

Memoria TFC

Rubén García Díaz-Maroto

Agustín Jiménez Fernández

Anibal Martin

EXO GAMES

Curso 2021/22

**Grado Superior
Desarrollo de aplicaciones web**



Contenido

1 Agradecimientos/dedicatoria.....	4
2 Resumen	5
3 Abstract	6
4 Introducción	7
5 Alcance funcional del sistema	8
6 Diseño técnico	10
6.1 Diagrama de clases	10
6.2 Esquema de bases de datos.....	12
6.3 Diseño de la interfaz de usuario	15
7 Tecnologías utilizadas.....	26
8 Código documentado	29
9 Presupuesto	31
10 Planificación.....	33
11 Manual de instalación.....	35
12 Manual de usuario	35
13 Conclusiones.....	36
14 Análisis de la competencia y de los clientes	36
15 Bibliografía.....	39



1 Agradecimientos/dedicatoria

Agradecemos a nuestro profesor del módulo de desarrollo de aplicaciones web, Javier Prieto Abad, por enseñarnos todo lo que conlleva el desarrollo de aplicaciones web en entornos servidor durante todo el año.

También queremos agradecer a Aníbal Martín por tutorizarnos y enseñarnos la importancia la seguridad en un despliegue de servidores junto a las metodologías ágiles (SCRUM, Kanban). Junto con el asesoramiento a la hora del desarrollo de nuestro proyecto de final de curso.

Agradecer a Rafael Chacón por inculcarnos sus ganas de aprender y disfrutar con lo que hacemos en el aula, a la par de enseñarnos bases de datos y sistemas informáticos.

A Javier Prieto por ser como un segundo tutor, guiarnos en la organización y hacernos llevar al día todo lo aprendido. Junto con la ayuda a la estructuración de código y limpieza de este.

Agradecer también a Charo Martínez por enseñarnos la estructuración y maquetación de diseño a la hora de realizar nuestro proyecto. Sobre todo, la importancia de aplicar una escalabilidad a los estilos de las páginas web.

Y por último agradecer a Rubén González Martín por enseñarnos las bases en la programación con JavaScript permitiéndonos así dinamizar nuestras aplicaciones.

2 Resumen

Nuestro proyecto tendremos un desarrollo web multiplataforma PC/Android.

- **Página web** con sistema de compra de videojuegos y consolas correspondientes en formato físico como digital, con posibilidad de múltiples funciones como:
 - Administración de roles ya sea cliente o administrador cambiará la interfaz de dicha aplicación junto con su funcionalidad respectiva.
 - Sistema de inicio/registro de sesión.
 - El cliente podrá seleccionar si quiere comprar o vender su producto en nuestra plataforma.
 - Gestión datos desde panel de administrador
 - Carrito.
 - Tipos de filtrado.
 - Catálogo de videojuegos/consolas.
 - Vista individual por videojuego/consola.
 - Demo(emulación) de cada videojuego seleccionado junto a la emulación de sus respectivas consolas.
 - Opción de venta de juegos de segunda mano y poder guardar los datos del añadido.
 - Principalmente el tema estará destinado a lo retro.
- **App Android** tendrá la funcionalidad de:
 - Manejo de usuarios.
 - Inserción de artículos de segunda mano.
 - Inserción de noticias.
 - Visualización de listas con los productos de segunda mano.
 - Visualización de listas de noticias.
 - Configuración de perfil



3 Abstract

Our project we will have a multiplatform web development PC/Android.

- **Website** with video games and consoles purchase system in physical and digital format, with the possibility of multiple functions such as:
 - Role management either client or administrator will change the interface of such application along with their respective functionality.
 - Login/registration system.
 - The client will be able to select if he wants to buy or sell his product in our platform.
 - Manage data from an administrator panel
 - Shopping cart.
 - Filtering types.
 - Video games/consoles catalog.
 - Individual view by videogame/console.
 - Demo (emulation) of each video game selected along with the emulation of their respective consoles.
 - Option to sell second hand games and to be able to save the data of the added one.
 - Mainly the theme will be aimed at retro.

- **Android App will have the functionality of:**
 - User management.
 - Insertion of second-hand items.
 - Insertion of news.
 - Visualization of lists with second-hand products.
 - Visualization of news lists.
 - Profile settings

4 Introducción

Nuestro proyecto está basado en una plataforma online de compra/venta de videojuegos como sus respectivas consolas debido al incremento del interés en relación a las nuevas tecnologías por parte del público.

Dentro del área de la tecnología vimos que destacaba mas en los temas destinados al ocio teniendo como referencia los videojuegos. Decidimos desarrollar un portal para facilitar la navegación a todo tipo de usuarios ya que en su mayoría todos disponen de un dispositivo móvil o un PC.

Al disponer de un entorno web/Android hace que las posibilidades de llegar y captar un mayor número de usuarios sean mayores, siendo nuestro público de referencia abarcando un rango de edad entre los 15-35 años mayoritariamente. Suele haber un porcentaje no tan alto en otros rangos de edad que consuman este tipo de servicios/productos.

Nos extendemos por el plano móvil ya que la mayoría de usuarios de la red utilizan sus dispositivos Android a diario y por la facilidad de carga de los elementos a mostrar en nuestra plataforma

Según un artículo de MDO Madridiario "Internet ha permitido que muchas de las actividades que antes se realizaban de forma presencial hoy se puedan hacer a través de un ordenador o de un móvil. Sucede de igual manera con los juegos, de los que hoy existen muchas variantes online, y siendo una de las industrias que mayor crecimiento reporta en la actualidad."

Este fue uno de los motivos principales por el que desarrollamos EXOGAMES.

Decidimos poner en marcha el proyecto siendo STEAM la única plataforma online con referencia a la nuestra buscada en Google, marcando la diferencia con la disponibilidad de venta de productos ya sean consolas o videojuegos y la emulación de los mismo en la plataforma web/Android.



5 Alcance funcional del sistema

▪ Listado de videojuegos

- Visualización de información individual de los videojuegos/consolas seleccionadas.
- Visualización de juegos/consolas por filtrado.

▪ Gestión de videojuego:

- Compra del producto detallada. (Aplicación Web)
- Ver fotos y videos del videojuego seleccionado.
- Demo (Emulación) del videojuego solo disponible en Aplicación Web.
- Visualización de valoraciones. (Aplicación Web)
- Descripción del mismo.

▪ Gestión de consola:

- Compra del producto detallada (Aplicación Web)
- Ver fotos y videos de la consola seleccionada. (Aplicación Web)
- Visualización de valoraciones. (Aplicación Web)
- Descripción del mismo.

▪ Venta de videojuegos/consolas:

- El usuario podrá publicar su producto respectivo en nuestra plataforma online facilitando una imagen, nombre, descripción, valoración personal, precio a desear.

▪ Noticias:

- En este apartado el usuario podrá visualizar todas las novedades actuales acerca del ámbito de las consolas y videojuegos.

▪ Gestión de usuarios:

- Añadir un perfil de usuario junto con su imagen, nombre y correo.
- Editar toda la información de cada usuario a su gusto.
- Eliminar la información del usuario seleccionado.

▪ Inicio/registro de sesiones:

- Añadir un usuario a la base de datos en base a su rol respectivo.
- Administración de cada usuario/producto desde el rol administrador.
(Aplicación Web)

▪ Panel administrador:

- CRUD de productos, usuarios y noticias
- Visualización de mensajes de cada cliente
- Visualización de los pedidos realizados en la web.
- Barra de búsqueda

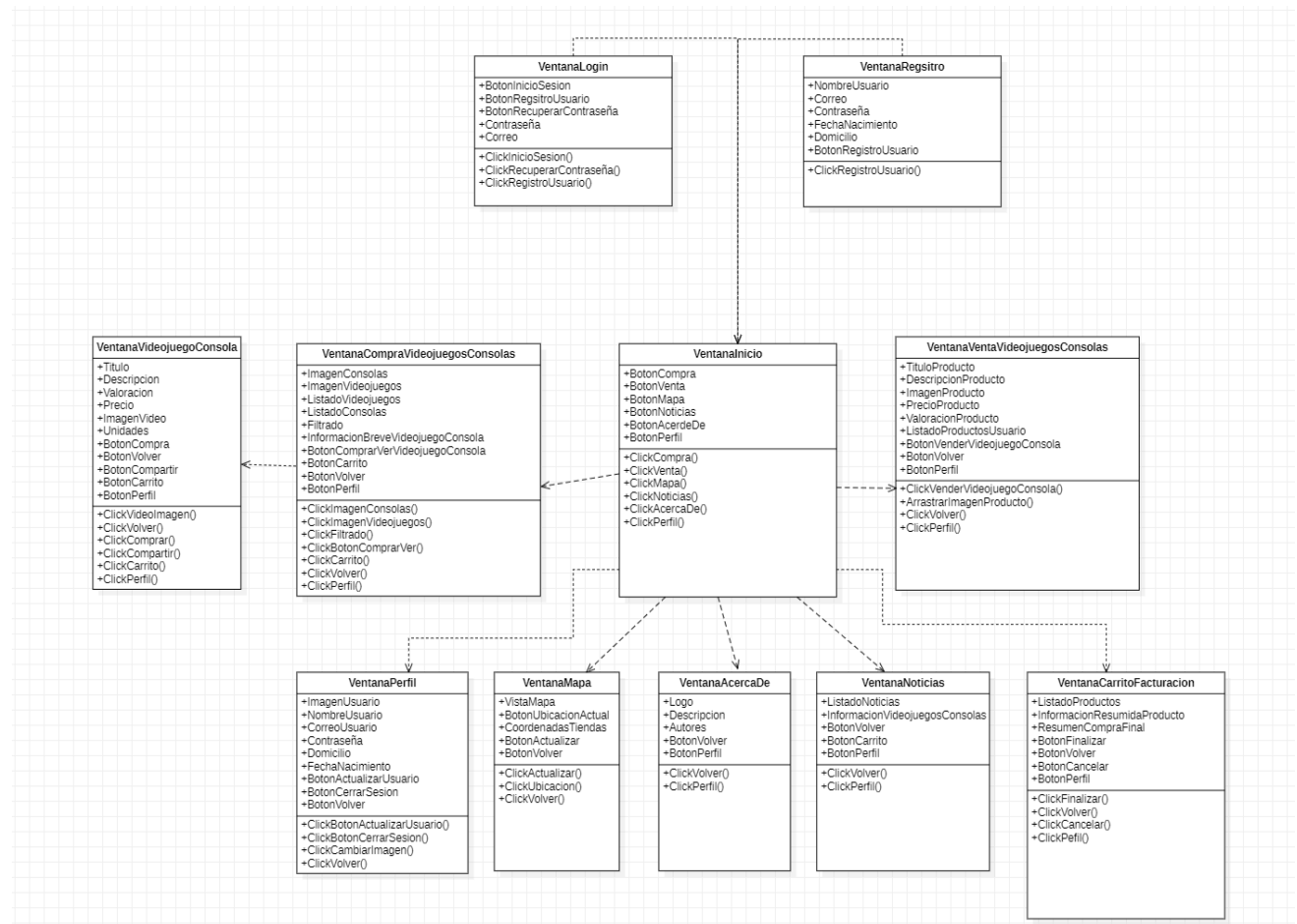
▪ Método de Pago y Pedidos

- Visualizar los pedidos realizados en el cliente actual tanto de productos vendidos como comprados.
- Verificación de datos y dirección del cliente. A futuro implementaremos método de pago como PayPal y Culqui desde Laravel.

