

TRABALLO FIN DE GRAO  
GRAO EN ENXEÑARÍA INFORMÁTICA  
MENCIÓN EN ENXEÑARÍA DO SOFTWARE



# **Desarrollo de una Aplicación Multiplataforma Escalable y Plataforma Analítica Complementaria para la Recopilación y Análisis de Datos de Uso**

**Estudiante:** Francisco Lareo García

**Dirección:** Outro Nome Completo

A Coruña, agosto de 2024.

*Dedicatoria*

### **Agradecementos**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

## Resumo

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

## Abstract

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

### Palabras chave:

- First itemtext
- Second itemtext
- Last itemtext
- First itemtext
- Second itemtext
- Last itemtext
- First itemtext

### Keywords:

- First itemtext
- Second itemtext
- Last itemtext
- First itemtext
- Second itemtext
- Last itemtext
- First itemtext

# Índice Xeral

---

<b>1</b>	<b>Introdución</b>	<b>1</b>
1.1	Contexto y motivación . . . . .	1
1.2	Objetivo del trabajo . . . . .	2
1.2.1	Desarrollo de una aplicación OTT multiplataforma y multicliente . . .	2
1.2.2	Desarrollo e integración de una aplicación para la recogida y visuali- zación de datos de uso . . . . .	2
<b>2</b>	<b>Fundamentos</b>	<b>3</b>
2.1	Introducción . . . . .	3
2.2	Fundamentos teóricos . . . . .	3
2.2.1	OTT(Over-the-Top) . . . . .	3
2.2.2	Arquitectura . . . . .	5
2.2.3	Experiencia de usuario (UX) . . . . .	7
2.2.4	Escalabilidad, adaptabilidad y multiplataforma . . . . .	8
2.3	Fundamentos tecnológicos . . . . .	10
2.3.1	Arquitectura Software: Comunicación entre componentes y desarro- llo multicliente . . . . .	10
2.3.2	Tecnologías Web: JavaScript, HTML y CSS . . . . .	11
2.3.3	Desarrollo multiplataforma . . . . .	11
<b>3</b>	<b>Metodología</b>	<b>14</b>
3.1	Introducción . . . . .	14
3.1.1	Diseño del proyecto . . . . .	14
3.2	Metodología de Desarrollo . . . . .	15
3.2.1	Implementación progresiva e incremental . . . . .	15
3.2.2	Adaptación continua y flexibilidad . . . . .	16
3.2.3	Reuniones de Validación y Planificación . . . . .	16
3.2.4	Ventajas de la metodología ágil . . . . .	16

3.3	Descripción de las Etapas de Desarrollo . . . . .	17
3.3.1	Fase de Análisis y Planificación . . . . .	17
3.3.2	Fase de Diseño . . . . .	17
3.3.3	Fase de Desarrollo . . . . .	18
3.3.4	Fase de Pruebas . . . . .	18
3.3.5	Fase de Despliegue y Mantenimiento . . . . .	18
<b>4</b>	<b>Análisis</b>	<b>20</b>
4.1	Introducción . . . . .	20
4.2	Requisitos . . . . .	20
4.2.1	Requisitos de la plataforma OTT . . . . .	21
4.3	Necesidades del cliente . . . . .	24
4.4	Estudio de viabilidad . . . . .	25
4.4.1	Viabilidad técnica . . . . .	25
4.4.2	Viabilidad económica . . . . .	27
4.4.3	Conclusión final . . . . .	29
4.5	Estudio de las características de los distintos sistemas operativos, dispositivos y tecnologías . . . . .	29
4.5.1	Sistemas operativos . . . . .	29
4.5.2	Dispositivos . . . . .	31
4.5.3	Tecnologías . . . . .	32
<b>5</b>	<b>Diseño</b>	<b>33</b>
5.0.1	Introducción . . . . .	33
<b>6</b>	<b>Conclusiones</b>	<b>34</b>
<b>A</b>	<b>Material adicional</b>	<b>37</b>
	<b>Bibliografía</b>	<b>40</b>

# Índice de Figuras

---

## Índice de Táboas

---



# Introducción

---

## 1.1 Contexto y motivación

¿Netflix? ¿HBO? ¿DAZN? ¿Amazon Prime Video? ¿Disney+? Me atrevería a afirmar que (casi) todos los lectores de este documento tienen o han tenido acceso a alguna de estas aplicaciones en los últimos meses. Diría también que son unas de las principales aplicaciones en las que uno piensa a la hora de descargar alguna aplicación en alguno de sus dispositivos. En muchos de estos dispositivos hasta vienen preinstaladas. Según un estudio realizado por la Comisión Nacional de los Mercados y la Competencia (CNMC) el 58% de los hogares españoles con acceso a Internet usaba algún servicio de vídeo en streaming a mediados del año 2023, frente al 37% de mediados de 2019. [1]. Otras fuentes sitúan este porcentaje en un 81% a principios de 2023 [2] e incluso en un 95% en junio de 2024 [3]. Estas cifras varían un poco dependiendo de la fuente, pero todas coinciden en algo: el consumo de contenido audiovisual a través de plataformas de streaming está en auge y ya supera a las plataformas televisivas tradicionales. Leyendo estos estudios y viendo la evolución de este mercado se entiende el por qué de la constante aparición de nuevas plataformas de este estilo. Desde películas y series hasta deportes, pasando por documentales, cursos, conciertos, etc. Estas plataformas se adaptan a cualquier estilo de contenido y a cualquier tipo de usuario.

No es únicamente el número de usuarios el que ha aumentado, sino también el tiempo que pasamos viendo contenido en estas plataformas, el número de plataformas distintas a las que accedemos y el número de dispositivos en los que lo hacemos. La causa de esto es la facilidad de acceso a estas plataformas desde cualquier lugar y dispositivo. Y es que, ¿quién no ha empezado a ver una serie en la televisión del salón, ha continuado en la tablet de la cocina y ha terminado en el móvil de la cama? ¿O quién no ha parado la serie para cambiar de plataforma y ver un partido de fútbol? El consumo de contenido audiovisual nunca había sido tan sencillo y accesible.

Sencillo para el consumidor claro, pero para llegar a esta versatilidad y facilidad de uso

hay un gran trabajo detrás para lograr que utilizar estas plataformas sea igual sin importar el tamaño, la marca o el sistema operativo del dispositivo ni el lugar en el que se encuentre el usuario.

## **1.2 Objetivo del trabajo**

### **1.2.1 Desarrollo de una aplicación OTT multiplataforma y multicliente**

Aquí surge el objetivo de este trabajo: desarrollar una aplicación multiplataforma para la visualización de contenido audiovisual en streaming.

En este trabajo se va a abordar todas las fases de desarrollo de una aplicación OTT (Over The Top), con dos características principales: multiplataforma y multicliente. Multiplataforma porque la aplicación esta orientada a poder adaptarse a cualquier dispositivo y sistema operativo. Multicliente porque el código no está pensado únicamente para un cliente en concreto, sino que está pensado para poder ser utilizado por cualquier cliente que quiera tener su propia plataforma donde mostrar su contenido.

Estos enfoques suponen varios retos a nivel de desarrollo, los principales: adaptación a las distintas necesidades de cada dispositivo donde se quiera dar soporte a la aplicación, con las características y limitaciones que conlleva utilizar las distintas plataformas, tecnologías y sistemas operativos; y la adaptación a las distintas necesidades de cada cliente, tratando de mantener un equilibrio entre la personalización de la aplicación y la reutilización del código.

