas y valora las aplicaciones que facilitan la organización de eventos. Funciones como publicar anuncios, filtrar búsquedas y chatear en tiempo real responden a esta necesidad de tener acceso rápido y sencillo a la información y a la interacción social.

Tecnológico

Infraestructura tecnológica disponible: Almería cuenta con una infraestructura tecnológica en constante mejora, con buen acceso a internet de alta velocidad y una amplia adopción de dispositivos móviles.

Esto crea un ambiente favorable para desarrollar una aplicación web robusta y accesible para la mayoría de los usuarios.

Acceso a Internet y uso de dispositivos: La disponibilidad de conexiones rápidas y el uso generalizado de smartphones aseguran que los usuarios podrán utilizar la plataforma sin problemas técnicos, lo que facilita la interacción y participación en los eventos.

Adopción de herramientas digitales en el entretenimiento: La transformación digital ha llevado a que cada vez más personas utilicen aplicaciones para organizar sus actividades de ocio. Los usuarios buscan herramientas que les permitan inscribirse en eventos, filtrar partidas (ya sean online o presenciales) y comunicarse en tiempo real, características en las que Tabletop Finder encaja perfectamente.

Microentorno

Además del análisis general, se estudia el microentorno, centrado en la competencia local y regional, las posibles colaboraciones y los hábitos de consumo específicos del sector. Esto ayudará a identificar oportunidades y desafíos concretos para posicionar la aplicación de la mejor manera en el mercado.

Esta evaluación completa del entorno ofrece una visión clara de cómo se pueden aprovechar las oportunidades y enfrentar los retos que presenta el mercado, asegurando que Tabletop Finder se desarrolle en un contexto favorable tanto a nivel macro como en el ámbito específico de los juegos de mesa y eventos de ocio.

Matriz DAFO y CAME

DAFO:

- *Fortalezas:* Interfaz intuitiva, enfoque local y adaptación a las necesidades de la comunidad, integración de funcionalidades (publicación, inscripción, chat).
- *Debilidades:* Recursos iniciales limitados, dependencia del crecimiento orgánico de la comunidad, posibles barreras tecnológicas en usuarios menos habituados a herramientas digitales.

- *Oportunidades:* Auge de la cultura de los juegos de mesa, incentivos a la digitalización y ayudas para emprendimientos, expansión a nuevos nichos (por ejemplo, rol, wargames).
- *Amenazas:* Competencia de plataformas similares, cambios normativos y fluctuaciones en el gasto en ocio.

CAME:

Se establecen estrategias para potenciar las fortalezas y oportunidades (por ejemplo, alianzas con tiendas y organizadores de eventos), mientras se mitigan las debilidades y amenazas (capacitación a usuarios, búsqueda de financiamiento y apoyo institucional).

Oportunidades de negocio:

El estudio identifica nichos de mercado poco explotados en la región, como la organización de eventos de juegos de mesa y torneos en locales especializados. Además, se destacan oportunidades para la creación de empleo en el sector digital y de ocio, respaldando la viabilidad del proyecto. La sinergia entre las tendencias del mercado y la propuesta de Tabletop Finder refuerza su idoneidad y potencial éxito.

Obligaciones fiscales, laborales y de prevención de riesgos

Fiscales:

Para garantizar el cumplimiento normativo, es esencial definir la forma jurídica adecuada para el proyecto. Nuestra opción sería constituir una sociedad limitada, lo que facilita la gestión y protege el patrimonio personal de los socios. Una vez establecida la entidad, se procederá al alta en la Agencia Tributaria, asegurando la inscripción en los registros correspondientes y la activación del sistema fiscal de la empresa. Esto implica el cumplimiento de las obligaciones tributarias, entre las que se destacan el pago del IVA, el Impuesto de Sociedades y otros tributos pertinentes conforme al calendario fiscal anual. La correcta gestión fiscal no solo es un requisito legal, sino que también contribuye a la transparencia y sostenibilidad financiera del proyecto.

Laborales:

Desde el punto de vista laboral, es obligatorio realizar el alta en la Tesorería

General de la Seguridad Social, lo que permite formalizar la contratación de personal. Esto incluye la elaboración de contratos laborales que se ajusten al grupo profesional y a las funciones que desempeñarán los empleados dentro de la empresa. La definición clara de roles y responsabilidades es fundamental para garantizar un ambiente de trabajo estructurado y eficiente, y para cumplir con la normativa vigente en materia laboral. De esta manera, se asegura que el personal contratado cuente con los derechos y garantías establecidos por la ley, contribuyendo a la estabilidad y crecimiento del equipo humano.

Prevención de riesgos:

Si el proyecto contase con un espacio físico, se deberían cumplir las normativas específicas en materia de prevención de riesgos laborales. Sin embargo, Tabletop Finder, al menos según lo planeado, no tendría uno.

Posibles ayudas o subvenciones

Para impulsar el crecimiento y la expansión del proyecto, vamos a investigar en detalle las ayudas y subvenciones que podemos aprovechar. Este estudio se dividirá en distintos niveles:

Local y Regional: Buscaremos programas de apoyo a la innovación y digitalización que ofrecen tanto las entidades locales como las autonómicas. Estas ayudas están pensadas para fomentar el emprendimiento y modernizar la tecnología, y suelen incluir recursos económicos y asesoramiento especializado.



Nacional: A nivel nacional existen subvenciones y programas específicos para startups y proyectos tecnológicos, impulsados por organismos gubernamentales y ministerios. Estos programas pueden ofrecer incentivos fiscales, financiación a bajo interés o incluso ayudas directas para poner en marcha y expandir la empresa.

Europeo: También contaremos con la opción de acceder a fondos y subvenciones de la Unión Europea, especialmente a través de programas que apoyan la innovación, la competitividad y la transformación digital en pequeñas y medianas empresas. Estos fondos pueden facilitar mucho la inversión inicial y el desarrollo de nuevas funcionalidades en la aplicación.



La combinación de todas estas ayudas será clave para que el proyecto sea financieramente viable, permitiendo cubrir los gastos iniciales y también impulsar su crecimiento y adaptarse a los cambios del mercado. Conseguir financiación externa, junto con una buena planificación interna, ayudará a consolidar Tabletop Finder como una iniciativa innovadora y sostenible en el sector del ocio y el entretenimiento.

Solución propuesta



Esta sección explica de forma sencilla la estrategia y la estructura que necesitamos para poner en marcha y hacer funcionar Tabletop Finder de manera exitosa.

Estudio económico y presupuestos: Primero, se hará un análisis detallado de la inversión inicial, identificando los costos fijos (como infraestructura, mantenimiento, licencias, etc.) y los costos variables (como marketing, actualizaciones y la expansión de funcionalidades). Además, se comparará la situación con la competencia para definir una política de precios que sea competitiva y sostenible.

Recursos humanos: Se definirán los perfiles profesionales necesarios para el desarrollo y la gestión de la aplicación, por ejemplo, desarrolladores, diseñadores, gestores de comunidad y personal de soporte. También se creará un organigrama que muestre los diferentes grupos de trabajo, las funciones específicas y las competencias que se requieren en cada puesto.

Recursos materiales: Se identificarán los recursos tecnológicos y materiales esenciales, como servidores, herramientas de desarrollo y software de gestión, que aseguren que el proyecto funcione correctamente y pueda crecer con el tiempo.

Base de datos

Diseño de la base de datos



Para Tabletop Finder se ha optado por un modelo de base de datos relacional, lo que significa que la información se organiza en tablas conectadas entre sí. Se utiliza MariaDB (muy similar a MySQL) y se gestiona a través de phpMyAdmin, una herramienta que facilita mucho la administración de la base de datos.

La estructura está compuesta por varias tablas, cada una con su función:

- **Usuarios:** Guarda los datos de cada usuario, como nombre, email y otros detalles personales. En esta tabla se guarda información como si el usuario es tienda o administrador, su descripción, o que avatar está usando.
- **Anuncio:** Contiene la información de los anuncios o partidas creadas, relacionándose con los usuarios, con plazasjuegos y con juegos mediante claves.
- **Fondos:** Almacena datos sobre imágenes o fondos que se usan en la aplicación, específicamente su nombre y ruta, para que luego al buscar los fondos de usuarios y anuncios, se obtengan de aquí.
- **Juegos:** Incluye información sobre los juegos disponibles, como nombre, tipo y descripción. Todos estos campos se recuperan cada vez que se tiene que mostrar el juego.
- **Notificaciones:** Registra las alertas y mensajes que se envían a los usuarios, sea notificaciones o mensajes de chat.
- **Plazasjuegos:** Es una tabla de unión que relaciona los anuncios con los usuarios que se apuntan a ellos. Esta se usa para sacar cuantas

Las tablas se conectan mediante claves primarias y foráneas, lo que ayuda a mantener la información consistente y bien organizada. Además, se ha elaborado un diagrama ER (Entidad-Relación) que muestra visualmente cómo se interconectan estas tablas, facilitando la comprensión de la estructura y el flujo de datos dentro de la aplicación.

Este modelo relacional es una opción madura y robusta, ideal para manejar datos estructurados como los de Tabletop Finder, y permite una buena escalabilidad y facilidad para futuras actualizaciones del sistema.