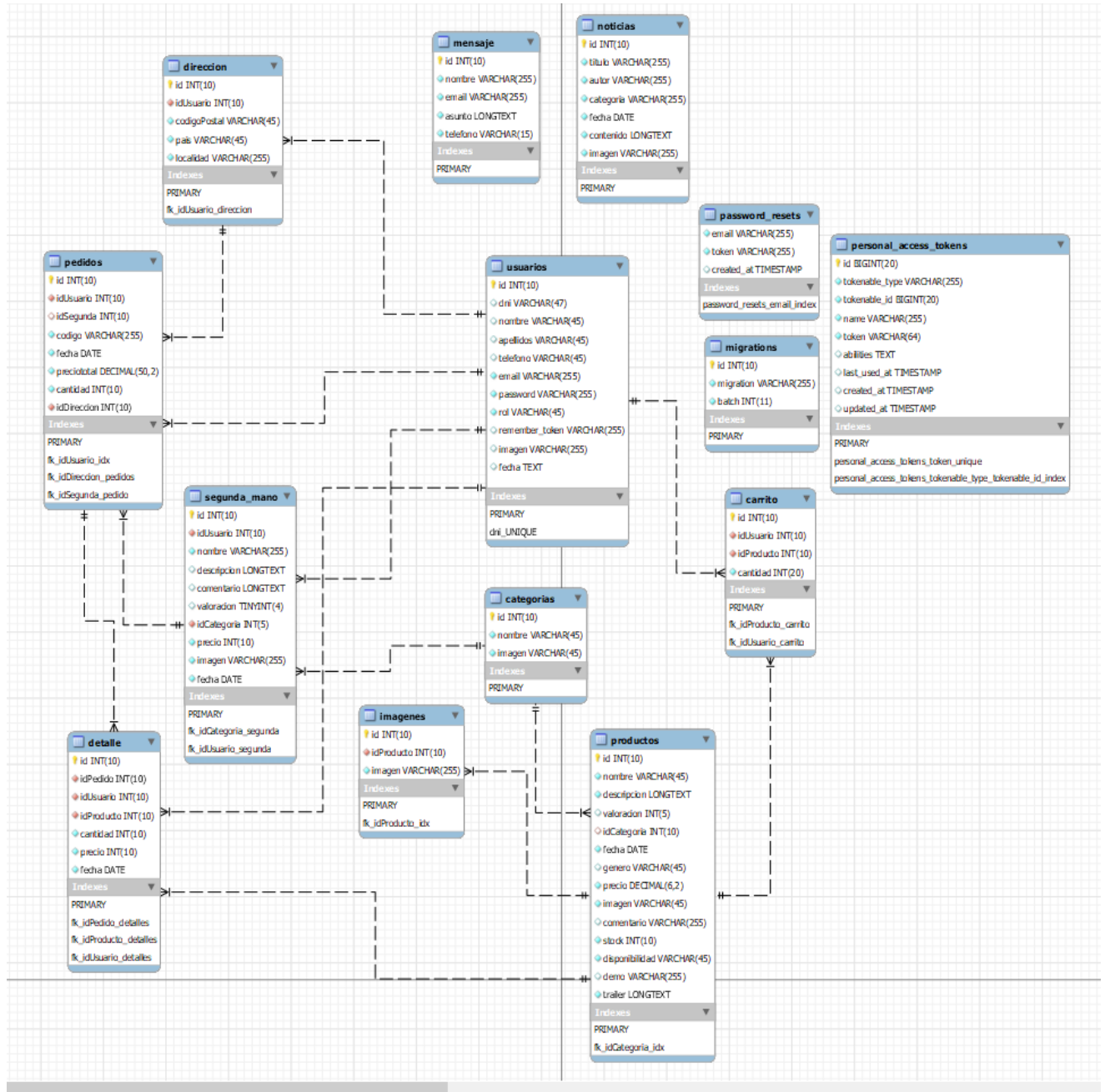
6 Diseño técnico

6.1 Diagrama de clases

Dispositivo Android



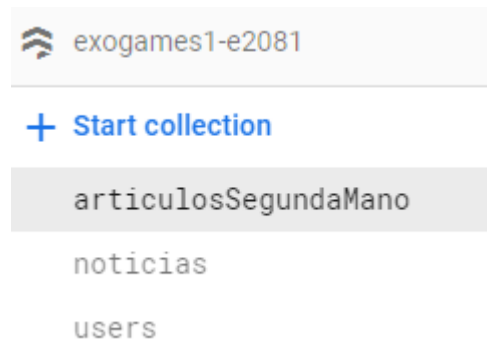
Aplicación Web



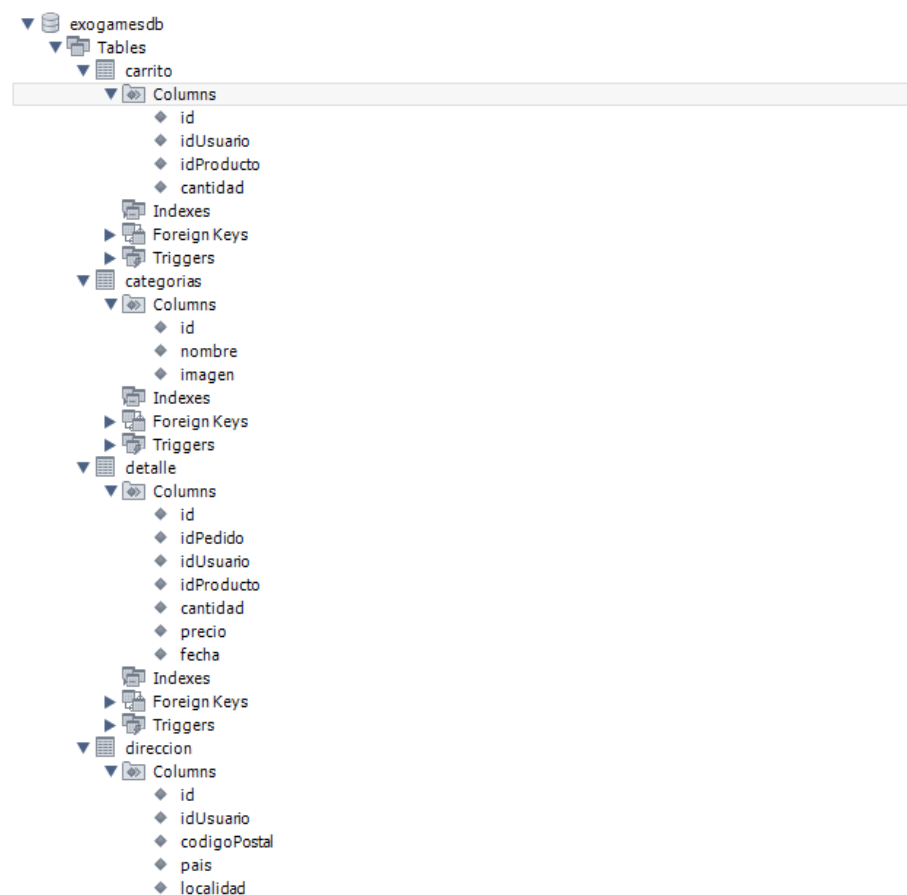


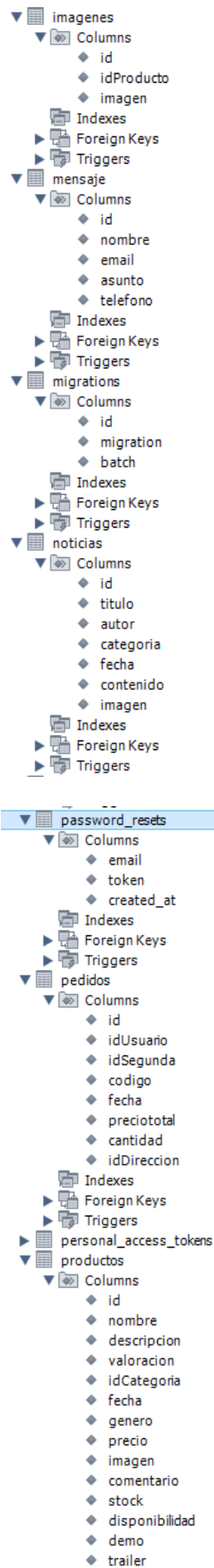
6.2 Esquema de bases de datos

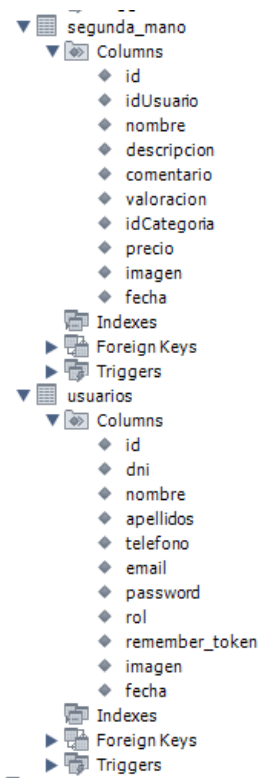
Dispositivo Android



Aplicación Web







6.3 Diseño de la interfaz de usuario

Planteamos la visualización general de la estructura sin emular la aplicación.

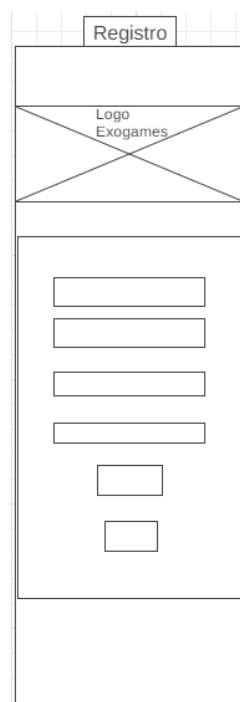
Utilizamos dos tipos de vistas en nuestro proyecto debido a que se diferencia en ciertos apartados debido a su funcionalidad específica para la plataforma. En Aplicación Web se utilizará la vista de monitor, Tablet y portátil a modo de utilizar los estilos responsive en la parte de FrontEnd. En Aplicación Android no será necesario aplicar más estilos.

En el apartado web se utiliza sistema de roles y en el apartado Android no será necesario debido a que el usuario siempre es cliente.

Ventana Login y Registro

En este apartado el usuario podrá darse de alta en nuestra web como en nuestra aplicación Android para poder acceder a la compra y venta de nuestros productos. Con opción de recordar la sesión del usuario.