### **1.2.2 Desarrollo e integración de una aplicación para la recogida y visualización de datos de uso**

Cuando se desarrollan aplicaciones comerciales siempre hay un objetivo claro: atraer a los usuarios y mantenerlos el mayor tiempo posible en la aplicación. Para ello, es necesario conocer a los usuarios, saber qué les gusta, qué no les gusta, qué les interesa, etc. Por ello, las empresas cada vez más están invirtiendo en herramientas de analítica para poder recoger y analizar los datos de uso de sus aplicaciones.

Complementariamente a la aplicación OTT, en este trabajo se va a abordar el desarrollo de una aplicación de analítica de datos de uso de la aplicación OTT. Esta aplicación recogerá datos que nos permitirá conocer el comportamiento de los usuarios en la aplicación, para poder mejorar la experiencia de usuario y adaptar la aplicación a las necesidades de los distintos usuarios de las distintas plataformas.

Este desarrollo incluye el análisis y seleccion de las métricas necesarias para conocer el comportamiento de los usuarios en este tipo de plataformas, la integración de la API de analítica en la OTT y el desarrollo de una aplicación de visualización de los datos recogidos.

# Fundamentos

---

## 2.1 Introducción

El desarrollo de aplicaciones multiplataforma representa un gran desafío técnico en un entorno tan complejo como el tecnológico actualmente, con una variedad tan amplia de dispositivos y plataformas. Para asegurar el éxito de un proyecto de estas características, es fundamental comprender tanto los principios teóricos que permiten crear aplicaciones flexibles y escalables como los fundamentos tecnológicos empleados en su desarrollo.

En este capítulo se presentan los fundamentos teóricos y tecnológicos necesarios en el desarrollo de las aplicaciones propuestas en este trabajo. Se explorarán conceptos teóricos clave como el funcionamiento de aplicaciones OTT, la experiencia de usuario, el diseño de interfaces, la adaptabilidad y la escalabilidad; así como las tecnologías esenciales para desarrollar una aplicación que se ajuste tanto a las necesidades del cliente como a la plataforma, e integrar una plataforma analítica complementaria.

## 2.2 Fundamentos teóricos

### 2.2.1 OTT(Over-the-Top)

#### **Definición, funcionamiento y clasificación de los servicios OTT**

En radiodifusión, un servicio OTT (siglas en inglés de over-the-top) o servicio de transmisión libre consiste en la transmisión de audio, vídeo y otros contenidos a través de internet sin la implicación de los operadores tradicionales en el control o la distribución del contenido [4].

Estos servicios han transformado radicalmente la forma en que los usuarios consumen contenido audiovisual y cómo los proveedores lo distribuyen. La clave de esta transformación radica en la capacidad de acceder a contenidos de manera inmediata, sin la necesidad de

esperar a que se emitan en televisión, radio, cine u otros medios tradicionales. Además, los servicios OTT permiten a los usuarios disfrutar de contenidos en cualquier lugar y momento, siempre que cuenten con una conexión a internet.

En cuanto al funcionamiento de los servicios OTT, estos se basan en un catálogo de contenidos que los usuarios pueden explorar según sus preferencias. Cada catálogo incluye una variedad de películas, series, documentales, y otros tipos de contenido, presentados con descripciones detalladas, géneros, calificaciones de usuarios, y recomendaciones basadas en el historial de visualización. Los usuarios pueden navegar por el catálogo, seleccionar el contenido que desean ver y comenzar la transmisión de manera inmediata, beneficiándose de la capacidad de pausar, rebobinar o adelantar el contenido según su conveniencia. Este sistema flexible no solo permite a los usuarios tener control total sobre lo que ven y cuándo lo ven, sino que también facilita la entrega personalizada de contenido, adaptándose continuamente a los gustos individuales de cada usuario.

El tipo de contenido ofrecido por los servicios OTT es muy variado, abarcando desde películas y series hasta música, deportes, noticias, cursos educativos y recursos generados por los usuarios. Esta diversidad permite que los servicios OTT se adapten a diferentes preferencias y necesidades, ofreciendo una experiencia personalizada para cada tipo de usuario. Además, el acceso a datos masivos sobre los hábitos de consumo de los usuarios ha permitido a las plataformas OTT optimizar sus catálogos y estrategias de marketing, logrando una conexión más profunda y relevante con su audiencia.

En función del modelo de negocio y la forma en que se ofrece el contenido, los principales tipos de servicios OTT son:

- SVOD (Subscription Video on Demand): Servicios que ofrecen acceso ilimitado a un catálogo de contenido a cambio de una suscripción mensual o anual.
- AVOD (Advertising Video on Demand): Servicios que proporcionan contenido gratuito a los usuarios a cambio de la visualización de anuncios publicitarios.
- TVOD (Transactional Video on Demand): Servicios que permiten a los usuarios alquilar o comprar contenido de manera individual.

### **Evolución y tendencias de los servicios OTT**

El crecimiento de los servicios OTT ha sido impulsado por varios factores clave, como la mejora de las conexiones a internet, que ha permitido una transmisión más rápida y de mayor calidad, y los avances tecnológicos en compresión de video y optimización de redes, que han mejorado significativamente la experiencia del usuario. Estas aplicaciones deben ser diseñadas para escalar eficientemente, manejando grandes volúmenes de tráfico durante eventos en vivo o lanzamientos populares, lo que plantea desafíos en términos de rendimiento. Además,

la personalización se ha convertido en un diferenciador esencial, impulsada por algoritmos de inteligencia artificial que analizan los patrones de consumo para ofrecer contenido relevante, mejorando la experiencia del usuario y fomentando la lealtad a la plataforma. La proliferación de dispositivos móviles y la preferencia por el contenido instantáneo han consolidado a los servicios OTT como la opción preferida frente a los medios tradicionales, reflejando una evolución en los hábitos de consumo hacia la conveniencia y la personalización.

Además de estos factores, el entorno competitivo también ha impulsado innovaciones en la oferta de servicios OTT. Las empresas han tenido que diferenciarse mediante la creación de contenido original exclusivo, el uso de inteligencia artificial para mejorar la personalización y la implementación de modelos de negocio híbridos que combinan suscripciones, publicidad y transacciones para maximizar el alcance y la rentabilidad. Estas estrategias no solo buscan atraer y retener a los usuarios, sino también garantizar la sostenibilidad del servicio en un mercado cada vez más fragmentado.

### 2.2.2 Arquitectura

Se entiende Arquitectura Software como los modelos y estándares que sirven de base para el diseño e implementación de sistemas de software. Indica la estructura, funcionamiento e interacción de los componentes de software [5].

El diseño de la arquitectura de un sistema de software para una aplicación OTT es clave para garantizar el correcto funcionamiento de la misma y la entrega eficiente, escalable y confiable de los contenidos. Existen varias posibilidades para el diseño de la arquitectura de una aplicación de estas características como la Arquitectura Monolítica, la Arquitectura Orientada a Servicios (SOA)... En este caso, se ha optado por una arquitectura basada en microservicios.

#### Arquitectura basada en microservicios

La arquitectura basada en microservicios es un enfoque para el diseño de aplicaciones que consiste en un conjunto de pequeños servicios, los cuales se ejecutan en su propio proceso y se comunican con mecanismos ligeros (normalmente una API de recursos HTTP como es el caso de este proyecto) [6]. Cada microservicio está especializado en una tarea concreta y trabaja de forma independiente.