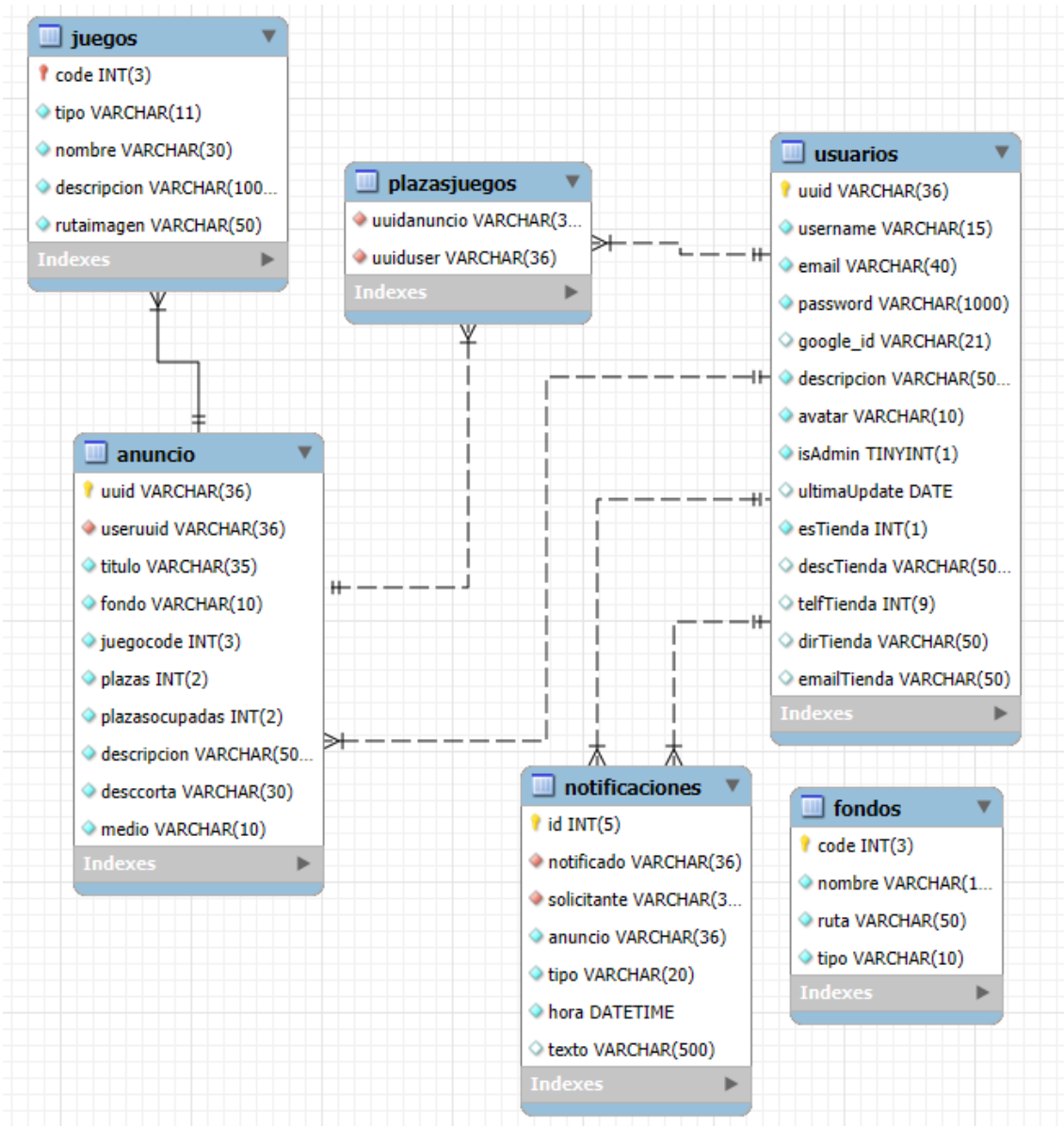
Porqué MariaDB y un modelo de datos relacional

Elegimos usar MariaDB y un modelo de base de datos relacional por varias razones que se adaptan muy bien a las necesidades de nuestro proyecto:

- **Estructura organizada:** El modelo relacional nos permite estructurar los datos en tablas interconectadas, lo que resulta ideal para manejar información clara y definida, como la de usuarios, anuncios, juegos, etc. Esto garantiza que las relaciones entre los datos sean coherentes y fáciles de mantener.
- **Solución robusta y de código abierto:** MariaDB es una opción robusta y confiable que se basa en MySQL. Al ser de código abierto, cuenta con una gran comunidad de desarrolladores y usuarios, lo que significa que siempre habrá soporte, actualizaciones y una amplia documentación disponible.
- **Integridad y consistencia de la información:** Gracias al uso de claves primarias y foráneas, el modelo relacional permite mantener la integridad de los datos, asegurando que la información esté siempre actualizada y sea consistente, algo fundamental cuando se trata de datos de usuarios y transacciones.

En resumen, optamos por MariaDB y el modelo relacional porque ofrecen una solución madura, bien documentada y perfectamente alineada con las necesidades de una aplicación como Tabletop Finder. Pero sobre todo, el principal motivo por el que elegimos el modelo relacional es porque ya tenemos experiencia en ello.

Diagrama de clases



Documentación de la aplicación

Introducción



Tabletop Finder es un proyecto desarrollado usando Laravel, PHP y JavaScript, en el que hemos aprovechado tecnologías como AJAX y el manejo del DOM para que la web sea lo más interactiva posible. La idea es crear un espacio donde los jugadores puedan encontrar partidas y eventos de forma sencilla y en tiempo real.

Para los usuarios, la aplicación permite:

- **Registrarse e iniciar sesión:** Crear tu cuenta y entrar de forma segura.
- **Crear partidas:** Organizar eventos y partidas según tu disponibilidad.
- **Apuntarse a partidas:** Unirte a las partidas que te interesen.
- **Gestionar solicitudes:** Los organizadores pueden aceptar o denegar a quienes quieran unirse a sus partidas.
- **Editar tu perfil:** Actualizar tu información personal y configurar tu cuenta a tu gusto.
- **Chatear con otros usuarios:** Usar el chat integrado para comunicarte en tiempo real.
- **Ver la lista de juegos:** Consultar y descubrir los juegos disponibles en la plataforma.

Además, los administradores tienen algunas funciones extras, como:

- **Eliminar usuarios:** Poder borrar cuentas cuando sea necesario.
- **Convertir usuarios en “tienda”:** Cambiar el perfil de algunos usuarios para que funcionen como tiendas o locales.
- **Registrar nuevos juegos:** Añadir nuevos juegos a la lista disponible.
- **Subir nuevos fondos y avatares:** Actualizar los recursos gráficos que pueden usar los usuarios en sus perfiles.

Con esta documentación queremos que te sea fácil entender cómo funciona Tabletop Finder, tanto si eres un usuario que quiere empezar a jugar, como si eres desarrollador y necesitas conocer el detalle de la aplicación.

Tecnologías utilizadas



Front-end

HTML5

CSS3

JavaScript.



(No se ha usado ningún framework adicional específico, aunque se podría integrar Bootstrap u otra librería en el futuro.)

Back-end

Laravel



PHP



Base de Datos

Sistema de gestión (MariaDB) administrada mediante phpMyAdmin.



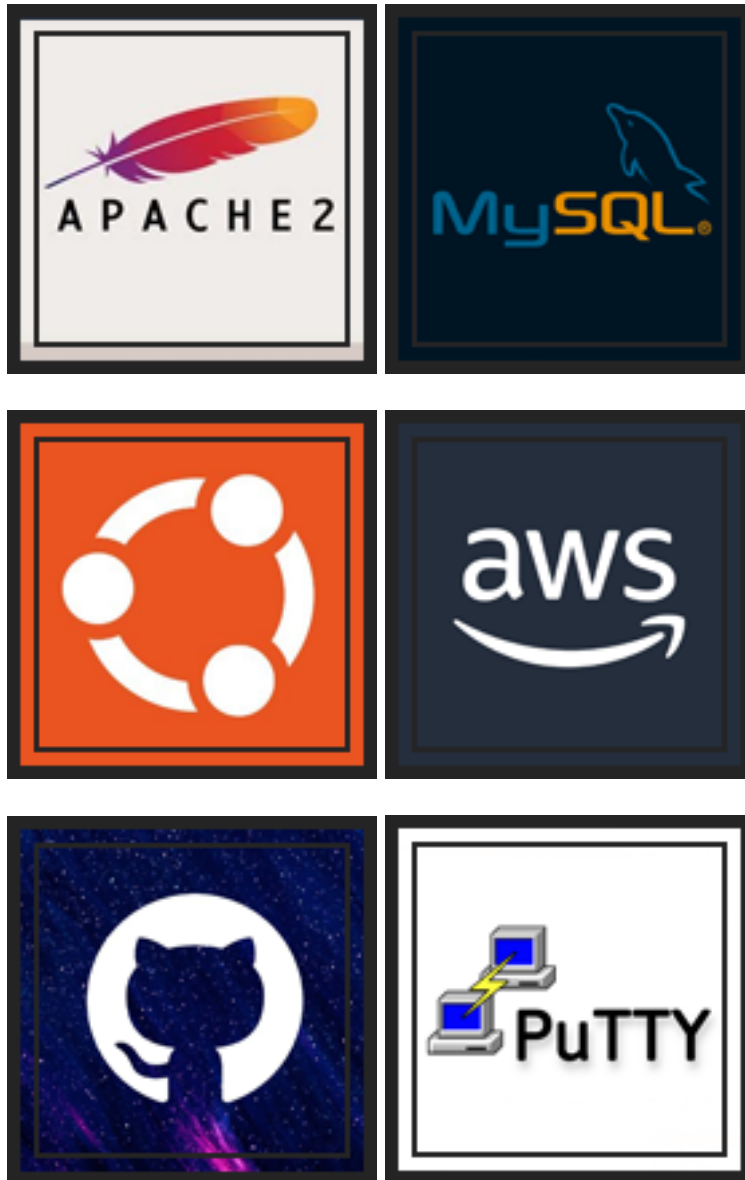
Desarrollo

Todas estas tecnologías se han estado desarrollando y usando a través de XAMPP con Apache2 y Visual Studio Code antes de ser subidas a la web