Aplicación Android



Aplicación Web

LOGIN

TABLET

LOGO

USUARIO

CONTRASEÑA

LOGIN

☐ RECORDAR

REGISTRO

PORTATIL

LOGO

USUARIO

CONTRASEÑA

LOGIN

☐ RECORDAR

REGISTRO

MONITOR

LOGO

USUARIO

CONTRASEÑA

LOGIN

☐ RECORDAR

REGISTRO

logocopymetodos pago

REGISTRO

TABLET

LOGO

NOMBRE

CORREO

PASSWORD

FECHA

DIRECCION

☐ RECORDAR

REGISTRO

PORTATIL

LOGO

NOMBRE

APELLIDOS

EMAIL

PASSWORD

FECHA

☐ RECORDAR

REGISTRO

MONITOR

LOGO

NOMBRE

APELLIDOS

EMAIL

PASSWORD

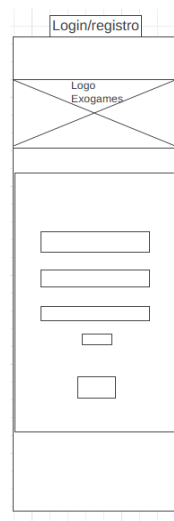
FECHA

☐ RECORDAR

REGISTRO

logocopymetodos pago

Aplicación Android

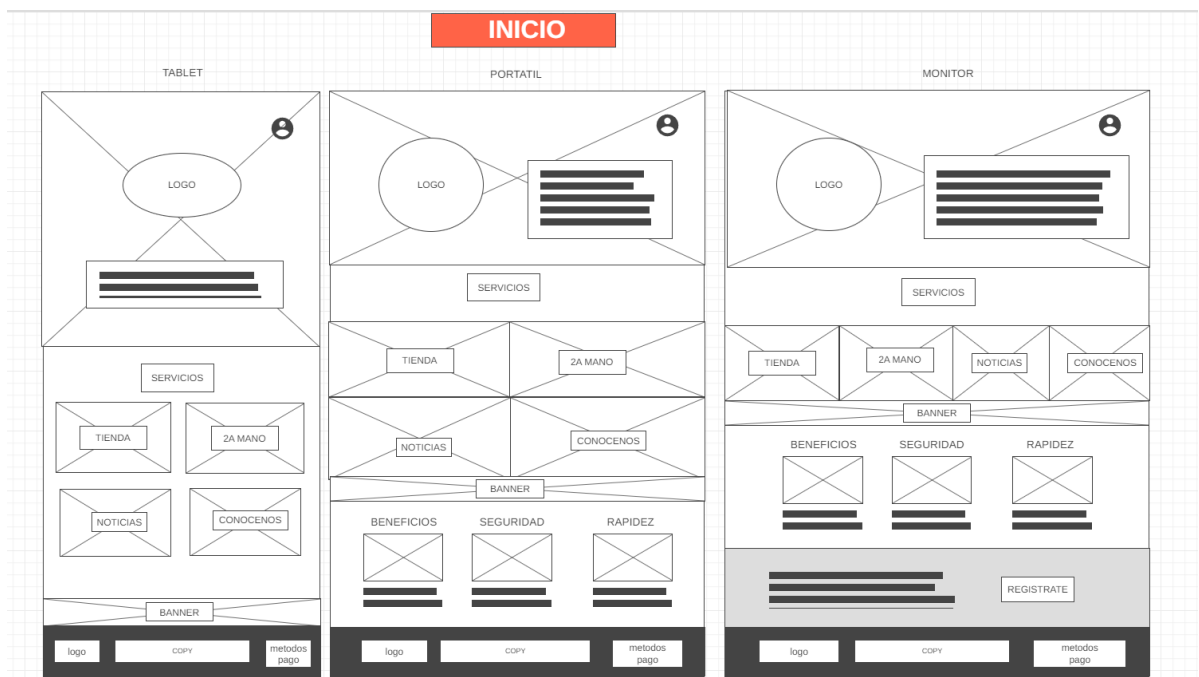


A plan de futuro si tenemos el tiempo necesario para verificar nuestro proyecto implementaríamos una ventana de recuperación de contraseña de la cual el usuario podrá introducir su correo electrónico y automáticamente se le enviará un correo verificando su identidad para poder restablecer/modificar sus credenciales.

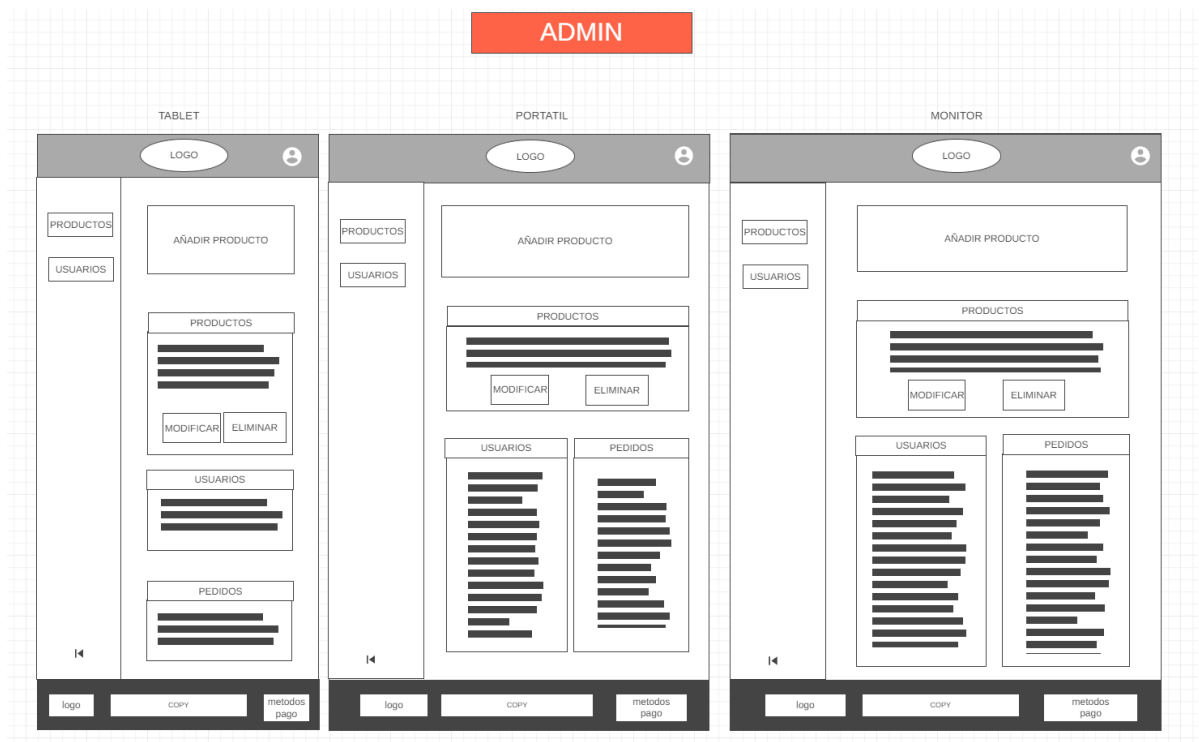
Ventana Principal

Aplicación Web

En este apartado mostrará al usuario ya accedido anteriormente con el rol tipo cliente ya que nuestra aplicación web utilizado sistema de roles al iniciar sesión o registrase. Mostrando un conjunto de secciones a modo de sinopsis de cada ventana de nuestra aplicación.

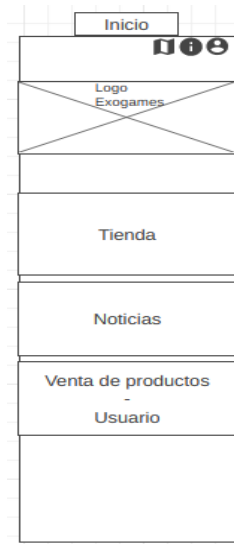


En la vista con el rol de administrador se mostrará las funciones de añadir tanto productos como clientes en nuestra base de datos. Como modificar estos mismo.

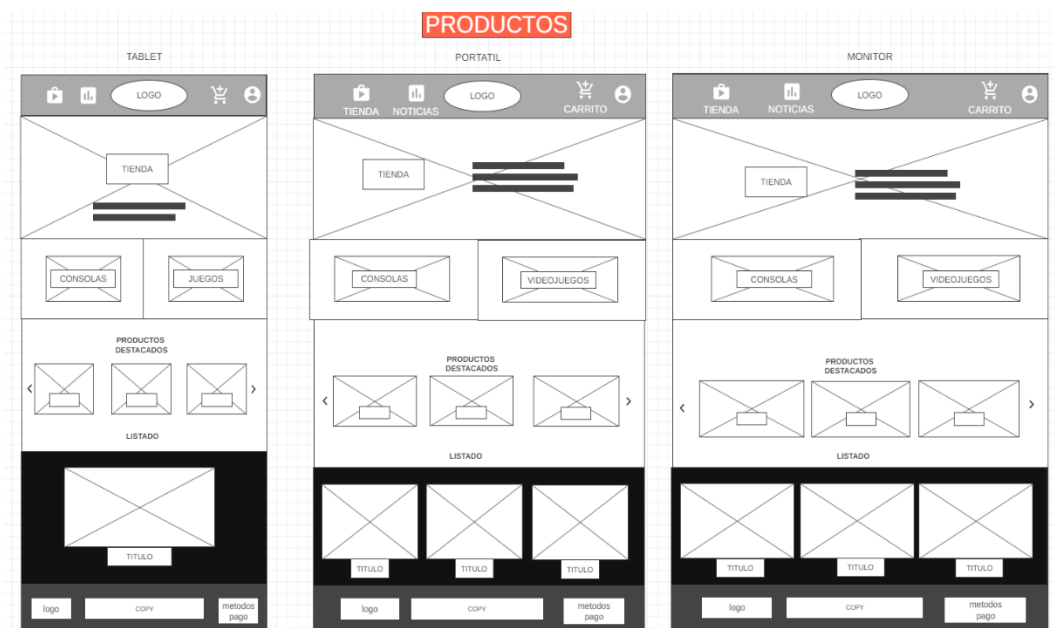


Aplicación Android

Posee la misma función que en el apartado de la aplicación web salvo la gestión de roles del cual siempre que iniciemos sesión el usuario accederá a la misma ventana.