De este modo, tendremos las funcionalidades de la plataforma OTT distribuidas en diferentes microservicios, aislando las funcionalidades y permitiendo que cada uno de ellos pueda ser desarrollado, desplegado y escalado de forma independiente. Esto facilita la evolución de la plataforma, ya que se puede mejorar cada microservicio con la confianza de que si se hace correctamente no afectará al resto de la plataforma. Lo mismo ocurre con la tolerancia a

fallos, ya que si la arquitectura está bien diseñada, un fallo en un microservicio no debería afectar al resto de la plataforma, o debería hacerlo lo menos posible.

Ejemplos de microservicios más comunes en una plataforma OTT son:

- **Gestión de usuarios:** encargado de la gestión de los usuarios de la plataforma, incluyendo el registro, autenticación y autorización de los mismos.
- **Gestión de contenidos:** encargado de la gestión y almacenamiento de los contenidos de la plataforma, de los metadatos de los contenidos y de los ficheros multimedia.
- **Orquestador:** encargado de orquestar las peticiones de los usuarios a los diferentes microservicios de la plataforma.
- **Recomendaciones:** encargado de la generación de recomendaciones personalizadas para los usuarios de la plataforma.

### Componentes de la arquitectura

En una plataforma OTT, además de los microservicios, existen varios componentes clave que conforman la infraestructura técnica y que son esenciales para su funcionamiento. Estos componentes se encargan de soportar las operaciones críticas, desde la ingesta y distribución de contenido hasta la experiencia de usuario final. Algunos de los componentes más importantes son:

- **CDN (Content Delivery Network):** Red de distribución de contenidos que permite la entrega rápida y eficiente de contenidos multimedia a los usuarios finales. La CDN almacena copias de los contenidos en servidores distribuidos geográficamente, lo que reduce la latencia y mejora la velocidad de carga de los contenidos.
- **Ingesta y gestión de contenidos:** Componente encargado de recibir y procesar y gestionar todos los contenidos que van a estar disponibles en la plataforma, permitiendo a los administradores de contenido subir, editar, etiquetar, organizar y , en general, gestionar los contenidos de la plataforma y sus metadatos, preparando estos para su distribución.
- **Gestión de usuarios:** Encargado de toda gestión relacionada con los usuarios de la plataforma, incluyendo el registro, autenticación, autorización, gestión de perfiles y preferencias, etc.
- **Interfaz de usuario:** Es la parte visible de la plataforma, donde los usuarios interactúan con la misma. Es el componente encargado de recibir la información de los microservicios y presentarla de forma amigable al usuario.

- **monitorización y análisis:** Componente encargado de monitorizar el rendimiento de la plataforma y de analizar los datos generados por los usuarios para mejorar la experiencia de usuario y la eficiencia de la plataforma.
- **Otros componentes:** Existen otros componentes que pueden ser necesarios en la plataforma: componentes de seguridad, monetización y publicidad, gestión de pagos, etc.

### 2.2.3 Experiencia de usuario (UX)

A diferencia de los medios tradicionales, donde el contenido se emite en un horario fijo y de una forma específica, sin darle al usuario la posibilidad de interactuar con el contenido, los servicios OTT permiten a los usuarios navegar por el catálogo de contenidos, seleccionar lo que desean ver, su información, su génera (temporadas, episodios...) y cualquier otra información relevante, y ver el contenido como y cuando quieran, sin restricciones de tiempo o lugar. Esta flexibilidad y control que desean los usuarios sobre el contenido es tanto una de las mejores características de los servicios OTT como uno de los mayores desafíos para los diseñadores.

La experiencia de usuario (UX) [7] se define como los factores y elementos relativos a la interacción del usuario con la interfaz de un sistema, dispositivo o aplicación. Estos factores son claves para definir si un usuario disfruta o no de la experiencia de uso de un producto o servicio. Es fundamental prestar atención a estos factores en el diseño de cualquier aplicación, y más aún en el caso de una plataforma OTT, donde los usuarios buscan una experiencia fluida, personalizada, intuitiva y atractiva.

#### Principios de diseño de UX

El diseño de un UX efectivo se basa en una serie de principios y prácticas que deben adaptarse a cada proyecto, a sus necesidades y características. En el caso de este tipo de aplicaciones, dos de estos principios son la usabilidad y la simplicidad. La plataforma para el consumo de los contenidos debe ser fácil de usar, intuitiva y accesible para todo tipo de usuarios, independientemente de su nivel de experiencia o conocimientos técnicos. Esto nos asegurará que cualquier usuario podrá hacer uso de nuestras plataformas. Otros principios importantes son la consistencia y la adaptabilidad entre dispositivos y plataformas; la accesibilidad, para garantizar que todos los usuarios puedan disfrutar de la plataforma; y la personalización, para ofrecer una experiencia única y relevante a cada usuario.

#### Elementos clave de la UX en una plataforma OTT

Las plataformas OTT son aplicaciones con una serie de componentes clave que no pueden faltar ya que son necesarios para el funcionamiento de la misma. De la misma manera,

estos componentes deben comportarse de una manera determinada. Los componentes más importantes son:

- **Contenidos** Elemento central de la plataforma sobre el que gira toda la experiencia de usuario. Los contenidos deben ser fáciles de encontrar, navegar y consumir, y deben estar presentados de forma atractiva y organizada.
- **Agrupaciones de contenidos** Ya sea en listas, series, temporadas, categorías, géneros, etc. Las agrupaciones de contenidos permiten a los usuarios explorar y descubrir nuevos contenidos de forma sencilla y atractiva.
- **Metadata** información sobre cualquiera de los elementos que conforman la plataforma: descripciones, fechas, títulos, actores, etc. Aporta al usuario el contexto necesario para entender y valorar los contenidos.
- **Páginas** Cada menú, serie, temporada, episodio, etc. debe tener su propia página con la información y funcionalidades necesarias para hacer uso de la plataforma.
- **Botones y controles** Los botones y controles de reproducción, pausa, adelanto, retroceso, etc. deben ser intuitivos y accesibles para que los usuarios puedan interactuar con el contenido de forma sencilla.

#### 2.2.4 Escalabilidad, adaptabilidad y multiplataforma

La escalabilidad, adaptabilidad y multiplataforma son conceptos fundamentales en el diseño y desarrollo de aplicaciones, especialmente en caso de estudio de este proyecto, una plataforma OTT con soporte multicliente y multiplataforma.

##### Escalabilidad

La escalabilidad es la capacidad de un sistema para crecer. En este crecimiento se incluye: el aumento de la capacidad de la aplicación para manejar más usuarios, datos y transacciones, adaptarse a las nuevas necesidades de los mercados y tecnologías, mejorar tanto las capacidades como el rendimiento del sistema y ampliar el catálogo de servicios y funcionalidades.

La plataforma debe ser capaz de escalar tanto horizontal como verticalmente. La escalabilidad horizontal se refiere a la capacidad de añadir más servidores a la infraestructura para distribuir la carga de trabajo y mejorar el rendimiento. Por otro lado, la escalabilidad vertical implica la mejora de los recursos de un servidor existente, como la memoria, el almacenamiento o la potencia de procesamiento.

La plataforma también debe estar pensada para futuras actualizaciones y mejoras, de modo que pueda adaptarse a los cambios en las necesidades del negocio y las preferencias de los



usuarios sin afectar a la estabilidad y el rendimiento del sistema. La escalabilidad es esencial para garantizar que la plataforma pueda crecer de manera sostenible y seguir siendo competitiva en un entorno en constante evolución.