Otras Herramientas

Github y Amazon Web Services para el despliegue, con un servidor Ubuntu Server que utiliza MySQL para la base de datos y Apache2 para servir la web, y PuTTY para conectarnos al servidor



Despliegue de la aplicación



Infraestructura

Servidor y hosting: Hemos optado por Amazon Web Services (AWS) para alojar Tabletop Finder. Utilizamos un servidor Linux en el que hemos instalado MySQL para la base de datos y Apache2 para servir la web.

Configuración del entorno de producción: El entorno está configurado para funcionar de manera estable y escalable, con todas las herramientas necesarias para un rendimiento óptimo y una fácil gestión del sistema.

Proceso de despliegue:

Subida de la aplicación: La aplicación se ha desplegado en el servidor utilizando Git. Esto nos ha permitido llevar un control de versiones preciso y facilitar la integración continua de los cambios realizados durante el desarrollo.

Migración de la base de datos: Para migrar la base de datos, hemos exportado nuestro esquema y datos a un archivo a través de phpMyAdmin. Posteriormente, utilizando PuTTY, se ha copiado y cargado ese archivo en el servidor, completando así el proceso de migración de la base de datos.

Optimización y seguridad

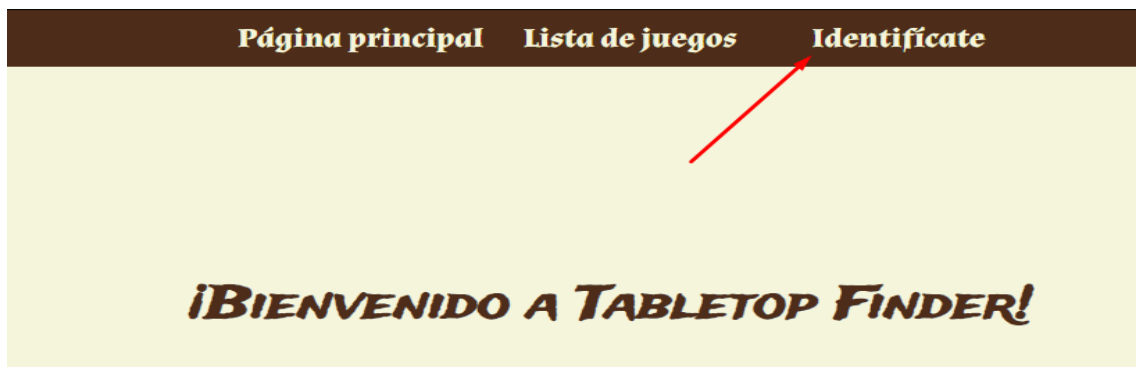
Medidas de seguridad: Se ha configurado un certificado SSL para asegurar todas las conexiones, y el servidor cuenta con un firewall que protege la infraestructura de posibles ataques.

Optimización del rendimiento: Se han realizado ajustes en la configuración del servidor y de la aplicación para asegurar una respuesta rápida y un funcionamiento fluido, mejorando así la experiencia del usuario final.

Manual del usuario

Registro e inicio de sesión

Cuando accedemos a la página por primera vez, nos encontramos en el menú la opción de “Identifícate”.



Al acceder, tenemos dos opciones: Login, y Nuevo usuario. Haremos click en “Nuevo usuario” para registrarnos.



Esto nos lleva al formulario de registro, el cual nos da instrucciones básicas para el registro.

[Página principal](#) [Lista de juegos](#) [Identifícate](#)

REGISTRO

Usuario:
Máximo 15 caracteres

Email:

Contraseña:
Debe contener 8 caracteres, una mayúscula, minúscula y número

Repita la contraseña:

Registrarse

Deberemos introducir nuestro nombre de usuario, un email con formato válido, y nuestra contraseña con las opciones indicadas.

Tras hacer el registro, nos llevará a la pantalla de login, indicándonos que el usuario se ha creado, y podremos logear usando nuestras credenciales.



Página principal **Lista de juegos** **Identificate**

LOGIN

El usuario se ha creado correctamente.

Usuario:

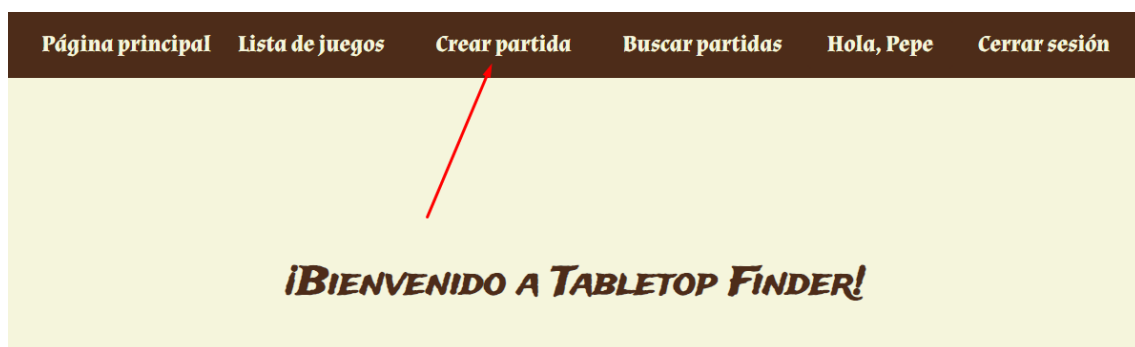
Contraseña:

Login

¿Nuevo usuario?

Publicación y gestión de anuncios

Una vez logeados, tendremos la opción de crear una partida en el menú.



Página principal **Lista de juegos** **Crear partida** **Buscar partidas** **Hola, Pepe** **Cerrar sesión**

¡BIENVENIDO A TABLETOP FINDER!

Al hacer click nos llevará al formulario de registro de la partida. Aquí podremos poner el anuncio, las plazas, descripciones, medio de la partida, fondo que usará el anuncio, y a que juego se jugará.

Crear partida

Título del anuncio <small>Máximo 35 caracteres</small> <input type="text"/>	Plazas <small>Máximo 99 plazas</small> <input type="text"/>	Vista Previa del Fondo 
Descripción <small>Esta es la descripción que aparecerá cuando los jugadores abran el anuncio. Máximo 500 caracteres</small> <input type="text"/>	Descripción corta <small>La descripción inicial para llamar la atención del usuario en el buscador. Máximo 30 caracteres</small> <input type="text"/>	
Fondo del anuncio <small>Fondo que aparecerá al abrir el anuncio. Debes seleccionar uno para crear el anuncio</small> <input type="text" value="-"/>	Medio de la partida <small>El medio por el que se jugará tu partida</small> <input type="text" value="Online"/>	
Juego <small>Puedes ver el listado de juegos disponibles aquí</small> <input type="text" value="-"/>		
<input type="button" value="Guardar"/>		

Seleccionando un fondo, podremos ver su vista previa.

Título del anuncio <small>Máximo 35 caracteres</small> <input type="text"/>	Plazas <small>Máximo 99 plazas</small> <input type="text"/>	Vista Previa del Fondo 
Descripción <small>Esta es la descripción que aparecerá cuando los jugadores abran el anuncio. Máximo 500 caracteres</small> <input type="text"/>	Descripción corta <small>La descripción inicial para llamar la atención del usuario en el buscador. Máximo 30 caracteres</small> <input type="text"/>	
Fondo del anuncio <small>Fondo que aparecerá al abrir el anuncio. Debes seleccionar uno para crear el anuncio</small> <input type="text" value="nevado"/>	Medio de la partida <small>El medio por el que se jugará tu partida</small> <input type="text" value="Online"/>	

Una vez que le demos a crear, y los datos sean válidos, la partida se creará y la página nos llevará al perfil de la misma.



Búsqueda de partidas

Podemos localizar el buscador de partidas en el menú.



Al hacer click, nos llevará a la página para buscar partidas.



En esta podremos o bien usar los filtros para buscar partidas por juego o medio, o simplemente mirar las partidas que se nos ofrecen.

Buscar Partidas

Juego

Not Alone
▼

Tipo de partida:

☒ Online
☐ Física

Buscar

Una vez hemos encontrado una partida que creemos que nos conviene, podemos entrar en ella haciendo click en su nombre.