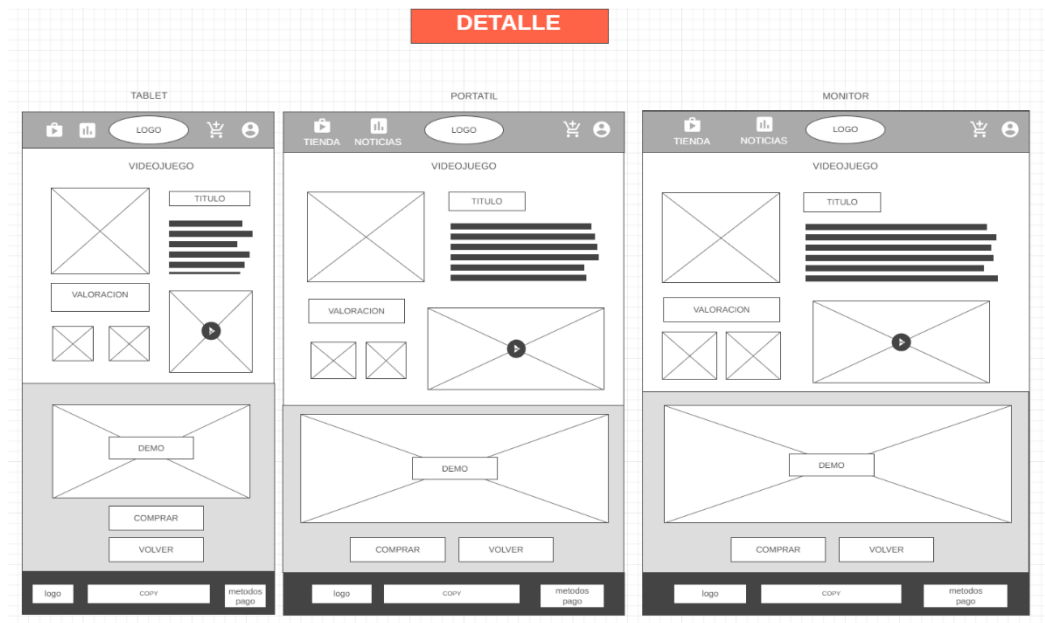


Aplicación Web



Aplicación Web

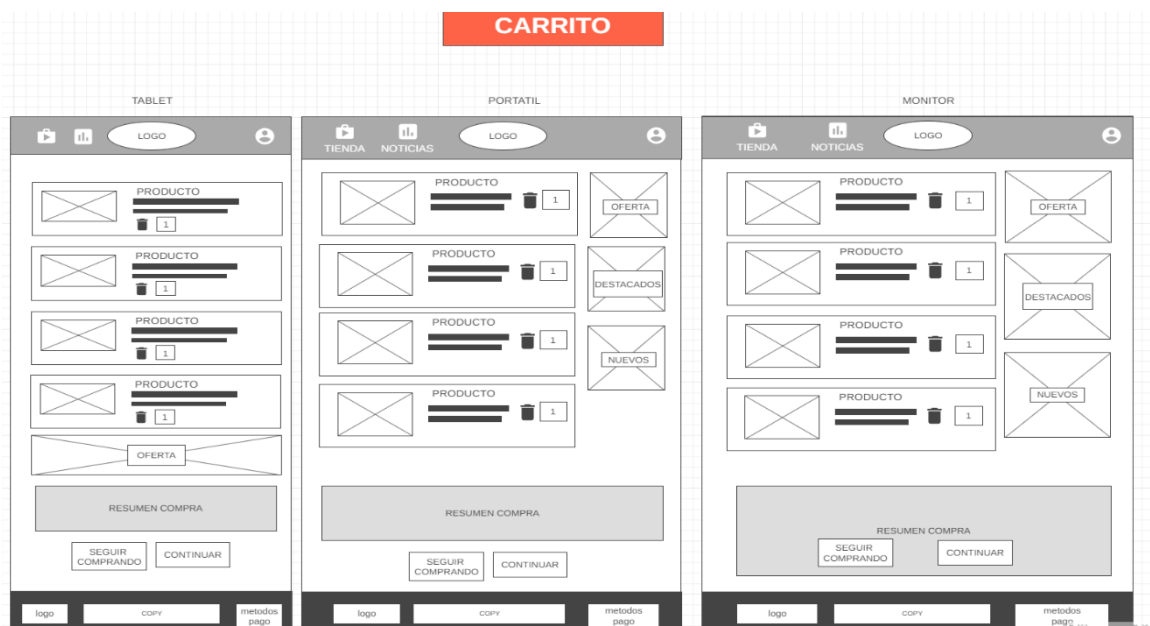
Se especifica la diferencia con la aplicación Android en la emulación del videojuego seleccionado a modo de demo que durara un tiempo limitado antes de comprarlo.



Ventana Carrito

Se almacenará los productos tanto consolas como videojuegos que haya comprado el usuario de nuestro catalogo disponible. Con la posibilidad de seguir comprando o finalizar la compra. Cada producto se podrá eliminar o ajustar las cantidades deseadas al finalizar. Mostrando un total sobre todo nuestro pedido.

Aplicación Web



Ventana Método de Pago

Se confirman la verificación de los datos del cliente junto a la elección del método de pago para realizar su pedido con sus respectivos campos de comprobación. Con la opción de continuar su pedido y mostrar su factura o cancelar el pedido.

Aplicación Web

METODO PAGO

Ventana Facturación

Mostrará nuestro pedido con la información completa de cada producto con posibilidad de imprimir o volver a la ventana de la Tienda. Indicando la fecha y los datos personales del cliente que ha facturado.

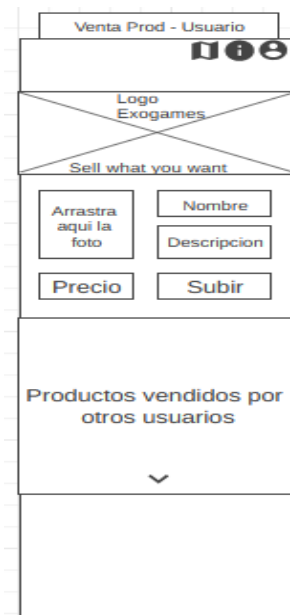
Aplicación Web

FACTURACIÓN

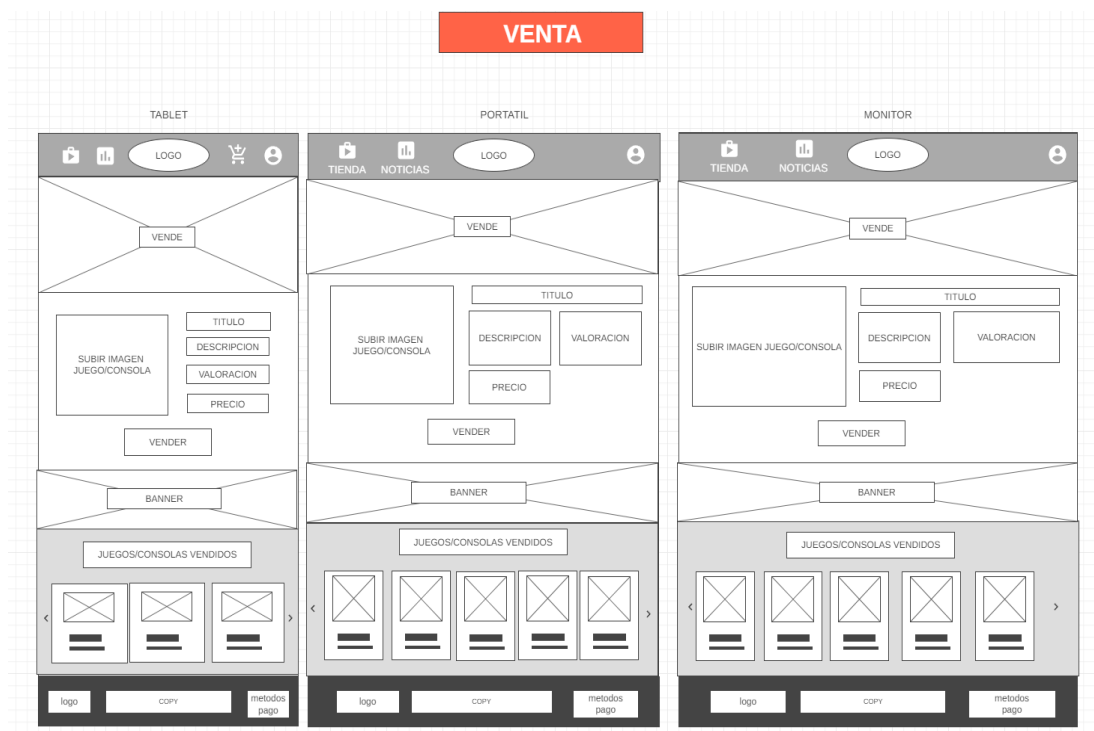
Ventana 2ª Mano (Venta)

En este apartado el objetivo es conseguir que el usuario al introducir el nombre del producto, la descripción, la valoración, la categoría y su precio respectivo junto con la opción de arrastrar la imagen del producto y poder venderlo con la verificación de los campos anteriores. Y se mostrara los productos vendidos por el usuario actual.

Aplicación Android



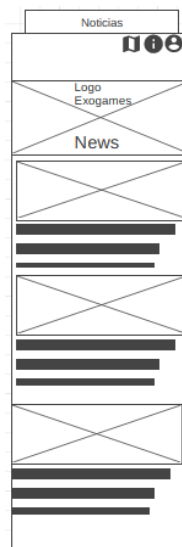
Aplicación Web



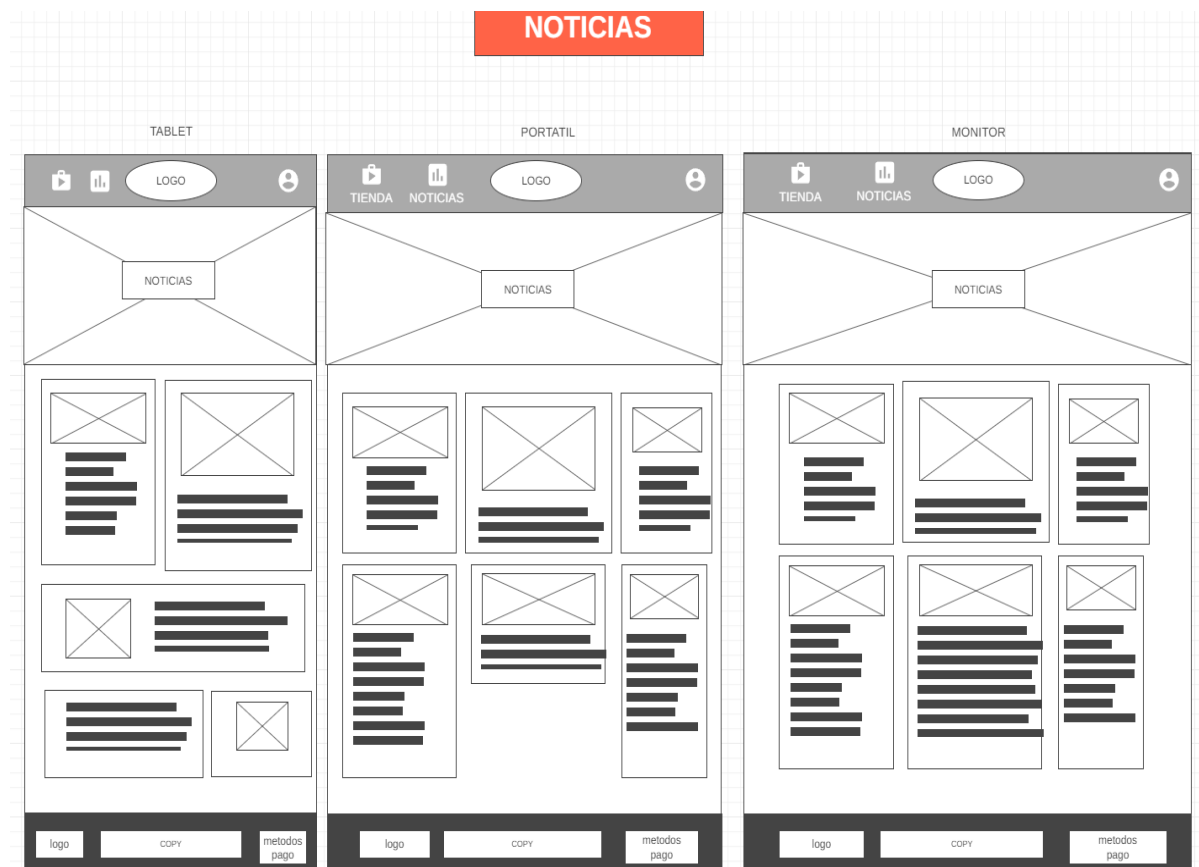
Ventana Noticias

El usuario visualizará una vista con diferentes noticias acerca del mundo de los videojuegos y su amplio sector. Esta ventana solo funcionara a modo de informativo.