### **Adaptabilidad**

La adaptabilidad es la capacidad de un sistema para ajustarse a diferentes contextos, entornos y dispositivos. En el caso de una plataforma OTT, la adaptabilidad implica la capacidad de ofrecer una experiencia de usuario consistente y de alta calidad en una amplia variedad de dispositivos, sistemas operativos y navegadores.

La adaptabilidad es esencial para garantizar que los usuarios puedan acceder al contenido de la plataforma desde cualquier dispositivo y en cualquier momento, sin importar las limitaciones técnicas o las preferencias personales. La plataforma debe ser capaz de adaptarse automáticamente a las características de cada dispositivo, como el tamaño de la pantalla, la resolución, la velocidad de conexión y la capacidad de procesamiento, para ofrecer una experiencia de usuario óptima y satisfactoria.

La adaptabilidad también implica la capacidad de la plataforma para adaptarse a los cambios en el mercado, las tecnologías y las preferencias de los usuarios. La plataforma debe ser flexible y modular, de modo que pueda integrar nuevas funcionalidades, servicios y contenidos de manera rápida y sencilla, sin afectar a la estabilidad y el rendimiento del sistema.

### **Multiplataforma**

La multiplataforma es la capacidad de un sistema para funcionar en diferentes dispositivos, sistemas operativos y navegadores. En el caso de una plataforma OTT, la multiplataforma implica la capacidad de ofrecer una experiencia de usuario consistente y de alta calidad en una amplia variedad de dispositivos, como ordenadores, smartphones, tablets, smart TVs y consolas de videojuegos.

La multiplataforma es esencial para garantizar que los usuarios puedan acceder al contenido de la plataforma desde cualquier dispositivo y en cualquier momento, sin importar las limitaciones técnicas o las preferencias personales, aumentando así el número de usuarios a los que se puede llegar y mejorando la visibilidad y la rentabilidad de la plataforma.

La multiplataforma también implica la capacidad de la plataforma para integrarse con otros sistemas y servicios, como redes sociales, sistemas de pago, motores de recomendación y herramientas de análisis, para ofrecer una experiencia de usuario completa y personalizada, adaptada a las necesidades y preferencias de cada usuario.

## Conclusiones

En resumen, la escalabilidad, adaptabilidad y multiplataforma son conceptos fundamentales en el diseño y desarrollo de una plataforma OTT, ya que permiten garantizar que la plataforma pueda crecer de manera sostenible, adaptarse a los cambios en el mercado y las tecnologías, y ofrecer una experiencia de usuario consistente y de alta calidad en una amplia variedad de dispositivos y contextos.

## 2.3 Fundamentos tecnológicos

### 2.3.1 Arquitectura Software: Comunicación entre componentes y desarrollo multicliente

#### Comunicación entre componentes

Una de las claves de la arquitectura software 2.2.2 es la comunicación entre componentes y microservicios 2.2.2. En el caso de esta aplicación esta comunicación se realiza a través de una API REST, que es una interfaz de programación de aplicaciones (API) que se ajusta a los principios de diseño del estilo arquitectónico de transferencia de estado representacional (REST).

REST es un estilo de arquitectura que se basa en la comunicación a través de HTTP, y que se caracteriza por ser sencillo, escalable y que se puede utilizar en cualquier entorno. En una API REST, los recursos son identificados por URLs y se acceden a través de los métodos HTTP estándar (GET, POST, PUT, DELETE). Esto permite que los clientes de la API puedan realizar operaciones sobre los recursos de forma sencilla y estandarizada.

En el caso de esta aplicación, la API REST se encarga de la comunicación entre los diferentes microservicios de la plataforma OTT, ofreciendo a la interfaz una forma de obtener la información necesaria para mostrar en la aplicación, y a los microservicios una forma de recibir las peticiones de la interfaz y devolver la información necesaria.

#### Desarrollo multicliente

Una de las ventajas que nos proporciona el API REST que utilizamos en esta aplicación es la de tener el poder de decisión sobre la información que queremos obtener. Así, en función de los parámetros que se añadan a la URL, la API nos devolverá la información solicitada o realizará una acción sobre un cliente en concreto.

Esta diferenciación nos permite, a través de la misma API, trabajar sobre diferentes clientes sin necesidad de crear una API nueva o un microservicio a medida en cada caso.

### 2.3.2 Tecnologías Web: JavaScript, HTML y CSS

La plataforma OTT está construida sobre la base de tecnologías web fundamentales: JavaScript (JS), HTML, y CSS. Estas tecnologías permiten el desarrollo de una interfaz de usuario interactiva y adaptable, que puede ser fácilmente desplegada en diferentes plataformas, incluyendo web, smart TVs y dispositivos móviles.

**JavaScript (JS)** JavaScript es el motor que impulsa la interactividad en la plataforma. Permite manipular el DOM, gestionar eventos del usuario, realizar llamadas asíncronas a la API de la CDN para obtener contenido dinámico y, sobre todo, implementar la lógica de negocio de la plataforma. JavaScript es un lenguaje de programación versátil y flexible, que permite la creación de aplicaciones web complejas y altamente interactivas.

**HTML** HTML define la estructura de la interfaz de usuario, organizando el contenido de manera semántica y accesible. Su compatibilidad con todos los navegadores web y dispositivos asegura que la plataforma pueda ser utilizada en una amplia gama de entornos, desde navegadores de escritorio hasta smart TVs.

**CSS** CSS es responsable de la presentación visual de la plataforma, permitiendo una estilización coherente y atractiva en todas las plataformas soportadas. A través de técnicas de diseño responsivo y el uso de transiciones y animaciones, CSS asegura que la interfaz se adapte y ofrezca una experiencia de usuario óptima en cualquier dispositivo.

### 2.3.3 Desarrollo multiplataforma

El desarrollo de una plataforma OTT que funcione de manera eficiente en múltiples dispositivos requiere un enfoque que combine la flexibilidad de las tecnologías web con herramientas específicas para adaptar la aplicación a distintos sistemas operativos y dispositivos. La aplicación se ha desarrollado utilizando JavaScript (JS), HTML, y CSS, tecnologías que permiten una fácil adaptación a la web y una base sólida para su conversión a otras plataformas, como televisores inteligentes y Android TV.

#### Adaptación a Web

Gracias a la utilización de JavaScript, HTML y CSS, la adaptación de la aplicación a la web es sencilla y directa. La aplicación se ha desarrollado siguiendo los principios de diseño responsivo, lo que permite que la interfaz se adapte automáticamente al tamaño de la pantalla del dispositivo en el que se visualiza. Además, la aplicación se ha probado en los principales navegadores web (Chrome, Firefox, Safari, Edge) para garantizar su compatibilidad y funcionamiento en distintos entornos.

### **Adaptación a Smart TVs**

La adaptación de la aplicación a televisores inteligentes es un proceso más complejo, ya que estos dispositivos suelen tener sistemas operativos propietarios y limitaciones y peculiaridades específicas. Cada fabricante de televisores inteligentes tiene su propio sistema operativo y plataforma de desarrollo, así como su propia manera de empaquetar, ejecutar y distribuir aplicaciones. Por ello, la adaptación de la aplicación a televisores inteligentes requiere un estudio y enfoque específico para cada plataforma.