Título	↑	Creador	Descripción	Juego	Plazas
Not Alone por las tardes		Marta	Me gustaría jugar Not Alone con un grupo por la zona del zapillo los martes por las tardes, de 5 a 9	Not Alone	3 / 5

Una vez dentro, podemos solicitar unirnos a través del botón “Quiero unirme”

Not Alone por las tardes

Creador de la partida

Marta

Quiero unirme

NOT ALONE POR LAS TARDES

Medio: online

Juego: Not Alone

Descripción: Me gustaría jugar Not Alone con un grupo por la zona del zapillo los martes por las tardes, de 5 a 9

Plazas: 3 / 5

Sistema de chat

Para acceder al sistema de chat, tan solo tenemos que acceder al perfil de un jugador; típicamente, a través de un anuncio.



Al entrar en su perfil, podemos darle al botón “Ver chat” para iniciar un chat con esa persona.



Cuando le demos por primera vez, el chat estará vacío.



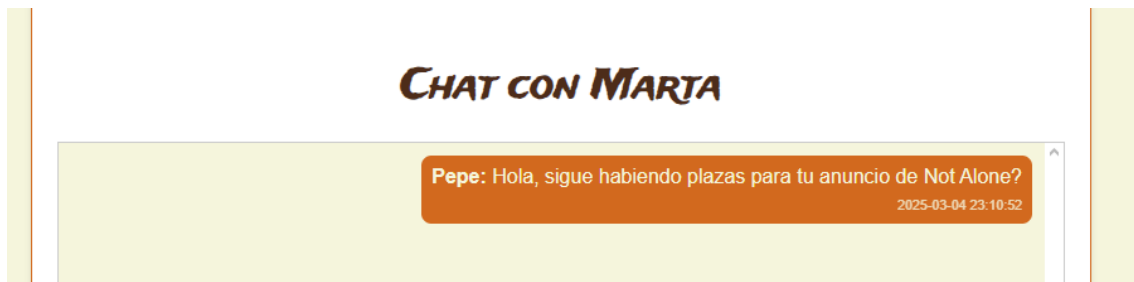
CHAT CON MARTA

Enviar mensaje a Marta

Mensaje

Enviar

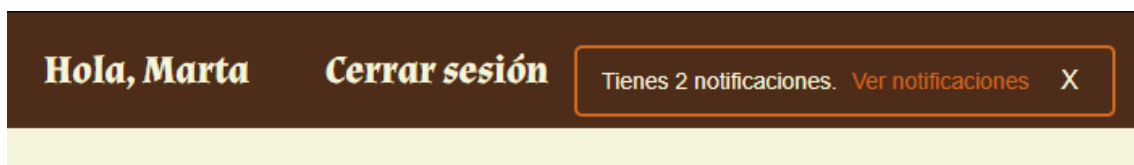
Podemos escribir un mensaje en la caja y darle al botón de Enviar para que se mande.



CHAT CON MARTA

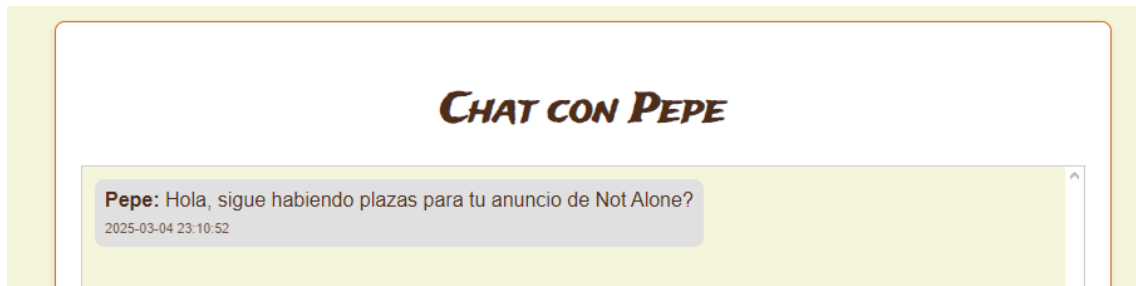
Pepe: Hola, sigue habiendo plazas para tu anuncio de Not Alone?
2025-03-04 23:10:52

Cuando tengamos mensajes nuevos, se nos notificará en el menú.



Hola, Marta **Cerrar sesión** Tienes 2 notificaciones. [Ver notificaciones](#) X

La notificación se borrará tanto si le damos al botón de “borrar notificación” como si accedemos al chat con esa persona.

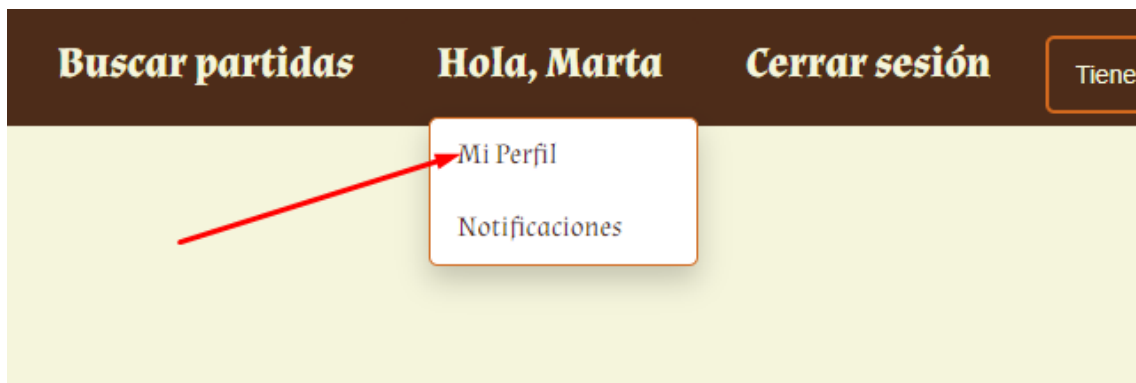


Donde podremos contestar.



Gestión de perfil

Una vez estemos logeados, podremos acceder a nuestro perfil haciendo click en el botón de saludo en el menú, y después, en “Mi perfil”.



Al acceder podremos ver tanto nuestros datos de perfil



Editar avatar

Perfil personal de Marta

Nombre: Marta

Email: marta@marta.com

Descripción: ¡Hola! Soy Marta, 23 años, y me encanta jugar juegos de mesa en mis ratos libres

Las partidas en las que participamos, y los anuncios que hemos creado.

Partidas en las que participas

Título	Descripción	Juego	Plazas
Este usuario no ha publicado anuncios.			

Tus anuncios

Título	Descripción	Juego	Plazas
Not Alone por las tardes	Me gustaría jugar Not Alone con un grupo por la zona del zapillo los martes por las tardes, de 5 a 9	Not Alone	3 / 5

Y que chats tenemos activos con otros usuarios.

Chats Activos

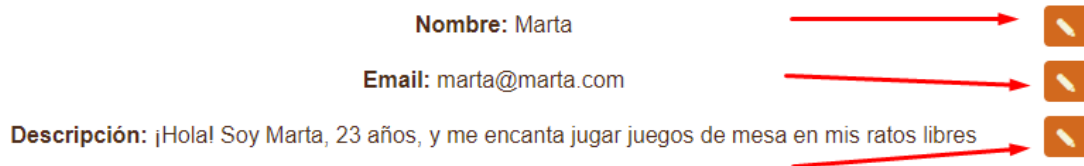


MasterMagic



Pepe

Haciendo click en los distintos botones del perfil, podremos editar nuestro nombre, email y descripción.



Todos ellos nos llevarán a una página donde podremos actualizar estos campos.

EDITAR USUARIO

Nombre

Actualizar Usuario

EDITAR EMAIL

Email

Actualizar Usuario

EDITAR USUARIO

Nombre

Actualizar Usuario

Adicionalmente, al hacer click en el botón de Editar Avatar podremos seleccionar un avatar nuevo de la selección de avatares que la página ofrece.