Aplicación Android



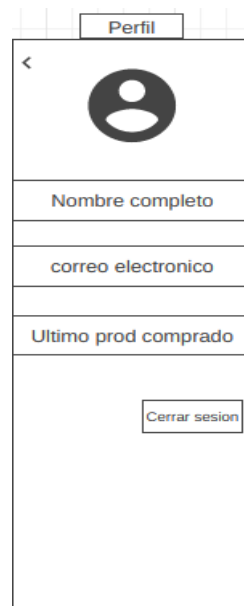
Aplicación Web



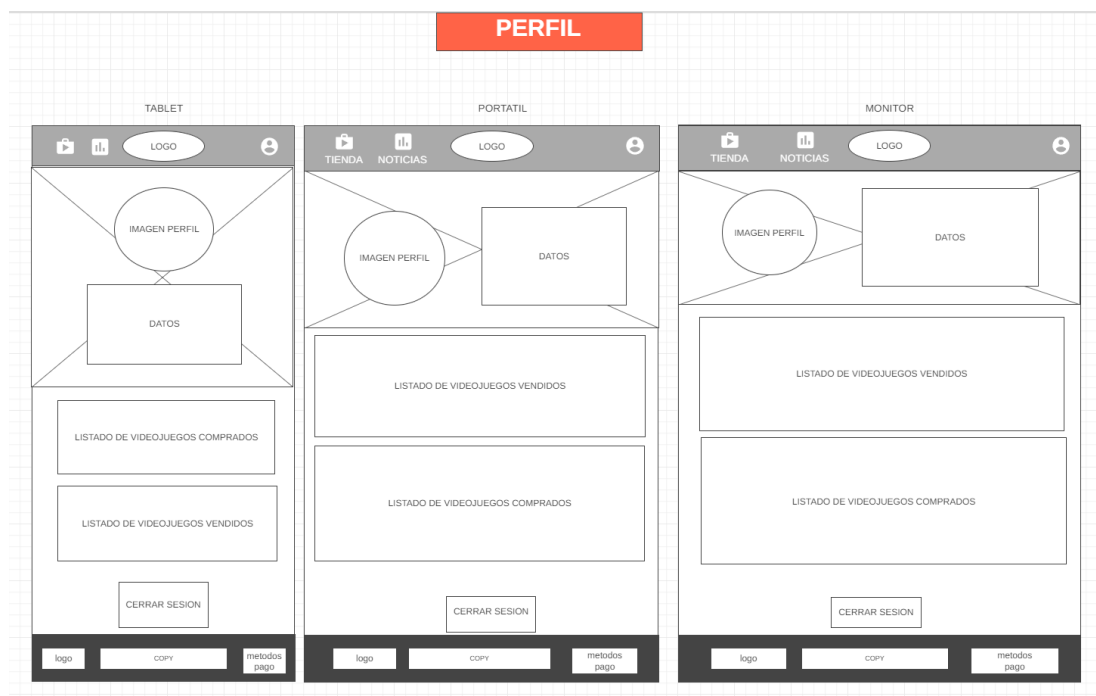
Ventana Perfil

Se mostrará la información del usuario junto con los productos comprados como vendidos/subidos a nuestra aplicación. Junto con la posibilidad de cerrar sesión. A plan de futuro nos gustaría añadir que se pudiera modificar los datos del mismo, pero solo la contraseña como el nombre de usuario.

Aplicación Android



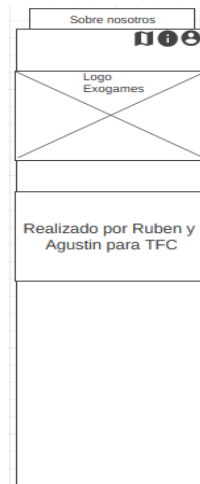
Aplicación Web



Ventana Contacto

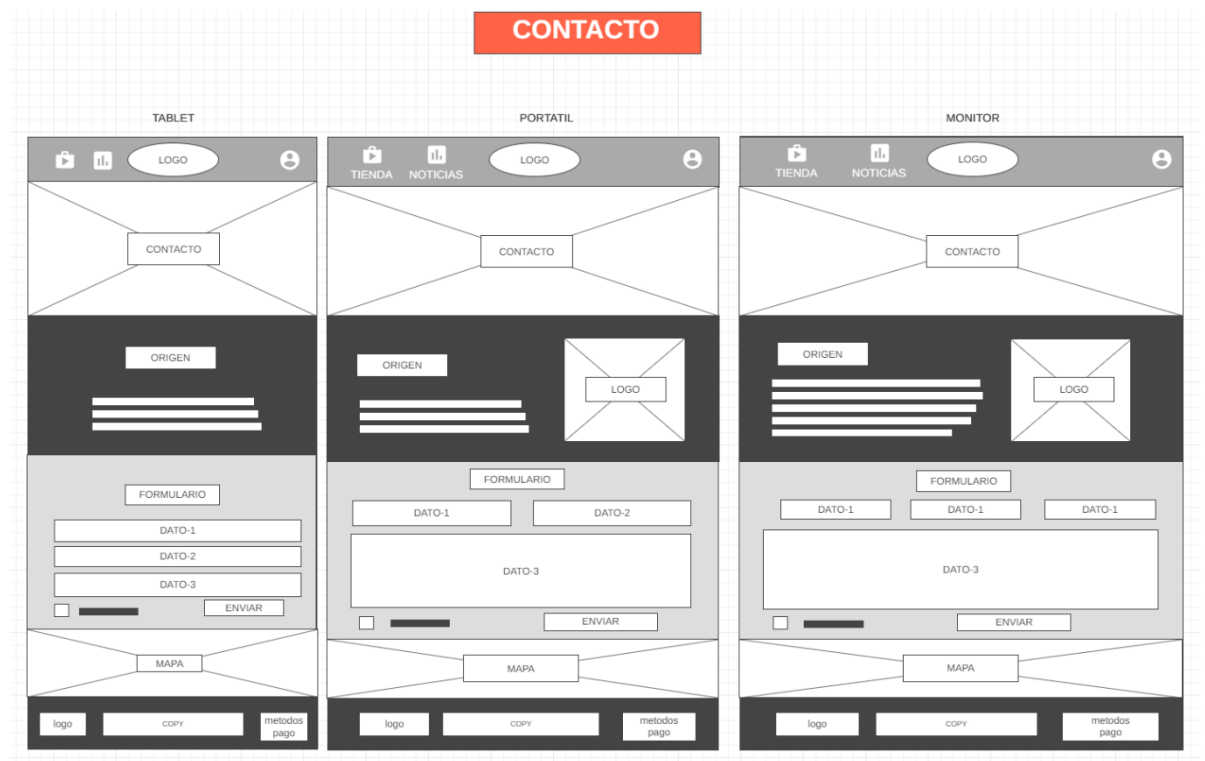
Visualiza la información sobre los fundadores para el desarrollo de la aplicación EXOGAMES. Junto con el logo de nuestro proyecto.

Aplicación Android



Aplicación Web

Se diferencia de la aplicación Android añadiendo un formulario para la consulta de cualquier duda o servicio necesario dentro de la aplicación misma. Y la visualización de la localización de nuestra empresa en un lugar físico.





7 Tecnologías utilizadas

- **JAVA:** Es un lenguaje de programación y una plataforma informática que se ejecuta en la Máquina Virtual Java (JVM) del cual se ha ido popularizando por su facilidad a la hora de portabilidad entre plataformas junto con su diseño innovador. En nuestra app al utilizar Android lo ejecutaremos mediante el SDK proporcionado. Siendo respaldado por Oracle y su licencia es gratuita.
- **GOOGLE FIREBASE:** Trata de una plataforma móvil creada por Google, cuya principal función es desarrollar y facilitar la creación de apps de elevada calidad de una forma rápida y sencilla. Utilizaremos bases de datos alojadas en Google Firebase a modo de sacar, insertar y modificar información de la misma.

La plataforma se dispone tanto en la nube como para diferente plataformas iOS, Android y web.

Firebase nos proporciona una serie de características:

- Integración dinámica de los usuarios usando Firebase Authentication;
- Enviar notificaciones a varias plataformas con Cloud Messaging;
- Actualización automática de cada dato sobre los proyectos creados.
- Crea análisis con sus resultados mediante Google Analytics creación de un servidor para el proyecto.
- Tiene un servicio Hosting propio.
- Soporte gratuito vía email, sin importar si el desarrollador utiliza la versión gratuita o de pago.
- Permite la monetización: desde el propio Firebase se puede agregar publicidad a la app.

- **ANDROID:** Es un sistema operativo móvil desarrollado por Google, basado en Kernel de Linux y otros softwares de código abierto. Siendo la plataforma para smartphones más popular que se ha posicionado como la mejor en el mercado móvil compitiendo con Apple y otros fabricantes de dispositivos hemos decidido emplearla en nuestro proyecto con su facilidad de diseño y amplitud de desarrollo.
- **PHP:** PHP es un lenguaje para programar páginas de contenido dinámico. Este es un lenguaje del lado del servidor, el servidor lee el lenguaje PHP, y manda los datos transformados en lenguaje HTML. Utilidad funcional a la hora de aplicarse en la Aplicación Web.
- **HTML:** El Lenguaje de Marcado de Hipertexto (HTML) es el código que se utiliza para estructurar y desplegar una página web y sus contenidos. Estructuración de nuestra Aplicación Web.
- **CSS:** CSS (en inglés Cascading Style Sheets) es lo que se denomina lenguaje de hojas de estilo en cascada y se usa para estilizar elementos escritos en un lenguaje de marcado como HTML. Es en este tipo de archivos donde reside todo el "aspecto" que tendrá nuestro proyecto. Aplicación de estilos sobre nuestra Aplicación Web y mejorar el diseño.
- **JAVASCRIPT:** Es un lenguaje interpretado en el cliente por el navegador al momento de cargarse la página, es multiplataforma, orientado a eventos con manejo de objetos, cuyo código se incluye directamente en el mismo documento HTML. Lo utilizamos para dar otro tipo de funcionalidad que PHP no administra de la misma forma sobre nuestros ficheros.
- **XAMPP:** Herramienta de desarrollo que te permite probar tu desarrollo web basado en PHP en tu propio ordenador sin necesidad de tener acceso a internet. Nos permitirá visualizar nuestra Aplicación Web de manera local sin necesidad de comprar un espacio de manera virtual y con todos los servicios de gestión configurados.
- **FLUTTER:** Flutter es un framework que nos proporciona un toolkit (conjunto de herramientas) que tienen como finalidad el crear interfaces de software. Flutter fue desarrollado para realizar apps que pudieran ejecutarse tanto en Android como en iOS con rendimiento nativo. Herramienta utilizada para el desarrollo como el diseño sobre nuestra Aplicación Android enlazada a Android Studio y Google Firebase.