#### **Tizen**

Tizen [8] es un sistema operativo de código abierto desarrollado por Samsung Electronics y la Linux Foundation. Tizen utiliza tecnologías web como HTML, CSS y JavaScript para el desarrollo de aplicaciones, lo que facilita la adaptación de la aplicación a este sistema operativo. Las peculiaridades de Tizen, como su sistema de empaquetado y distribución de aplicaciones, se han tenido en cuenta durante el desarrollo de la aplicación para garantizar su compatibilidad y funcionamiento. De este modo, la aplicación ha sido probada utilizando la plataforma Tizen studio la cual nos permite emular el funcionamiento de la aplicación en un televisor Samsung con Tizen OS y, más importante y confiable, instalar y ejecutar la aplicación en un televisor Samsung. El formato de empaquetado de la aplicación para Tizen es un archivo .wgt, el cual contiene todos los recursos y archivos necesarios para la ejecución y distribución de la .

#### **WebOS**

WebOS [9] es un sistema operativo de código abierto utilizado por LG Electronics para sus televisores inteligentes. WebOS utiliza tecnologías web como HTML, CSS y JavaScript para el desarrollo de aplicaciones, permitiendo la adaptación de la aplicación a este sistema operativo. Lg proporciona un CLI (Command Line Interface) para la ejecución de las funcionalidades necesarias para la creación, empaquetado y distribución de aplicaciones para WebOS. Además, también está disponible una extensión para Visual Studio Code que facilita aún más el desarrollo. La aplicación ha sido probada en un televisor LG con WebOS a través de ambas herramientas, garantizando su compatibilidad y funcionamiento. El formato de empaquetado de la aplicación para WebOS es un archivo .ipk.

#### **Android TV**

Android TV [10] es una plataforma de televisión inteligente desarrollada por Google. Está basado en el sistema operativo Android. La adaptación de la aplicación a este sistema operativo varia en comparación con las anteriores. Android no permite el uso de las tecnologías de

desarrollo js, html y css de manera nativa por lo que se requiere de un proceso de conversión de la aplicación a un formato compatible con Android TV.

**Conversión a Android TV: Cordova** Para ello se ha utilizado la herramienta Cordova [11], que permite la utilización de las tecnologías estándar en lugar de las tecnologías nativas de determinadas plataformas como el caso de Android.

Apache Cordova es una plataforma de desarrollo móvil que permite a los desarrolladores utilizar tecnologías web estándar como HTML5, CSS3 y JavaScript para crear aplicaciones multiplataforma. Una de las principales ventajas de Cordova es que permite acceder a las funcionalidades nativas del dispositivo a través de una serie de APIs, lo que facilita la creación de aplicaciones que pueden ejecutarse en múltiples plataformas sin necesidad de escribir código específico para cada una de ellas. Esto no solo reduce el tiempo de desarrollo, sino que también simplifica el mantenimiento y la actualización de las aplicaciones.

Además, Cordova cuenta con una amplia comunidad de desarrolladores y una gran cantidad de plugins que extienden sus capacidades. Estos plugins permiten integrar funcionalidades adicionales como el acceso a la cámara, el almacenamiento local, las notificaciones push y mucho más. La flexibilidad y extensibilidad de Cordova lo convierten en una opción popular para el desarrollo de aplicaciones móviles híbridas, especialmente en proyectos donde el tiempo y los recursos son limitados. La capacidad de Cordova para adaptarse a diferentes plataformas, incluyendo Android TV, demuestra su versatilidad y potencia como herramienta de desarrollo.

La aplicación ha sido probada en un dispositivo Android TV, garantizando su compatibilidad y funcionamiento. El formato de empaquetado de la aplicación para Android TV es un archivo .apk.

### **Empaquetado y ejecución de la aplicación**

En el caso de las aplicaciones de TV lo que se ha hecho es crear una carpeta que sirve de plantilla para la creación de los archivos necesarios para la ejecución de la aplicación en los distintos sistemas operativos de las TV. La aplicación se ha ido desarrollando con una estructura que permite que en el momento de querer empaquetar una aplicación para probar o distribuir un código, sea sencillo añadir ese código a la plantilla y ejecutar con las herramientas necesarias para cada sistema operativo.

## Capítulo 3

# Metodología

---

### 3.1 Introducción

En esta sección se describe la metodología de trabajo seguida para el desarrollo de esta plataforma OTT. Se detallan las fases de desarrollo, las herramientas utilizadas, y las metodologías de trabajo empleadas para la implementación de la plataforma.

#### 3.1.1 Diseño del proyecto

Crear un proyecto que a partir de un mismo código fuente genere una aplicación OTT con la información y funcionalidades para cada cliente, adaptándose a la mayor cantidad de dispositivos posibles. Esta fue la premisa principal del proyecto y el objetivo que se me planteó cuando comencé en la empresa.

La primera etapa del proyecto fue el análisis y diseño de la plataforma. Se definieron los objetivos generales del proyecto:

- Crear una plataforma OTT que permita a los clientes de la empresa tener su propia aplicación de distribución de contenido.
- Permitir a los clientes personalizar su aplicación lo máximo posible con su propia marca y contenido.
- Crear una plataforma que sea escalable y que pueda adaptarse lo máximo posible a las necesidades de cada cliente.
- Crear un código que permita ejecutar la aplicación en la mayor cantidad de dispositivos posibles.

**Y lo más importante:** No tener que crear un código fuente para cada cliente y dispositivo.

Una vez clara la idea general del proyecto, se procedió a la definición de los requisitos y funcionalidades de la plataforma (páginas, menús, funcionalidades, etc.), así como el alcance del proyecto. Se realizó un análisis de mercado y se decidió comenzar dando soporte por el momento en los sistemas operativos Android, Tizen y WebOS, con la premisa de que el código debe estar preparado para ampliarse a otros sistemas operativos en el futuro.

Otro punto importante fue el estudio de los productos similares ya existentes creados por la empresa, para comenzar creando la base del producto con las mismas características visuales y funcionales, para luego ir evolucionando y mejorando el producto.

Se realizó una planificación inicial del proyecto, definiendo las tareas y funcionalidades prioritarias, con la idea de tener un producto mínimo viable lo antes posible para comenzar a presentarlo y probarlo.

## 3.2 Metodología de Desarrollo

El desarrollo de la plataforma OTT se llevó a cabo siguiendo una metodología ágil. Esta metodología permite una mayor flexibilidad en el desarrollo del proyecto, y facilita la adaptación a los cambios que puedan surgir durante el desarrollo del proyecto. Se asegura así la evolución y mejora controlada y continua del producto final.

### 3.2.1 Implementación progresiva e incremental

El proceso de desarrollo se estructuró en torno a la implementación de funcionalidades de manera progresiva e incremental. En lugar de intentar desarrollar todas las características al mismo tiempo, el proyecto se dividió en ciclos más pequeños, cada uno enfocado en la entrega de una funcionalidad específica.

Estos ciclos se componen de las siguientes fases:

- **Análisis:** En esta fase se escoge la funcionalidad a implementar y se identifican los requisitos de la misma. La selección de la funcionalidad a implementar se realiza en base a la prioridad de la misma y a la dependencia de otras funcionalidades.
- **Diseño:** En esta fase se diseña la arquitectura de la funcionalidad a implementar. Se identifican los componentes necesarios y se definen las interacciones entre ellos.
- **Implementación:** En esta fase se implementa la funcionalidad siguiendo el diseño previamente definido.
- **Pruebas:** En esta fase se realizan pruebas unitarias y de integración para asegurar que la funcionalidad implementada cumple con los requisitos definidos en la fase de análisis.