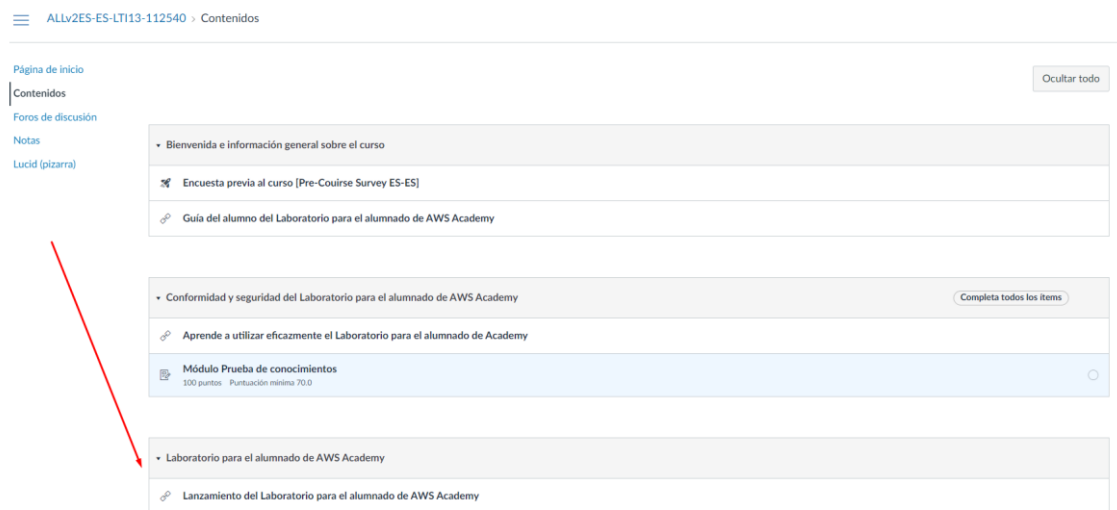
Despliegue en internet

Creación de la instancia

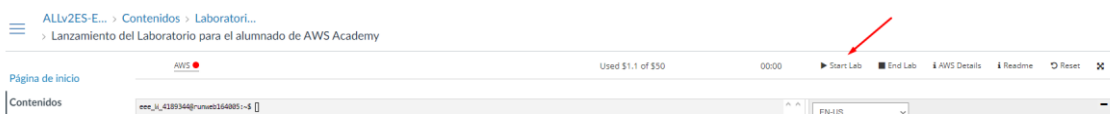
Para desplegar el servidor, hemos elegido Amazon Web Service, para el cual el instituto ha sido tan amable de apuntarnos a un classroom que nos regala 50 euros de AWS para hacer el despliegue.



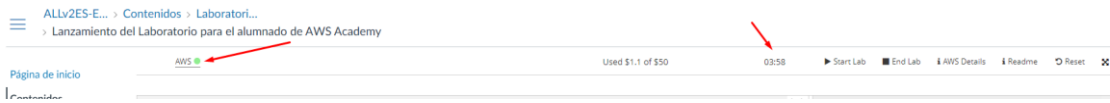
Para ello, nos hemos venido a esta opción en nuestros Contenidos:



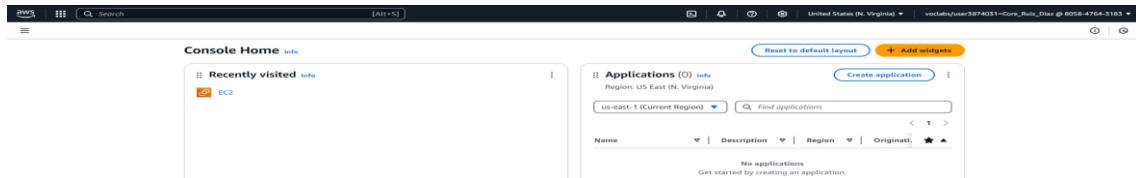
En ella, tenemos acceso a nuestra consola, donde podemos iniciar una sesión “Lab” de cuatro horas



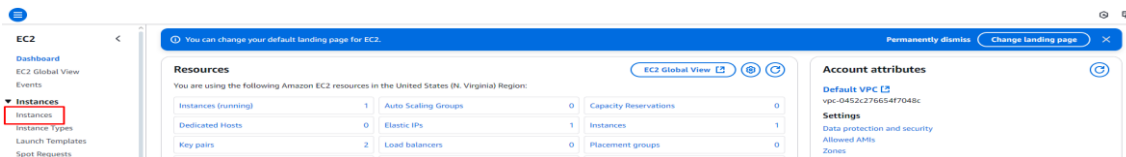
Una vez que el icono en AWS está verde, y el tiempo corriendo, nuestra sesión está activa.



Hacemos click en AWS, y nos llevará a una nueva panina; nuestra Home de la consola.



En esta, buscamos el EC2 y entramos en el, lo cual nos da acceso a nuestras instancias.



Una vez aquí, hemos creado una instancia nueva para poder alojar nuestro servidor; en nuestro caso creamos una máquina de Ubuntu server con las siguientes características (las fotos fueron tomadas post creación)

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

Sevidor Tabletop Finder

[Add additional tags](#)

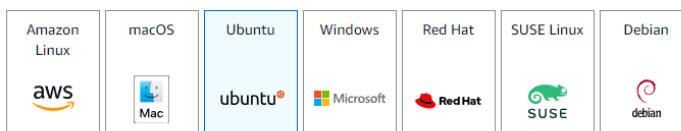
▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents

Quick Start



[Browse more AMIs](#)
Including AMIs from
AWS, Marketplace and
the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-04b4f1a9cf54c11d0 (64-bit (x86)) / ami-0a7a4e87939439934 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

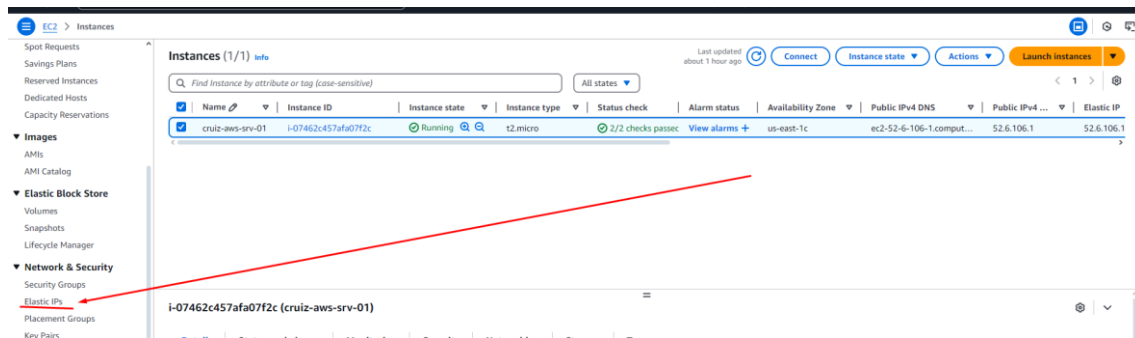
Description

Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

Asignación de una IP estática (elástica)

Una vez hemos creado la instancia, le vamos a asignar una elastic ip; se trata de una IP que nos asigna Amazon que es permanente, de manera que la máquina no tiene una IP nueva cada vez que la lanzamos. Esto es necesario tanto para que nuestro servidor de apache se mantenga online, como para poder conectarnos a la ip de la máquina.



Y le asignamos una elastic ip a nuestra instancia:



(Las capturas fueron tomadas tras hacer la IP, pero normalmente nos dejaría seleccionar a que instancia queremos asignarle la ip)

Allocate Elastic IP address Info

Elastic IP address settings Info

Public IPv4 address pool

- ☒ Amazon's pool of IPv4 addresses
- ☐ Public IPv4 address that you bring to your AWS account with BYOIP. (option disabled because no pools found) [Learn more](#)
- ☐ Customer-owned pool of IPv4 addresses created from your on-premises network for use with an Outpost. (option disabled because no customer owned pools found) [Learn more](#)
- ☐ Allocate using an IPv4 IPAM pool (option disabled because no public IPv4 IPAM pools with AWS service as EC2 were found)

Network border group Info

Global static IP addresses
AWS Global Accelerator can provide global static IP addresses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using the Amazon global network. [Learn more](#)

[Create accelerator](#)

Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

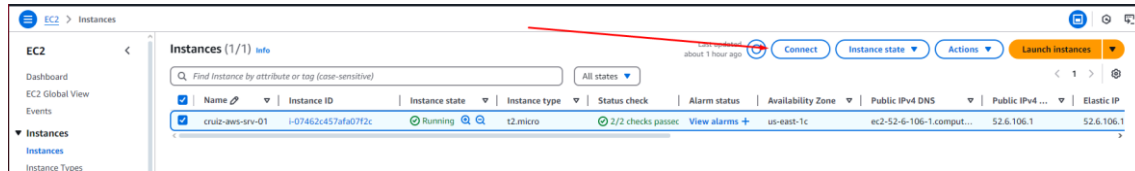
[Add new tag](#)

You can add up to 50 more tag

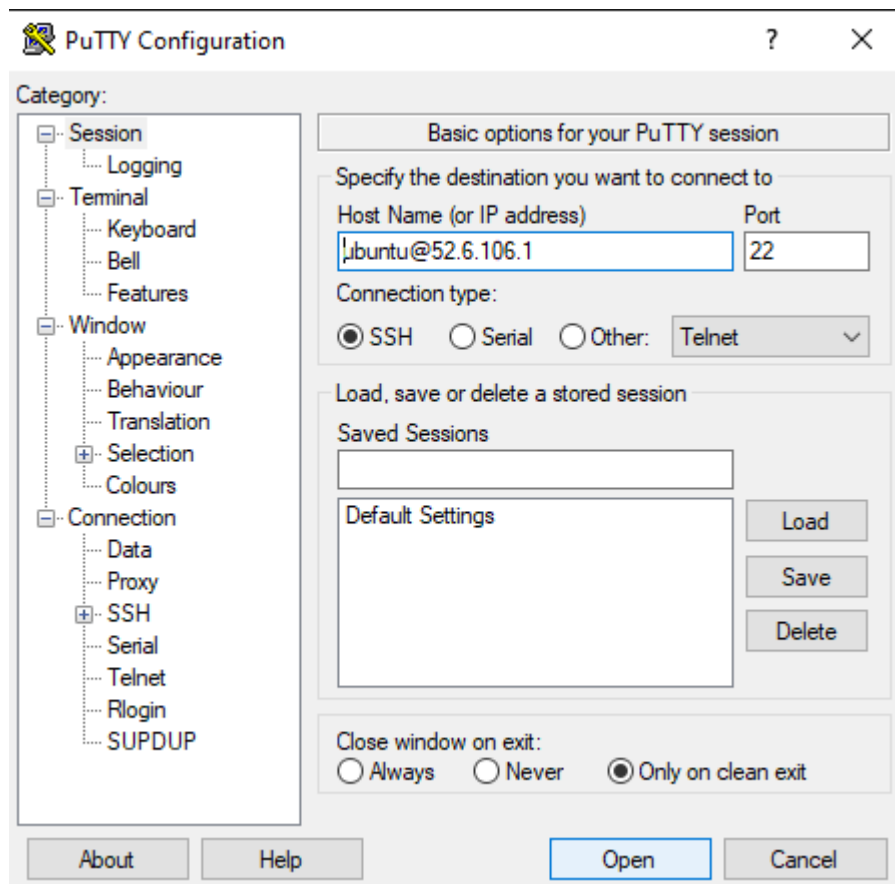
[Cancel](#) [Allocate](#)