Características de Flutter:



1. Fast: Desarrollo con el objetivo de agilizar/mejorar tanto el desarrollo como el rendimiento de la aplicación.
 2. Open: Desarrollo de código abierto provocando que constantemente se genere una gran cantidad de documentación gracias a los aportes de Google
 3. Beautiful: Diseñar aplicaciones con interfaces desde muy sencillas hasta interfaces muy vistosas y expresivas
 4. Productive: Generar aplicaciones multiplataforma (tanto para móviles, desktop y web).
- **LARAVEL:** Se trata de un framework creado para trabajar con PHP creado en el año 2011 por Taylor Otwell, de desarrollo con una curva de aprendizaje muy rápida y que maneja una sintaxis expresiva, elegante, con el objetivo de eliminar la molestia del desarrollo web facilitando las tareas comunes, como la autenticación, enrutamiento, sesiones y caché.

El uso en nuestra Aplicación Web será necesario para la implementación de funcionalidad como diseño de la misma sin necesidad de la estructuración de carpetas utilizando PHP mayormente como lenguaje funcional.

Ventajas:

1. Reducción de costos y tiempos en el desarrollo y posterior mantenimiento de la aplicación creada
 2. Flexible y adaptable no sólo al uso del sistema MVC tradicional
 3. Modular y con un amplio sistema de paquetes
 4. Sencillez a la hora de utilizar los datos mediante Eloquent
 5. Facilita el manejo de las rutas de nuestra aplicación, así como la generación de URLs amigables
 6. Uso del sistema de plantillas Blade
 7. Una gran comunidad y mucha documentación, sobre todo en su sitio oficial
 8. Cuenta con una herramienta de líneas de comando llamada Artisan
- **CANVA:** Se trata de una web de diseño gráfico y composición de imágenes para la comunicación fundada en 2012, y que ofrece herramientas online para crear tus propios diseños, tanto si son para ocio como si son profesionales. Su aplicación en nuestro proyecto será para implementar un diseño previo de como quedaría finalizada la Aplicación Web.

- **FLUTTERFLOW:** Crea la interfaz de usuario de la aplicación Flutter sin escribir una sola línea de código. También tiene una integración de Firebase. En nuestra aplicación Android nos ayudara a realizar el diseño de las ventanas de una forma más rápida y eficaz.
- **FIGMA:** Es un programa de edición de gráficos vectoriales y una aplicación de prototipado que está pensada fundamentalmente para diseñar interfaces, prototipos, wireframes, etcétera. Su aplicación en nuestro proyecto será para implementar un diseño previo de como quedaría finalizada la Aplicación Android.

8 Código documentado

El código documentado se encuentra en el USB administrado de manera presencial al tutor. Todas las funciones tienen una línea a modo de explicación.

```
File Edit Selection View Go Run Terminal Help
web.php - EXOGAMES - Visual Studio Code (Administrator)

EXPLORER
  composer.json
  routes.php
  web.php
  EXOGAMES
    carrito.blade.php
    contacto.blade.php
    dashboard.blade.php
    home.blade.php
    individual.blade.php
    menu.blade.php
    metodo_segunda.blade.php
    metodo.blade.php
    noticias.blade.php
    productos.blade.php
    segunda.blade.php
    tienda.blade.php
    welcome.blade.php
    welcome.blade.php
  routes
  routes.php
  console.php
  storage
  tests
  vendor
  bin
  brick
  composer
  dffdev
  doctrine
  dragonmantank
  egulias
  guzzlehttp
  lucas
  takephp
  filp
  bruticake
  graham-campbell
  hamcrest
  laravel
  laravelcollective
  league

routes.php
25
26
27 //LOGIN Y REGISTRO
28 Auth::routes(); //se almacenan las siguientes rutas:
29 Para loginController
30 GET /login ->Para mostrar el formulario
31 POST /login ->Donde se envia el formulario
32 POST /logout ->Para cerrar sesion
33 Para RegisterController
34 GET /register ->Para mostrar el formulario
35 POST ->Donde se envia el formulario
36
37
38 //RUTAS PARA EL USUARIO
39 // Ruta a la vista del menu/home que mostrara nuestra web al ser usuario
40 Route::get('/',[HomeController::class, 'index'])->name('menu');
41
42 // Ruta a la vista de la pagina de noticias
43 Route::get('noticias',[NoticiaController::class, 'mostrar'])->name('noticias');
44
45 // Ruta a la vista de la pagina de contacto
46 Route::get('contacto',[HomeController::class, 'contacto'])->name('contacto');
47
48 // Ruta con la funcionalidad de enviar un mensaje al administrador mediante el formulario de la pagina contacto
49 Route::post('contacto',[HomeController::class, 'contacto_send'])->name('contacto.send');
50
51 // Ruta a la vista de la pagina de perfil del usuario
52 Route::get('perfil',[UsuarioController::class, 'mostrar_perfil'])->name('perfil');
53
54 // Ruta a la funcion que actualizara el perfil del usuario
55 Route::patch('perfil',[UsuarioController::class, 'update_usuario'])->name('perfil.edit');
56
57 // Ruta a la vista de los pedidos realizados por el usuario
58 Route::get('perfil/pedido/{id}',[UsuarioController::class, 'mostrar_pedido'])->name('perfil.pedido');
59
60 // Ruta a la vista de los productos de segunda mano realizados por el usuario
61 Route::get('perfil/pedido_segunda/{id}',[UsuarioController::class, 'mostrar_pedido_segunda'])->name('perfil.pedido_segunda');
62
63 // Ruta a la vista de la pagina de segunda mano
64 Route::get('segunda',[ProductoController::class, 'mostrar'])->name('segunda');
65
66 // Ruta a la funcion que creara un producto de segunda mano por el usuario
67 Route::post('segunda',[ProductoController::class, 'vendidos'])->name('segunda.create');
68
69 // Ruta a la vista de la pagina de metodo de pago de los productos vendidos
70 Route::get('metodo/segunda',[MetodoController::class, 'segunda'])->name('metodo_segunda');
71
72 // Ruta que actuara como funcionalidad de comprobacion de datos del usuario insertados en el formulario de metodo de pago para los productos de
73 Route::post('metodo/segunda',[MetodoController::class, 'pedido_segunda'])->name('comprobar_segunda');
```



```
// Ruta a la vista de la pagina de segunda mano
Route::get('segunda',[ProductoController::class, 'mostrar'])->name('segunda');

// Ruta a la funcion que crea un producto de segunda mano por el usuario
Route::post('segunda',[ProductoController::class, 'vendidos'])->name('segunda.create');

// Ruta a la vista de la pagina de metodo de pago de los los productos vendidos
Route::get('metodo/segunda',[MetodoController::class, 'segunda'])->name('metodo.segunda');

// Ruta que actuara como funcionalidad de comprobacion de datos del usuario insertados en el formulario de metodo de pago para los productos de segunda mano
Route::post('metodo/segunda',[MetodoController::class, 'pedido_segunda'])->name('comprobar.segunda');

// Ruta a la vista de la pagina de tienda de productos
Route::get('tienda',[HomeController::class, 'tienda'])->name('tienda');

// Ruta a la vista de la pagina del producto seleccionado mas detalladamente con su demo
Route::get('tienda/{id}',[HomeController::class, 'producto_individual'])->name('individual');

// Ruta a la vista de la pagina de carrito de productos
Route::get('carrito',[CarritoController::class, 'index'])->name('carrito');

// Ruta a la vista de la pagina de carrito de productos
Route::post('tienda/carrito',[CarritoController::class, 'agregar_carrito'])->name('agregar.carrito');

// Ruta que llamara a la funcion de aumento de cantidad de un producto seleccionado en el carrito y comprobacion de stock
Route::post('carrito/aumentar/{id}',[CarritoController::class, 'aumentar_cantidad'])->name('aumentar.carrito');

// Ruta que llamara a la funcion de reducir la cantidad de un producto seleccionado en el carrito y comprobacion de stock
Route::post('carrito/reducir/{id}',[CarritoController::class, 'reducir_cantidad'])->name('reducir.carrito');

// Ruta que llamara a la funcion de eliminar el producto seleccionado y eliminadnolo del carrito
Route::delete('tienda/carrito/eliminar/{id}',[CarritoController::class, 'quitar_carrito'])->name('quitar.carrito');

// Ruta a la vista de la pagina de metodo de pago despues de finalizar el carrito
Route::get('metodo',[MetodoController::class, 'index'])->name('metodo');

// Ruta que actuara como funcionalidad de comprobacion de datos del usuario insertados en el formulario de metodo de pago
Route::post('metodo',[MetodoController::class, 'pedido'])->name('comprobar');

//RUTAS PARA EL ADMIN

// Ruta a la vista del panel del admin
Route::get('dashboard',[AdministradorController::class, 'dash'])->name('dashboard');

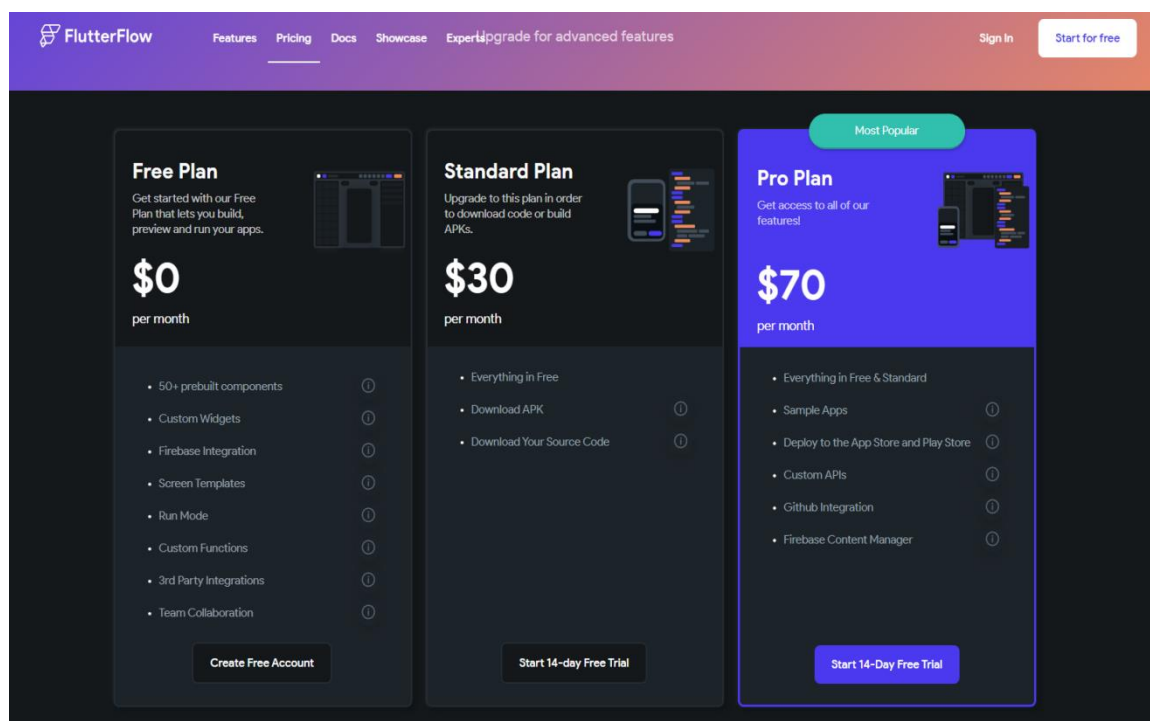
//Ruta a la vista del listado de monedas recibidos que el formulario de compras mostrados al admin
```

```
composer.json HomeController.php 2 x
app > Http > Controllers > HomeController.php > ...
6 use App\Models\Comentario;
7 use App\Models\Contacto;
8 use App\Models\Carrito;
9 use App\Models\Filtros;
10 use App\Models\Imagenes;
11 use App\Models\Producto;
12
13 use Illuminate\Http\Request;
14 use Illuminate\Support\Facades\Auth;
15
16 class HomeController extends Controller
17 {
18
19
20
21
22 public function index()//Comprobacion de los permisos de usuarios si es admin te lleva al dashbord y si es cliente te muestra la ventana de menu
23 {
24     if (Auth::check()) {
25         if (auth()->user()->rol === "cliente") {
26             return view('menu');
27         } else {
28             return redirect()->route('dashboard');
29         }
30     } else {
31
32     }
33
34     return view('menu');
35 }
36 }
37
38
39 public function tienda(Request $request)//Muestra los productos por filtrado con la opcion de comprarlos
40 // y seleccion de categoria si es consola o videojuego //si el usuario no ha iniciado sesion solo mostrara el listado con los filtros y las categorias sin la opcion de compra
41 {
42
43     if ($request) {
44         if (Auth::check()) {
45             if (auth()->user()->rol === "cliente") {
46                 $carrito = Carrito::join('productos as p', 'carrito.idproducto', '=', 'p.id')->select('carrito.cantidad', 'p.imagen', 'p.nombre', 'p.precio', 'carrito.id')->where('id'
47                 ->limit(2)->get();
48
49                 $carritototal = Carrito::where('idusuario', '=', auth()->user()->id)->get();
50
51                 $compras = count($carritototal);
```