### 3.2.2 Adaptación continua y flexibilidad

Una de las ventajas de la metodología ágil es la adaptación continua y la flexibilidad que ofrece, permitiendo hacer frente a los cambios y nuevas necesidades que puedan surgir durante el desarrollo del proyecto y ajustar las prioridades y el enfoque del proyecto en caso de ser necesario.

Durante el desarrollo de la plataforma OTT, se realizaron reuniones periódicas los clientes para revisar el progreso del proyecto y validar las funcionalidades implementadas. Estos clientes pudieron proponer cambios y mejoras en la plataforma, que fueron evaluados y estudiados. Debido a la naturaleza multicliente del proyecto, en todo momento se buscó que los cambios o mejoras realizadas en base a peticiones de un cliente en específico no afectaran a los demás clientes. Uno de los objetivos principales que tiene el proyecto es seguir aumentando la flexibilidad y personalización de la plataforma para cada cliente.

### 3.2.3 Reuniones de Validación y Planificación

Después de cada ciclo de desarrollo, se llevaron a cabo reuniones de validación para revisar las funcionalidades completadas y planificar las siguientes etapas. Estas reuniones fueron fundamentales para asegurar que el proyecto se mantuviera alineado con los objetivos del cliente y para permitir la rápida adaptación a nuevos requisitos. En estas sesiones, se discutía el estado actual del proyecto, se evaluaban posibles mejoras, y se decidía la próxima funcionalidad a implementar. Este enfoque colaborativo garantizó una toma de decisiones informada y una respuesta rápida a los cambios necesarios.

### 3.2.4 Ventajas de la metodología ágil

La metodología ágil ofrece una serie de ventajas que permiten un desarrollo más eficiente y controlado del proyecto. Algunas de las ventajas más destacadas son las siguientes:

- **Mayor flexibilidad:** La metodología ágil permite adaptarse a los cambios y nuevas necesidades que puedan surgir durante el desarrollo del proyecto.
- **Mayor control:** Al dividir el proyecto en ciclos más pequeños, se facilita el seguimiento y control del progreso del proyecto.
- **Mayor calidad:** Al realizar pruebas continuas a lo largo del desarrollo del proyecto, se asegura que el producto final cumple con los requisitos y está libre de errores.
- **Mayor satisfacción del cliente:** Al involucrar al cliente en el proceso de desarrollo y permitirle proponer cambios y mejoras, se asegura que el producto final cumple con sus expectativas.



### 3.3 Descripción de las Etapas de Desarrollo

En esta sección, se describen las principales etapas del desarrollo del proyecto, desde la planificación inicial hasta el despliegue final. Cada etapa jugó un papel crucial en la evolución del proyecto, permitiendo una implementación organizada y eficiente de la plataforma OTT.

#### 3.3.1 Fase de Análisis y Planificación

##### Recolección de Requisitos

En esta etapa, se realizó un análisis exhaustivo para identificar los requisitos funcionales y no funcionales del proyecto. Esto incluyó reuniones con los *stakeholders* para entender las expectativas y necesidades del cliente, así como un análisis de mercado para asegurar que la plataforma cumpliera con los estándares actuales.

##### Estudio de Viabilidad

Se evaluó la viabilidad técnica del proyecto, considerando las tecnologías disponibles, las capacidades del equipo, y las restricciones de tiempo y presupuesto. Este estudio permitió definir el alcance realista del proyecto y establecer prioridades.

##### Planificación Inicial

Se elaboró un cronograma detallado con hitos clave, que incluyó la planificación de *sprints* y la asignación de recursos para las diferentes fases del desarrollo.

#### 3.3.2 Fase de Diseño

##### Diseño de Arquitectura

Durante esta etapa, se definió la arquitectura general de la plataforma. Se decidió utilizar una arquitectura basada en microservicios para garantizar la escalabilidad y flexibilidad del sistema. También se diseñó la estructura de comunicación entre los componentes y se eligieron las tecnologías más adecuadas para cada parte del sistema.

##### Diseño de la Interfaz de Usuario

Se crearon prototipos y *wireframes* para la interfaz de usuario utilizando herramientas como Figma o Adobe XD. Estos diseños fueron iterados y refinados a través de revisiones con los *stakeholders*, asegurando que la interfaz cumpliera con las expectativas de usabilidad y estética.

### 3.3.3 Fase de Desarrollo

#### Implementación de Funcionalidades

Esta fase fue el núcleo del proyecto, donde se desarrollaron las funcionalidades clave de la plataforma OTT. Cada funcionalidad se implementó de manera incremental, siguiendo un ciclo ágil de análisis, diseño, codificación y pruebas. Se utilizaron tecnologías web como JavaScript, HTML y CSS para asegurar la compatibilidad multiplataforma.

#### Optimización y Refactorización

A medida que se añadían nuevas funcionalidades, el código se refactorizaba y optimizaba para mejorar el rendimiento y la mantenibilidad. Este proceso incluyó la simplificación de la lógica de negocio, la mejora de la estructura del código, y la optimización de la carga y la respuesta de la aplicación.

### 3.3.4 Fase de Pruebas

#### Pruebas Unitarias y de Integración

Se realizaron pruebas unitarias para cada componente, asegurando que las funciones individuales funcionaran correctamente. Luego, se realizaron pruebas de integración para verificar que los componentes interactuaran de manera efectiva y sin conflictos.

#### Pruebas de Compatibilidad

Se llevaron a cabo pruebas exhaustivas en diferentes dispositivos y sistemas operativos (web, Tizen, WebOS, Android TV) para asegurar que la aplicación funcionara correctamente en todos los entornos previstos. Esto incluyó pruebas de rendimiento para garantizar que la aplicación se ejecutara sin problemas en dispositivos con capacidades de hardware limitadas.

### 3.3.5 Fase de Despliegue y Mantenimiento

#### Despliegue en Producción

Una vez que todas las funcionalidades principales fueron implementadas y probadas, se procedió al despliegue de la plataforma en los entornos de producción. Se utilizaron *pipelines* de CI/CD para automatizar el proceso de despliegue, minimizando los riesgos de errores y asegurando una transición suave a producción.

**Plan de Mantenimiento**

Se desarrolló un plan de mantenimiento para garantizar la continuidad y estabilidad de la plataforma tras su lanzamiento. Esto incluyó la planificación de actualizaciones regulares, la gestión de incidentes, y la respuesta a problemas reportados por los usuarios. También se incluyó un plan para la incorporación de nuevas funcionalidades según las necesidades futuras de los clientes.

## Capítulo 4

# Análisis

---

### 4.1 Introducción

El análisis es la primera fase del ciclo de vida de un proyecto de software. En esta fase se recopilan los requisitos del sistema, se identifican los objetivos y se establecen las restricciones que debe cumplir el sistema. El análisis es una fase crucial en el desarrollo de software, ya que de ella depende el éxito o fracaso del proyecto. En esta fase se establece la base sobre la que se construirá el sistema, por lo que es importante que se realice de forma correcta y exhaustiva.

En esta sección se describirá como se ha realizado el análisis del proyecto, los objetivos que se pretenden alcanzar y las restricciones que se han establecido.