Conexión a la máquina

Tras esto, podemos acceder a nuestra máquina bien usando el botón connect de la página de instancias:



O bien de una forma externa. Nosotros hemos usado PuTTY, usando el archivo de credenciales que Amazon nos da.



Tras esto, entramos a nuestro Ubuntu server:

```

ubuntu@ip-172-31-85-163: ~
System load:  0.0          Processes:      110
Usage of /:   38.9% of 6.71GB  Users logged in:  0
Memory usage: 59%          IPv4 address for enX0: 172.31.85.163
Swap usage:   0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

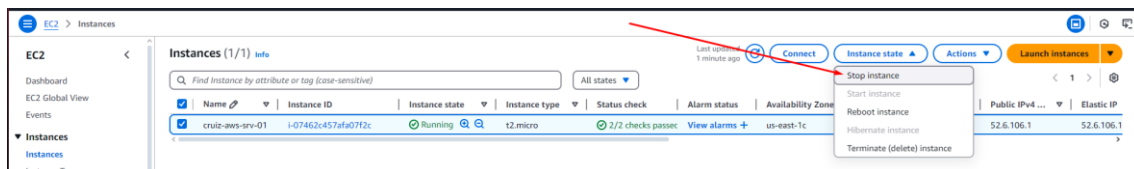
5 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Wed Feb 26 21:19:56 2025 from 87.223.72.50
ubuntu@ip-172-31-85-163:~$
  
```

Y ya es cuestión de hacer lo que queremos. En nuestro caso, levantar el servidor fue tan simple como instalar apache2 y mysql, importar la base de datos y clonar con git el repositorio de nuestro proyecto.

Finalmente, una vez que terminamos de trabajar en nuestra máquina, podemos pararla en la sección de instancias:



Para ello también podemos simplemente terminar la sesión Lab en nuestra consola, y esto parará la máquina:



Para conectarnos a la máquina cuando está activa, usaremos la dirección <http://52.6.106.1/>

Mock-up

Hemos recreado el mockup de la página en Figma:

<https://www.figma.com/design/jApPUwO6lXRJm2J5JNTxTw/Tabletop-Finder?node-id=0-1&t=ciEU7RenLCEDwEY1-1>

El esquema de la página se ve tal que así:



Decisiones de diseño

Para las fuentes, he decidido usar Arial para los textos normales, y Almendra para los títulos y textos destacados.



Para los colores, elegí un beige claro y un marrón que hicieran contraste, con un naranja cálido para los detalles y los botones que resaltase un poco.



No se han utilizado iconos, salvo aquellos usados para indicar que áreas de los perfiles son editables, el cual es un simple carácter de Unicode ✎.

Elementos interactivos

Algunos de los elementos interactivos de la página son:

Botones: Tales como los de registro, inicio de sesión, o gestión de notificaciones.

Puedes ver el listado de juegos disponibles aquí

-

Guardar

Solicitud de unión: El usuario Noelia desea unirse a tu anuncio Not Alone por las tardes .

Aceptar Rechazar

Solicitud de unión: El usuario MasterMagic te ha aceptado en Perdidos por Tabletop Simulator .

Eliminar notificación

Notificación de mensaje: Noelia te ha enviado un mensaje.

Eliminar notificación Ver el mensaje

Menús: Como el menú principal, o los menús emergentes del usuario o admin



Formularios y campos de entrada: En los registros de usuario, anuncio o secciones de admin

The image shows two screenshots of the application's registration and announcement forms. The top screenshot is titled 'CONVERTIR USUARIO EN TIENDA' and contains two sections. Each section has a 'Selecciona usuario:' label, a dropdown menu with '-- Selecciona un usuario --', and a button labeled 'Hacer tienda'. The bottom screenshot is a detailed announcement form. It has two columns of fields. The left column includes 'Titulo del anuncio' (Maximum 35 characters), 'Descripción' (Maximum 500 characters), 'Fondo del anuncio' (Maximum 500 characters), and 'Juego' (Maximum 30 characters). The right column includes 'Plazas' (Maximum 99 plazas), 'Descripción corta' (Maximum 30 characters), 'Medio de la partida' (Maximum 30 characters), and 'Online'. A 'Guardar' button is at the bottom.

Integración con Google y RRSS

Google Analytics



Se ha hecho tanto la con Google analytics, integrand su código en nuestro codigo html del que parte cada página:

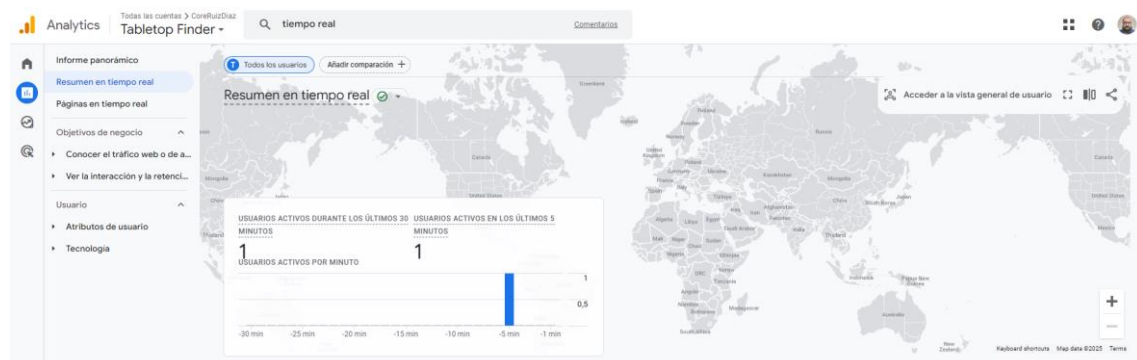
```

ubuntu@ip-172-31-85-163: /var/www/tabletopfinder/aplicacion/resources/views/layouts
GNU nano 7.2          basico.blade.php
[!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js">
  <link href="https://cdn.datatables.net/v/dt/dt-1.13.4/datatables.min.css" rel="stylesheet">
  <script src="https://cdn.datatables.net/v/dt/dt-1.13.4/datatables.min.js"></script>
  <!-- Google tag (gtag.js) -->
  <script async src="https://www.googletagmanager.com/gtag/js?id=G-DQSBF1HDHW"></script>
  <script>
    window.dataLayer = window.dataLayer || [];
    function gtag(){dataLayer.push(arguments);}
    gtag('js', new Date());

    gtag('config', 'G-DQSBF1HDHW');
  </script>

  @yield('scripts')
  
```

Y podemos comprobar que, efectivamente, funciona y recoge datos.



Google ads



Mi propuesta para palabras clave que usar en Google Ads:

- Palabras clave generales:

- juegos de mesa
- partidas de rol
- juegos de rol
- juegos de mesa online
- partidas de rol online

- Palabras clave específicas y de localización:

- juegos de mesa Almería
- partidas de rol en Almería
- eventos de juegos de mesa Almería
- torneos de rol Almería
- convocatorias juegos de mesa Almería

- Palabras clave relacionadas con funcionalidades de la aplicación:

- crear partidas de rol
- buscar partidas de juegos de mesa
- registrarse en partidas de rol
- chat para jugadores de mesa
- plataforma de juegos de rol

- Segmentación geográfica:

- Enfocar la campaña en la provincia de Almería o áreas cercanas.

- Segmentación demográfica:

- Personas interesadas en juegos de mesa, rol y actividades lúdicas.

- Considerar rangos de edad que suelen participar en estas actividades (por ejemplo, 14-45 años).

Intereses y comportamientos:

Usuarios que sigan páginas o grupos relacionados con juegos de mesa y rol en redes sociales.

Personas que hayan buscado previamente información sobre eventos, torneos o convenciones de rol.

Esta combinación te ayudará a atraer tanto a los jugadores habituales como a aquellos que buscan nuevos eventos o partidas en la zona, maximizando la relevancia de tus anuncios y mejorando la tasa de conversión.