9 Presupuesto

FlutterFlow dispone de una serie de tarifas con sus ventajas respectivamente del cual preferimos por apostar el plan Standard por 30\$/mes para la obtención de los servicios gratuitos del plan Free junto con la compilación del APK. También posee una gran ventaja para la obtención del código fuente de nuestro proyecto para mayor facilidad de portabilidad.

Decidimos no utilizar la versión trial debido a problemas de exportación de proyecto cuando llegue la fecha de pago mensual y así ahorrar tiempo y futuros posibles problemas.



En cuanto al presupuesto, disponíamos de un crédito que proporciona Google para proyectos de este tipo de 300 USD (273,48€) para poder realizar las peticiones a Google. Gracias a la nube de Firebase nos facilita el uso de restricciones para poder saber quién tienes permiso tanto de leer como editar la información, saber el perfil de registro actual de cada usuario, monitorización desde cualquier dispositivo portátil como de escritorio. Elegimos el Plan Spark para un comienzo de la aplicación, pero con mentalidad de uso constante a gran escala a futuro la mejor solución a elegir sería el Plan Blaze. Mostramos la tabla comparativa de los planes definidos anteriormente.



Tarifas Firebase

Spark	Blaze
Gratis USD 0 por mes	Pago por uso
<ul style="list-style-type: none">Cuotas de uso de Database, Firestore, Storage, Functions, Phone Auth, Hosting y Test LabCapacidad para extender tu proyecto con Google Cloud Platform	<ul style="list-style-type: none">Incluye el uso gratuito que se calcula por día. Después, pagas solo por lo que use tu proyecto.Capacidad para extender tu proyecto con Google Cloud Platform
✓ Se incluyen en todos los planes Analytics, notificaciones, informes de fallos, asistencia y más	✓ Se incluyen en todos los planes Analytics, notificaciones, informes de fallos, asistencia y más
Ver los detalles del plan completo	Ver los detalles del plan completo
Plan actual	Seleccionar plan

Authentication		
Autenticación telefónica: Canadá, EE.UU. y la India	10,000 por mes	USD 0.01 por verificación
Autenticación telefónica: Todos los demás países	10,000 por mes	USD 0.06 por verificación
Otros servicios de autenticación	✓	✓

Gracias al utilizar el plan Spark de Google Firebase podemos disponer de un uso de nuestro presupuesto reducido para el desarrollo de nuestra Aplicación Android.

Como idea a futuro a la hora de publicar nuestra aplicación Android, utilizaríamos el plan Pro ya que nos facilitaría la integración de GitHub, junto al despliegue de la aplicación sobre la tienda Play Store disponible en los dispositivos Android.

Como plan a futuro preferiríamos alojar nuestro proyecto de la Aplicación Web en un hosting online del cual poder utilizar el espacio que dispone sin problemas de

instalación, pero debido a las licencias que poseen nuestras emulaciones/demo de ciertos videojuegos no creemos posible su desarrollo.

Si nos planteáramos la idea de utilizar un hosting con su dominio nos decantaríamos por la web de Hostinger. Posee numerosas tarifas con muchas facilidades para no configurar desde cero el espacio.

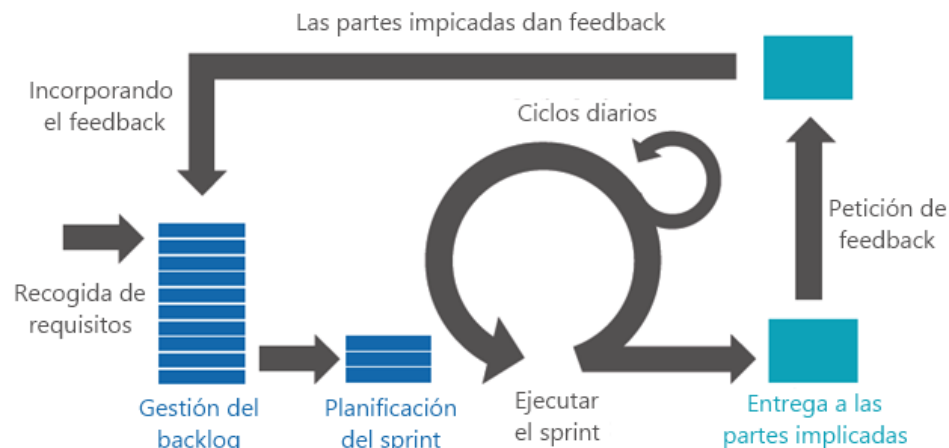
Elige Tu Plan De Web Hosting

Hosting Sencillo	MÁS VENDIDO Hosting Premium	Hosting Empresarial
La solución ideal para principiantes	El paquete perfecto para webs personales	Optimizado para negocios pequeños y medianos
3,29€ AHORRA 70%	10,19€ AHORRA 80%	15,99€ AHORRA 75%
0,99€/mes	1,99€/mes	3,99€/mes
Añadir al carro	Añadir al carro	Añadir al carro
2,99 €/mes al renovar	4,99 €/mes al renovar	7,99 €/mes al renovar
<ul style="list-style-type: none"> ✓ 1 Sitio web ✓ 30 GB de Almacenamiento SSD ✓ ~10000 Visitas Mensuales ✓ 1 Cuenta de correo ✓ SSL Incluido (valor: 11,95 €) ✗ Dominio gratis ✓ 100 GB Ancho de banda ✓ WordPress Administrado ✓ Aceleración de WordPress ✗ Herramienta Staging de WordPress ✓ Garantía de 30 DÍAS de Devolución del Dinero ✓ 2 Bases de Datos ✓ Acceso GIT 	<ul style="list-style-type: none"> ✓ 100 Sitios web ✓ 100 GB de Almacenamiento SSD ✓ ~25000 Visitas Mensuales ✓ Email Gratis ✓ SSL Incluido (valor: 11,95 €) ✓ Dominio Incluido (valor: 8,99 €) ✓ Ancho de Banda ilimitado ✓ WordPress Administrado ✓ Aceleración de WordPress ✗ Herramienta Staging de WordPress ✓ Garantía de 30 DÍAS de Devolución del Dinero ✓ Bases de datos ilimitadas ✓ Acceso GIT ✓ Acceso SSH 	<ul style="list-style-type: none"> ✓ 100 Sitios web ✓ 200 GB de Almacenamiento SSD ✓ ~100000 Visitas Mensuales ✓ Email Gratis ✓ SSL Incluido (valor: 11,95 €) ✓ Dominio Incluido (valor: 8,99 €) ✓ Ancho de Banda ilimitado ✓ WordPress Administrado ✓ Aceleración de WordPress ✓ Herramienta Staging de WordPress ✓ Garantía de 30 DÍAS de Devolución del Dinero ✓ Bases de datos ilimitadas ✓ Acceso GIT ✓ Acceso SSH ✓ Copias de seguridad Diarias (valor: 25,08 €)
Ver todas las funciones	Ver todas las funciones	Ver todas las funciones

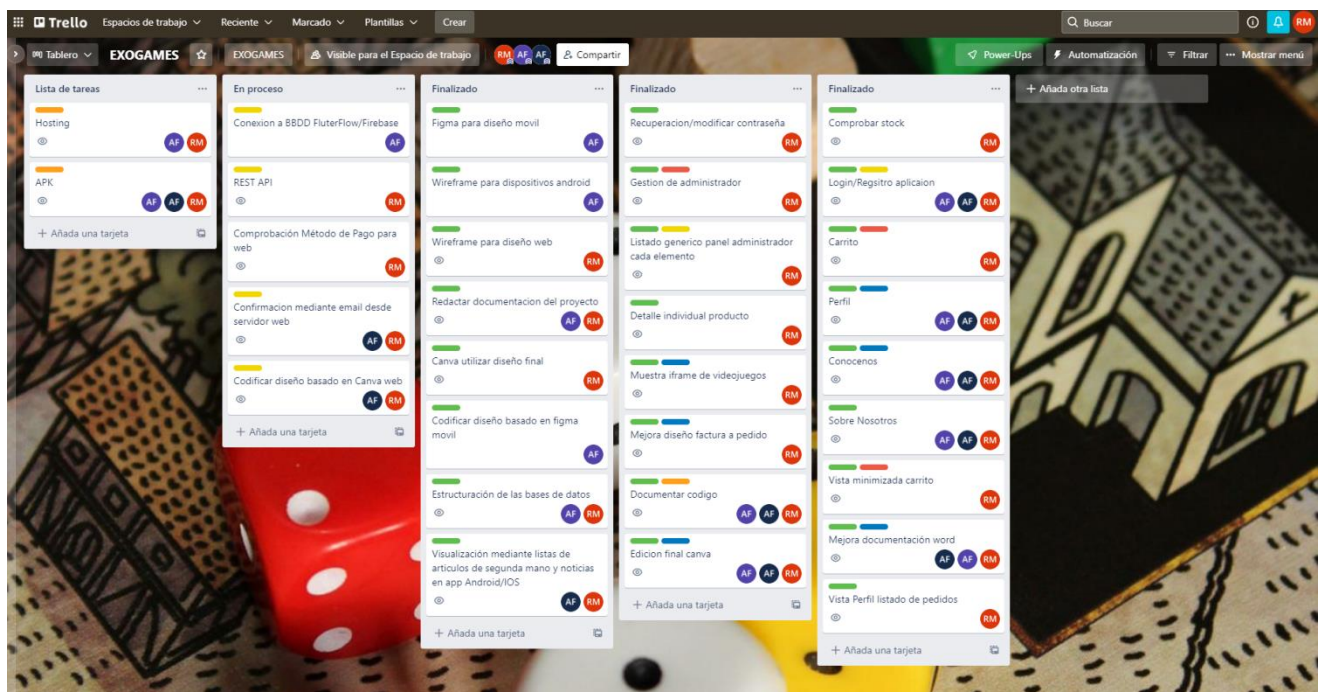
Elegiríamos el plan Hosting Premium debido a su gran capacidad de almacenamiento como la posibilidad de añadir numerosos correos electrónicos para poder acceder, incluyendo el dominio elegido sin la necesidad de buscarlo en otro sitio y la capacidad de guardar un mayor número de sitios webs.