### 4.2 Requisitos

En software, se entiende por requisito a una condición o capacidad que debe cumplir un sistema para resolver un problema o satisfacer una necesidad. Son las funciones, características y restricciones que debe cumplir un producto final.

La identificación de requisitos es una etapa crucial en cualquier proyecto software ya que marca el camino a seguir en el desarrollo del sistema. Si no se identifican correctamente los requisitos, el sistema no trabajará correctamente y no cumplirá ni sus especificaciones ni las expectativas del cliente. Por ello, es importante que los requisitos sean claros, precisos y completos.

Para hacer una correcta identificación de los requisitos, hay que tener en cuenta y conocer los dos tipos de requisitos que nos podemos encontrar: los requisitos funcionales, que describen las funciones que debe realizar el sistema, y los requisitos no funcionales, que describen las características que debe tener el sistema.

Además, es importante conocer de donde se deben obtener los requisitos. Si leemos el apartado 7.2 de la norma ISO 9001:2000, Realización del producto. Procesos relacionados con

el cliente, este nos dice que los requisitos relativos se obtienen de:

- Los requisitos especificados por el cliente.
- Requisitos no especificados por el cliente pero necesarios para el uso especificado o para el uso previsto, cuando sea conocido.
- Requisitos legales y reglamentarios relacionados con el producto.
- Requisitos adicionales determinados por la organización.

Todo los requisitos que se detecten en esta fase deben ser revisados por la organización para asegurarse de que son correctos, completos, se entienden y pueden ser suministrados. Este proceso de revisión debe llevarse a cabo a lo largo de todo el proyecto cuando alguno de estos requisitos sufra alguna modificación.

#### 4.2.1 Requisitos de la plataforma OTT

En esta sección se describirán los requisitos recogidos para la plataforma OTT. Los requisitos detallados a continuación son los relacionados con la creación de la aplicación multi-plataforma y multiciente, que recibe la información y determina como mostrar esta. Dentro de estos requisitos no se incluyen los relacionados con otros componentes de la plataforma, como la CDN, la gestión de usuarios, la gestión de contenidos, etc.

##### Requisitos funcionales

**Requisitos especificados por el cliente** En este caso los requisitos no son unicamente de un cliente, sino que como ya se ha comentado en la introducción, se han recogido los requisitos de varios clientes. Ejemplos de requisitos especificos de clientes:

##### Generales

- El usuario debe tener una pantalla donde poder buscar contenido.
- El usuario debe disponer de una pantalla con los ajustes de la aplicación.

##### Radio Television de Castilla y León. CyLTV Play

- El usuario debe tener disponible una pantalla donde se muestre todo el contenido en directo.
- El usuario debe tener disponible una pantalla donde se muestre todo el contenido a la carta.
- El usuario debe tener una pantalla disponible para cada delegación del canal La 8.

**Oh!Jazz**

- El usuario debe poder iniciar y cerrar sesión.
- Los usuarios no loggados no pueden acceder a todos los contenidos.
- El usuario debe poder cambiar de idioma la plataforma.

**Requisitos no especificados por el cliente** Este tipo de plataformas tienen unos requisitos comunes que no han sido especificados por los clientes, pero que son necesarios para el correcto funcionamiento de la plataforma. Estos requisitos son:

- La interfaz debe recibir la información necesaria para saber qué y cómo mostrar los contenidos.
- La aplicación debe ser capaz de determinar, para que cliente está trabajando y tomar decisiones en función de esto.
- La aplicación debe ser capaz de determinar, para que plataforma está trabajando y tomar decisiones en función de esto.
- La aplicación debe crear una interfaz de usuario que sea atractiva, intuitiva y fácil de usar.
- La aplicación debe crear una interfaz que se adapte a distintas resoluciones.
- La aplicación debe ser capaz de reproducir los contenidos multimedia en cualquier sistema operativo.
- La aplicación debe estar personalizada con los logos y colores de cada cliente.
- La aplicación debe funcionar con los botones del mando a distancia en el caso de las televisiones.
- La aplicación debe mostrar los datos correctamente sin alterar la información.

**Requisitos no funcionales****Requisitos de rendimiento**

- La aplicación debe ser capaz de mostrar los contenidos en el menor tiempo posible.
- La aplicación debe ser capaz de reproducir los contenidos multimedia sin cortes.
- La experiencia de usuario debe ser lo más fluida posible.
- La aplicación debe ser capaz de adaptarse a distintas resoluciones.

**Requisitos de seguridad**

- La aplicación debe ser segura y no permitir accesos no autorizados.
- La aplicación debe proteger los datos de los usuarios.
- La aplicación debe proteger los datos de los clientes.

**Requisitos de usabilidad**

- La aplicación debe ser intuitiva y fácil de usar.
- La aplicación debe ser atractiva y agradable visualmente.
- La aplicación debe ser accesible para todo tipo de usuarios.

**Requisitos de mantenimiento**

- La aplicación debe ser fácil de mantener y actualizar.
- La aplicación debe ser fácil de escalar.
- La aplicación debe ser fácil de desplegar.

**Requisitos de escalabilidad**

- La aplicación debe ser capaz de adaptarse a un aumento de usuarios.
- La aplicación debe ser capaz de adaptarse a un aumento de contenidos.
- La aplicación debe ser capaz de adaptarse a un aumento de clientes.

**Requisitos de disponibilidad**

- La aplicación debe estar disponible en todo momento.
- La aplicación debe ser capaz de recuperarse de fallos.
- La aplicación debe ser capaz de recuperarse de caídas.

**Requisitos de compatibilidad**

- La aplicación debe ser compatible con todos los sistemas operativos.
- La aplicación debe ser compatible con todos los navegadores.
- La aplicación debe ser compatible con todos los dispositivos.

**Requisitos de internacionalización**

- La aplicación debe ser capaz de adaptarse a distintos idiomas.
- La aplicación debe ser capaz de adaptarse a distintas culturas.
- La aplicación debe ser capaz de adaptarse a distintas zonas horarias.

**Requisitos de personalización**

- La aplicación debe ser personalizable con los logos y colores de cada cliente.
- La aplicación debe ser personalizable con los logos y colores de cada plataforma.
- La aplicación debe ser personalizable con los logos y colores de cada delegación.

**Requisitos de adaptabilidad**

- La aplicación debe ser capaz de adaptarse a distintas resoluciones.
- La aplicación debe ser capaz de adaptarse a distintos dispositivos.
- La aplicación debe ser capaz de adaptarse a distintos navegadores.

**4.3 Necesidades del cliente**

Como se ha comentado en la sección 4.2.1, el análisis de requisitos en esta aplicación es un tanto especial, ya que no existe un solo cliente sino que existen varios clientes cada uno con sus necesidades y requisitos. El objetivo del proyecto es satisfacer y cumplir con esos requisitos y necesidades a partir de una única aplicación.

Esto supone que además del estudio de viabilidad técnica, para cada requisito se ha estudiado la manera de implementarlo de forma que, o bien sirva para todos los clientes, o en caso de ser un requisito muy específico y necesario para un cliente, hacerlo de manera que no afecte al resto de clientes.