Redes sociales

Twitter (<https://x.com/tabletopalmeria>)



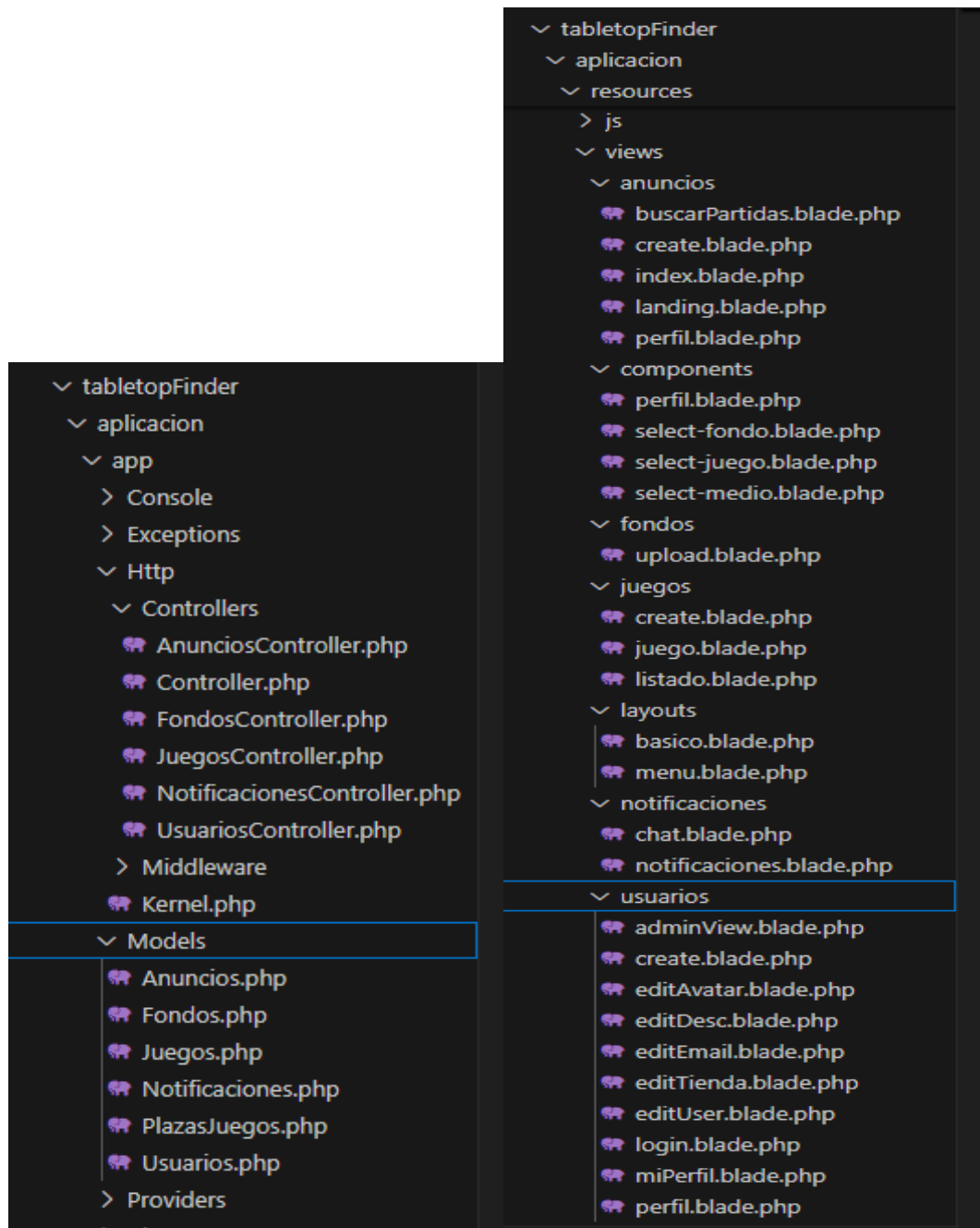
Facebook



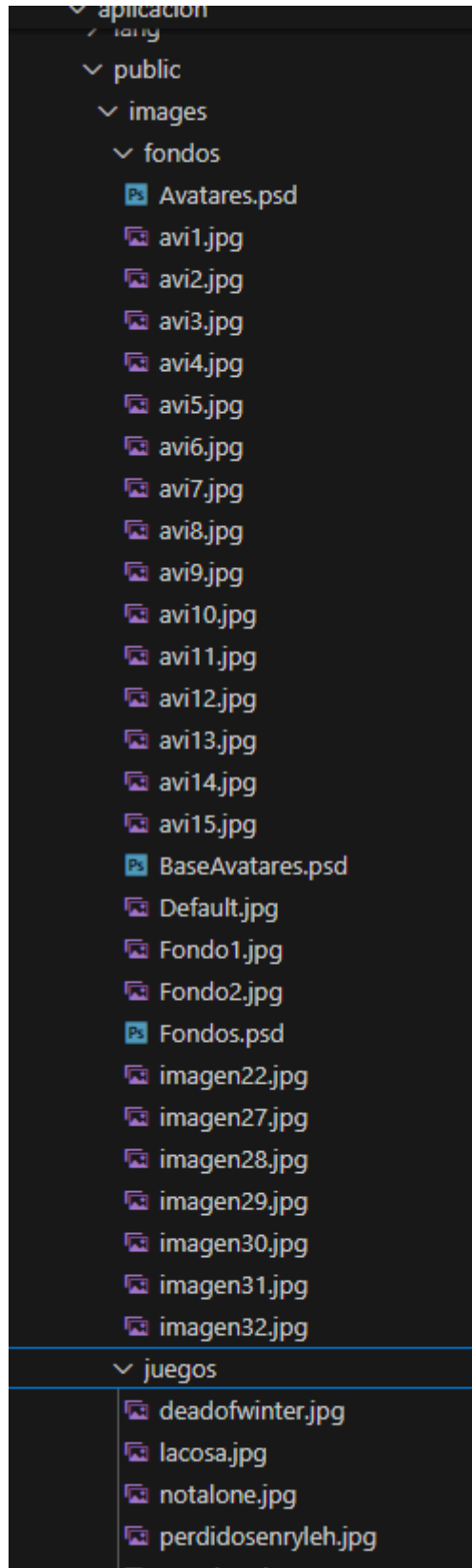
Documentación del código

Estructura de datos

Ya que usamos Laravel, tenemos controladores, modelos y vistas, que se reparten de la siguiente manera:



Y, a la hora de organizar las imágenes, en public/images tenemos dos carpetas: fondos y juegos



Aprovechamos la capacidad de los blade de laravel, lo cual nos permite segmentar las vistas, haciéndolas bastante manejables.

A la hora de documentar el código, se ha especificado al principio de cada método qué hace y donde se usa, para saber dónde buscar si hay fallos o necesidad de retocar algo:

```
/*
 * Método que manda las partidas a la datable a través de AJAX con el formato que le pedimos.
 * Se usa a través de la datable de buscarPartida
 *
 */

public function getListado(Request $request)
{
    $where = 'WHERE 1=1';

    if (!empty($request->search['value'])) {
        $where .= ' AND (';
        $stringAdded = false;
        foreach ($request->columns as $col) {
            $searchable = json_decode($col['searchable']);
            if ($searchable) {
                if ($stringAdded) {
                    $where .= ' OR ';
                }
                $where .= $col['name'] . " LIKE '%" . $request->search['value'] . "%'";
                $stringAdded = true;
            }
        }
    }
}
```

Tan solo si algún bloque de código se hacía más complicado se ha especificado algo, pero en general, es un código simple y sencillo:

```
// Este bloque nos manda los anuncios que le pedimos desde la datable con el conteo de jugadores por partida en tiempo real, así como
// el nombre de los juegos y los jugadores

$anuncios = DB::select(
    'SELECT a.*, j.nombre as juegonombre, u.username as usuarionombre
    FROM anuncio a
    LEFT JOIN juegos j ON a.juegocode = j.code
    LEFT JOIN usuarios u ON a.userid = u.uid ' .
    $where .
    ' AND a.plazas <> (SELECT COUNT(*) FROM plazasjuegos p WHERE p.uuidanuncio = a.uuid) ' .
    $orderBy . ' ' . $paginacion
);
```

En todo momento, desde el planteamiento de las bases de datos, se ha buscado que toda la información se busque en las bases de datos constantemente, en lugar de guardar información que luego podría cambiar; por lo que es muy común encontrar búsquedas con joins.

```
$nombreJuego = DB::table('anuncio')
    ->join('juegos', 'anuncio.juegocode', '=', 'juegos.code')
    ->select('juegos.nombre')
    ->where('anuncio.uuid', $uuid)
    ->get();
```

Sobre todo, para coger la información de los usuarios, las imágenes, y los juegos desde las tablas.

```
$anuncios = DB::select(
    'SELECT a.*, j.nombre as juegonombre, u.username as usuarionombre
    FROM anuncio a
    LEFT JOIN juegos j ON a.juegocode = j.code
    LEFT JOIN usuarios u ON a.useruuid = u.uuid ' .
    $where .
    ' AND a.plazas <> (SELECT COUNT(*) FROM plazasjuegos p WHERE p.uuidanuncio = a.uuid) ' .
    $orderBy . ' ' . $paginacion
);

$recordsFiltered = DB::select(
    'SELECT COUNT(a.uuid) as recordsNum
    FROM anuncio a
    LEFT JOIN juegos j ON a.juegocode = j.code
    LEFT JOIN usuarios u ON a.useruuid = u.uuid ' .
    $where .
    ' AND a.plazas <> (SELECT COUNT(*) FROM plazasjuegos p WHERE p.uuidanuncio = a.uuid)'
)[0]->recordsNum;
```