10 Planificación

Elegimos la metodología ágil SCRUM por su gran efectividad a la hora de llevar a cabo el proyecto. Al ser un proyecto con una cantidad mínima de equipo para poder realizarlo, solo necesitábamos 2 días por semana para hacer uso del Sprint. Estas reuniones las realizábamos mediante la herramienta llamada Discord.



Para tener una mejor organización del proyecto y saber los pasos a seguir utilizamos la herramienta Trello, herramienta de gestión de proyectos que hace que la colaboración sea sencilla y visible organizándolo en columnas. Este software de administración de proyectos basada en el método Kanban, método para gestionar el trabajo usando el famoso *To Do*, *Doing* y *Done*, y sirve para gestionar tareas permitiendo organizar el trabajo en grupo de forma colaborativa mediante tableros virtuales compuestos de listas de tareas en forma de columnas.



Aún está en desarrollo debido a la utilización de la herramienta Discord en mayor medida, iremos añadiendo tareas a Trello según avancen los días.

11 Manual de instalación

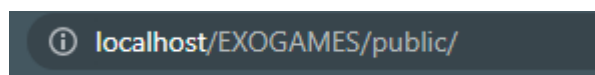
Su instalación en el dispositivo móvil se realizará a través del APK que será distribuido a través de plataformas o webs de aplicaciones.

Una idea a futuro que deseamos realizar es alojar de nuestro proyecto en un hosting de manera virtual. Pero debido a problemas de derechos por la emulación de nuestros videojuegos no es recomendable su alojamiento. Por lo cual utilizaremos XAMPP para su uso en un servidor local.

Anteriormente será necesario instalar Composer junto con Laravel. Gracias a esta guía explicativa paso por paso en sistemas Windows 10:

<https://cynoteck.com/es/blog-post/installing-laravel-8-on-windows-10-xampp/#:~:text=Extensi%C3%B3n%20PHP%20XML%2C%20Instalando%20Laravel%20A,tambi%C3%A9n%20generar%20la%20clave%20ANSI.>

Para poder iniciar nuestra web será necesario desplegar los servicios de Apache y MySQL que ofrece XAMPP, a continuación, dirigirnos a nuestro navegador predeterminado e introducir en la barra de búsqueda la siguiente instrucción mostrada en la imagen. Siendo la dirección localhost/ continuada del nombre del directorio de nuestro proyecto EXOGAMES/ seguido del directorio public/ que almacena nuestra vista de la web.



12 Manual de usuario

El fichero documentado se encuentra subido a Git Hub, adjuntamos su link:

<https://github.com/rubxter99/EXOGAMES/blob/f0571d1d066a704b25c2f86c6c6747ae4f8aff44/MemoriaTFC/ManualUsuarioRubenGarciaAgustinJimenez.docx>

13 Conclusiones

Hemos decidido plantear nuestro proyecto en base al sector de los videojuegos debido a su constante crecimiento ocupando rangos de edad muy variados no limitados al público joven. También vimos que se podía sacar provecho sacando una aplicación estilo ecommerce, obteniendo beneficios por parte de las aplicaciones como Android y Web. Atrayendo clientes gracias a nuestro servicio de compra/venta.

Nuestro mayor inconveniente por parte de la aplicación Web fue la gestión de pedidos para comprobar su estado, ya sea: entregado, disponible, en curso y cancelado; como también su método de compra.

Nuestro mayor inconveniente por parte de la Aplicación Android es que las tiendas físicas de videojuegos, están en una base de datos local la cual habría que actualizar para poder vender a nivel nacional, pues solo se encuentran tiendas en la comunidad de Madrid.

La idea inicial de nuestro proyecto fue complicada de procesar a nuestro número de miembros para el desarrollo de este proyecto, pero al final planificando reuniones con nuestro tutor Aníbal Martín conseguimos mejorarla.

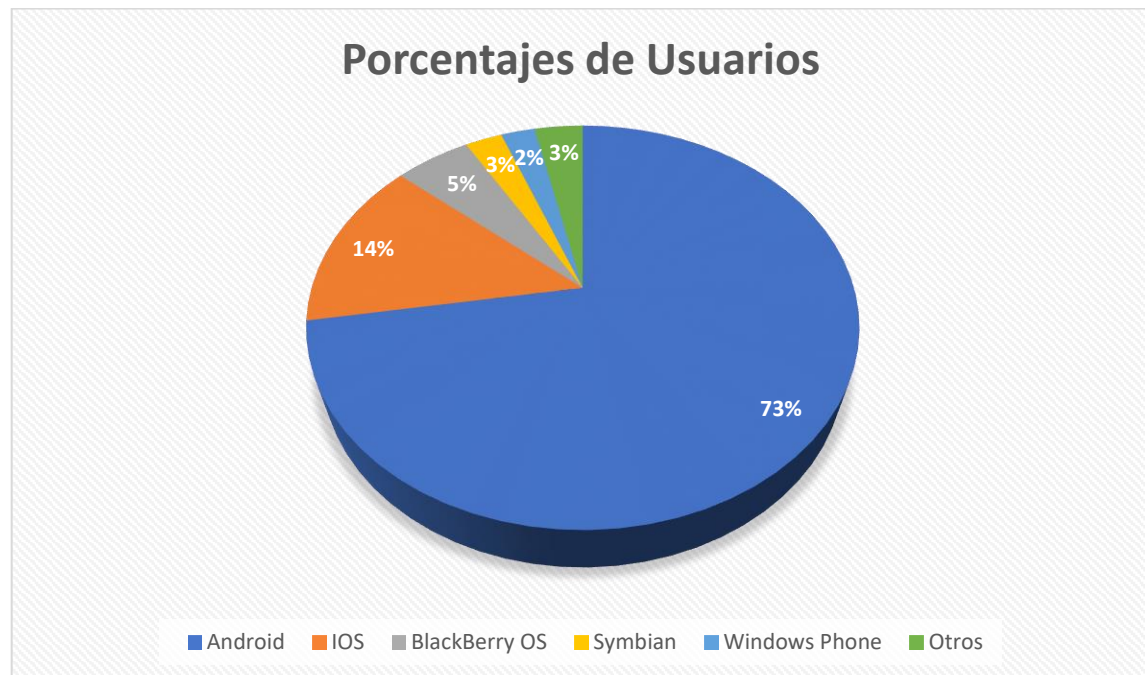
Al hacer reuniones constantes no hemos tenido problemas en el diseño ni en el desarrollo de nuestro proyecto. Ha sido una experiencia gratificante al utilizar nuevas tecnologías.

14 Análisis de la competencia y de los clientes

Comparativa Principales Mercados en el Sector de los Videojuegos Global (2021-2025)



Comparativa Sistemas Operativos para Dispositivos Móviles



Comparativa en escala de las Plataformas en el Sector de Videojuegos (2020)

Evolución del mercado de los videojuegos y sus plataformas

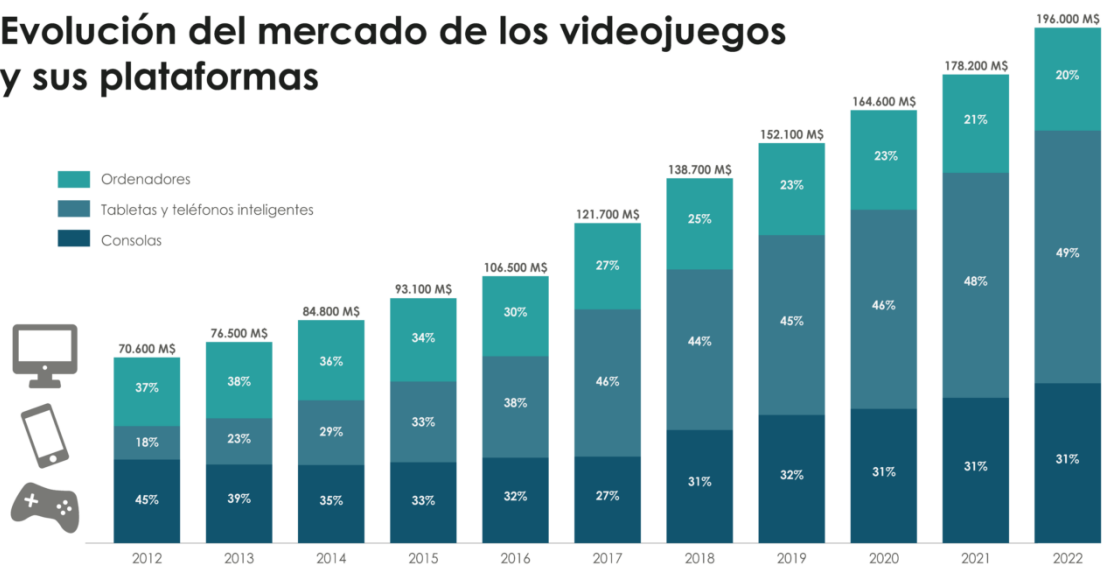


Gráfico:
Abel Gil Lobo (2020)
Fuente:
Newzoo (2019)

EOM
elordenmundial.com

Comparativa entre Sectores del Mercado (2017-2021)



Comparativa entre el uso de versiones Android

Android Platform/API Version Distribution

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99,8%
4.2 Jelly Bean	17	99,2%
4.3 Jelly Bean	18	98,4%
4.4 KitKat	19	98,1%
5.0 Lollipop	21	94,1%
5.1 Lollipop	22	92,3%
6.0 Marshmallow	23	84,9%
7.0 Nougat	24	73,7%
7.1 Nougat	25	66,2%
8.0 Oreo	26	60,8%
8.1 Oreo	27	53,5%
9.0 Pie	28	39,5%
10 Android 10	29	8,2%

En nuestro caso comenzamos la compatibilidad a partir de la versión 6.0 (Marshmallow), debido a su alto porcentaje de uso por usuario.

15 Bibliografía

<https://firebase.google.com/docs/auth/android/custom-auth?hl=es#java>

<https://api.rawg.io/docs/>

<https://jsoneditoronline.org/>

<https://stackoverflow.com/>

<https://stackoverflowrun.com/>

<https://flutterflow.io/>

<https://www.canva.com/>

<https://www.figma.com/>

<https://trello.com/>

<https://wireframe.cc/>

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

<https://github.com/>

<https://www.nigmacode.com/laravel/subir-proyecto-laravel-a-hosting/>

<https://es.stackoverflow.com/questions/125755/ventana-emergente-mensaje-en-php>

<https://profile.es/blog/que-es-flutter-sdk/>