La gran mayoría del código se basa en hacer búsquedas en las tablas y devolver variables con la información, por ejemplo, a la hora de mostrar las vistas:

```
$usuarios = DB::table('usuarios')->whereIn('uuid', $jugadoresIds)->get();

$creador = DB::table('usuarios')->where('uuid', $anuncio->useruuid)->first();

$avatarNames = $usuarios->pluck('avatar')->unique();
if ($creador && !$avatarNames->contains($creador->avatar)) {
    $avatarNames->push($creador->avatar);
}

$fondosAvatar = DB::table('fondos')
    ->whereIn('nombre', $avatarNames)
    ->get()
    ->keyBy('nombre');

// Estos tres campos son necesarios para saber que le mostramos al usuario

$usuarioYaInscrito = DB::table('plazasjuegos')
    ->where('uuidanuncio', $uuid)
    ->where('uuiduser', auth()->user()->uuid)
    ->exists();

$usuarioYaSolicitante = DB::table('notificaciones')
    ->where('anuncio', $uuid)
    ->where('solicitante', auth()->user()->uuid)
    ->exists();

$usuarioEsCreador = ($anuncio->useruuid == auth()->user()->uuid);

return view('anuncios.perfil', compact(
    'anuncio',
    'fondo',
    'usuarios',
    'fondosAvatar',
    'usuarioYaInscrito',
    'usuarioEsCreador',
    'usuarioYaSolicitante',
    'nombreJuego',
    'creador'
));
}
```

Mención especial a nuestro archivo web:

```
// Sección de administrador

Route::get('/usuarios/admin', [UsuariosController::class, 'admin'])->name('usuarios.admin');
Route::post('/usuarios/makeTienda', [UsuariosController::class, 'makeTienda'])->name('usuarios.makeTienda');
Route::post('/usuarios/banHammer', [UsuariosController::class, 'banHammer'])->name('usuarios.banHammer');

// Sección de anuncios

Route::get('/', [AnunciosController::class, 'landing'])->name('anuncios.landing');
Route::get('/anuncios/index', [AnunciosController::class, 'index'])->name('anuncios.index');
Route::get('/anuncios/landing', [AnunciosController::class, 'landing'])->name('anuncios.landing');
Route::get('/anuncios/create', [AnunciosController::class, 'create'])->name('anuncios.create');
Route::post('/anuncios/store', [AnunciosController::class, 'store'])->name('anuncios.store');
Route::post('/anuncios/getListado', [AnunciosController::class, 'getListado'])->name('anuncios.getListado');
Route::get('/anuncios/buscarPartidas', [AnunciosController::class, 'buscarPartidas'])->name('anuncios.buscarPartidas');
Route::get('/anuncios/{id}/', [AnunciosController::class, 'show'])->name('anuncios.show');
Route::delete('/anuncios/{uuid}', [AnunciosController::class, 'destroy'])->name('anuncios.destroy');

// Sección usuarios

Route::get('/usuarios/create', [UsuariosController::class, 'create'])->name('usuarios.create');
Route::get('/usuarios/logout', [UsuariosController::class, 'logout'])->name('usuarios.logout');
Route::get('/usuarios/miPerfil', [UsuariosController::class, 'miPerfil'])->name('usuarios.miPerfil');
Route::get('/usuarios/login', [UsuariosController::class, 'login'])->name('usuarios.login');
Route::post('/usuarios/authenticate', [UsuariosController::class, 'authenticate'])->name('usuarios.authenticate');
Route::post('/usuarios/createUser', [UsuariosController::class, 'createUser'])->name('usuarios.createUser');
Route::get('/usuarios/{nombre}', [UsuariosController::class, 'show'])->name('usuarios.show');

// Edición de campos en perfil

Route::put('/usuarios/{id}/updateEmail', [UsuariosController::class, 'updateEmail'])->name('usuarios.updateEmail');
Route::get('/usuarios/{id}/editDesc', [UsuariosController::class, 'editDesc'])->name('usuarios.editDesc');
Route::put('/usuarios/{id}/updateDesc', [UsuariosController::class, 'updateDesc'])->name('usuarios.updateDesc');
Route::get('/usuarios/{id}/editEmail', [UsuariosController::class, 'editEmail'])->name('usuarios.editEmail');
Route::put('/usuarios/{id}/updateUser', [UsuariosController::class, 'updateUser'])->name('usuarios.updateUser');
Route::get('/usuarios/{id}/editUser', [UsuariosController::class, 'editUser'])->name('usuarios.editUser');
Route::put('/usuarios/{id}/updateAvatar', [UsuariosController::class, 'updateAvatar'])->name('usuarios.updateAvatar');
Route::get('/usuarios/{id}/editAvatar', [UsuariosController::class, 'editAvatar'])->name('usuarios.editAvatar');
Route::get('/usuarios/{id}/editTienda', [UsuariosController::class, 'editTienda'])->name('usuarios.editTienda');
Route::put('/usuarios/{id}/updateTienda', [UsuariosController::class, 'updateTienda'])->name('usuarios.updateTienda');

// Sección fondos

Route::get('/fondos/create', [FondosController::class, 'create'])->name('fondos.create');
Route::post('/fondos/upload', [FondosController::class, 'upload'])->name('fondos.upload');
```

Hacemos utilización del DOM en alguna que otra clase:

```
<script>
    document.addEventListener('DOMContentLoaded', function() {
        let select = document.getElementById('fondo');
        if (select) {
            select.addEventListener('change', function() {
                let imagenSeleccionada = this.options[this.selectedIndex].getAttribute('ruta');
                document.getElementById('crearanuncio-imagen-preview').src = imagenSeleccionada;
            });
        }
    });
</script>

<h2 class="crearanuncio-titulo">Crear partida</h2>

<div class="crearanuncio-form-container">
    <!-- Formulario -->
    <form action="{{ route('anuncios.store') }}" method="post" class="crearanuncio-form">
        @csrf
```

Y también hay algunas consultas AJAX, ya que usamos DataTable:

```

@section('scripts')
<script>
$(document).ready(function() {
    function inicializarDataTable(selector, esTienda) {
        var pageLength = esTienda == 0 ? 10 : 20;
        return $(selector).DataTable({
            "paging": true,
            "pagingType": "numbers",
            "lengthChange": false,
            "searching": false,
            "pageLength": pageLength,
            "info": false,
            "order": [[1, "asc"]],
            "processing": true,
            "serverSide": true,
            "ajax": {
                "url": "{ route('anuncios.getlistado') }",
                "type": "POST",
                "data": function(d) {
                    d.juego = $('#juego').val();
                    d.tipoPartida = $('input[name=tipoPartida]:checked').val();
                    d._token = "{{ csrf_token() }}";
                    d.esTienda = esTienda;
                    console.log(d);
                }
            },
            "columns": [
                {
                    data: 'uuid',
                    name: 'uuid',
                    orderable: false,
                    searchable: false,
                    visible: false
                },
                {
                    data: 'titulo',
                    name: 'titulo',
                    render: function(data, type, row) {
                        return '<a href="/tabletopfinder/aplicacion/public/anuncios/' + encodeURIComponent(row.uuid) + ">' + data + '</a>';
                    }
                },
                {
                    data: 'creador',
                    name: 'creador',

```

Bibliografía y documentación

Laravel Documentation

<https://laravel.com/docs>

Referencia para el framework utilizado en el back-end.

PHP Manual

<https://www.php.net/manual/en/>

Documentación oficial del lenguaje PHP.

MariaDB Documentation

<https://mariadb.com/kb/en/documentation>

Fuente para la configuración y el uso de MariaDB.

phpMyAdmin Documentation

<https://docs.phpmyadmin.net/>

Referencia sobre la gestión de la base de datos a través de phpMyAdmin.

WCAG 2.1 Guidelines

<https://www.w3.org/TR/WCAG21/>

Guía para asegurar la accesibilidad y el contraste en el diseño.

Amazon Web Services (AWS) Documentation

<https://aws.amazon.com/documentation/>

Documentación sobre el despliegue y configuración de servidores en AWS.

MDN Web Docs

<https://developer.mozilla.org/en-US/>

Referencia para HTML, CSS y JavaScript, ofreciendo guías y ejemplos prácticos.

Stackoverflow

<https://stackoverflow.com/>

Referencia para dudas buscadas en internet.

W3Schools

<https://www.w3schools.com>

Referencia para dudas buscadas en internet.

Coderanch

<https://coderanch.com>

Referencia para dudas buscadas en internet.

Reddit

<https://reddit.com>

Referencia para dudas buscadas en internet.

CSS-Tricks

<https://css-tricks.com>

Sitio web con consejos y técnicas para el diseño y maquetación web, muy útil para resolver dudas relacionadas con CSS y front-end.