Un ejemplo de esto es la interfaz. La interfaz ya cuenta con las funcionalidades necesarias para mostrar los colores, logos e imágenes de cada cliente. En cuanto a la presentación de los contenidos general, esta es muy similar en cualquier OTT del mercado: listas horizontales con imágenes y textos de los distintos contenidos. Esto no supone ningún problema ya que son los clientes los que gestionan los textos y las imágenes y los responsables de que estos estén en armonía para una interfaz atractiva. El reto está en elementos del menú dedicados a mostrar de manera especial los contenidos. Un ejemplo de esto, las listas de destacados. Estas listas son aquellas que se sitúan, por lo general, en la parte superior de las pantallas y



muestran los contenidos más importantes. El problema está en que cada cliente tiene una idea de como quiere que se muestren estos contenidos. Las opciones de configuración si se quiere dar libertad al cliente son muchas. El cliente puede decidir si quiere botón de play, si quiere que se muestre el título, el subtítulo, la duración, la fecha de publicación, etc. Para poder dar soporte a todas estas opciones lo que se ha hecho es crear un sistema de "etiquetado" o "clasificación" tanto de los contenidos como de las listas en los que se encuentran estos contenidos. De esta manera, en función del tipo de lista (también conocido como widget) y del tipo de contenido, estos se mostrarán de una manera u otra. Esto tiene una serie de ventajas y de inconvenientes. Si hablamos de ventajas, la principal es que cada cliente puede personalizar la interfaz a su gusto, diferenciandose del resto de clientes. Si hablamos de inconvenientes, el principal es que para cada tipo de lista y de contenido se debe dar soporte en la aplicación. Esto supone un esfuerzo extra en el desarrollo y en la gestión de los contenidos.

Este ha sido el enfoque que se ha dado a la aplicación desde un principio debido a la arquitectura de los distintos componentes creados y del funcionamiento de otras plataformas que pertenecen a la empresa. El enfoque en el que se está trabajando ahora mismo es en una mayor personalización de widgets y contenidos, sin necesidad de tener que crear un nuevo tipo de estos elementos cada vez que se requiera una personalización o presentación nueva. Lo mismo para las distintas pantallas.

## 4.4 Estudio de viabilidad

Cuando se comenzó a trabajar con la idea de crear una aplicación que pudiera generar aplicaciones para distintos sistemas operativos y dispositivos a partir de un mismo código, una de las primeras preguntas que surgieron fue si era posible hacerlo. Para obtener una respuesta a esta pregunta, era necesario saber si era viable una aplicación híbrida entre dispositivos móviles y televisores, si existía una tecnología que permitiera hacerlo en una cantidad de sistemas operativos y dispositivos interesante y si era posible hacerlo de manera eficiente y económica. En este apartado se va a dar respuesta a estas preguntas, analizando la viabilidad técnica y económica del proyecto.

### 4.4.1 Viabilidad técnica

La **viabilidad técnica** de un proyecto se refiere a la condición que hace posible realizar un proyecto o llevar a cabo una idea desde el punto de vista de la tecnología. Para estudiar la viabilidad de este proyecto es necesario saber si existe una herramienta o tecnología que permita trabajar en una cantidad de sistemas operativos y dispositivos lo suficientemente grande como para que compense realizar el proyecto. A continuación se detallará este estudio.

En cuanto a infraestructura, la empresa ya dispone de los componentes necesarios, total-

mente funcionales, para dar soporte a la aplicación. Tanto el CMS, como la base de datos y servidores de aplicaciones están en pleno funcionamiento y son capaces de soportar la carga de trabajo que supondría la aplicación, por lo que la viabilidad técnica en este aspecto es positiva.

### **Adaptación a web**

La aplicación web es una aplicación que se ejecuta en un navegador web. Este tipo de aplicaciones permiten una gran flexibilidad y portabilidad, no solo a la hora de ser explotadas por los usuarios, sino también a la hora de ser desarrolladas. Estas aplicaciones permiten utilizar una enorme variedad de tecnologías y lenguajes de programación, por lo que este requisito no es determinante a la hora de estudiar la viabilidad técnica del proyecto ya que podremos utilizar casi cualquier tecnología que nos permita adaptarnos a los otros dispositivos.

### **Adaptación a televisores**

El caso de los televisores es muy distinto. Aunque los televisores inteligentes actuales cuentan con un navegador web, estos no son ni tan potentes ni tan cómodos de utilizar como los de los dispositivos móviles o los ordenadores, por lo que aunque una persona pueda acceder a la aplicación desde su televisor, no es la opción que queremos para los televisores. La opción que aplica para este proyecto es la de crear una aplicación nativa para televisores. Estas aplicaciones son aplicaciones que se ejecutan en el sistema operativo del televisor, por lo que es necesario que el sistema operativo del televisor sea compatible con la tecnología que utilicemos para desarrollar la aplicación.

Los sistemas operativos que se estudiaron en un primer momento para este proyecto son los más utilizados en televisores inteligentes: Android Tv, Tizen, webOs y Apple Tv. El estudio comenzó con buscar que tecnologías permite cada opción de modo nativo. La opción que se buscaba era la de utilizar JavaScript, HTML y CSS para el desarrollo. Analizando Tizen y webOs encontramos que ambas opciones permiten el desarrollo a través de estas tecnologías, por lo que la viabilidad técnica comenzaba a ser posible. El problema llegó con Android Tv y Apple Tv. Estos sistemas operativos no permiten el desarrollo de aplicaciones a través de estas tecnologías, si no que tienen sus propias tecnologías. Buscando soluciones encontramos una opción para Android Tv, que es [Apache Cordova](#). Apache Cordova es una herramienta que permite desarrollar aplicaciones híbridas para distintos sistemas operativos a través de HTML, CSS y JavaScript. En el caso de Apple Tv no hubo suerte y no se encontró una solución viable con la que conseguir desarrollar con la tecnología escogida y "traducir" o "adaptar" el código a la tecnología de Apple Tv como se hizo con Android Tv.

Sin embargo, aunque no se encontró una solución para Apple Tv, el hecho de que se encontrara una solución para Android Tv y que las otras dos opciones permitieran el desarrollo

con las tecnologías escogidas supone trabajar sobre un 27% del mercado mundial de televisores inteligentes [12]. Aunque no es un porcentaje muy alto, es un porcentaje lo suficientemente grande como para que el proyecto sea viable.

Ahora que el proyecto ya está en marcha y ya existen versiones completamente funcionales, se están estudiando las implementaciones de nuevos sistemas operativos y dispositivos. Entre estas opciones está Vidaa OS (Hisense), que tiene el 7.8% del mercado mundial ahora mismo, posicionándose como el segundo sistema operativo más utilizado. El primer estudio que se hizo sobre este sistema operativo fue positivo, ya que indicó que se podía desarrollar con las tecnologías escogidas, por lo que a corto plazo se continuará con el estudio y se comenzará con la integración si el estudio de viabilidad técnica es positivo.

Por otro lado, Apple Tv se desarrollará por separado, y se buscará la manera de adaptar el código de la aplicación a la tecnología de Apple Tv e Ios en un futuro, si fuera posible.

### **Adaptación a dispositivos móviles**

En una primera etapa del proyecto, los objetivos no están centrados ahora mismo en que este proyecto genere aplicaciones para dispositivos móviles, pero si la adaptación de la página web para que sea *responsive* y permita una buena experiencia de usuario en dispositivos móviles. Además, si que existe esta implementación como objetivo a medio plazo por lo que el código permite una rápida integración de esta funcionalidad. Sin embargo, todo indica que al igual que para los televisores, el sistema operativo Ios no permitirá el desarrollo con las tecnologías escogidas, por lo que por el momento se centrará en Android y en la adaptación de la página web para dispositivos móviles.

### **Conclusión**

La conclusión tras el estudio de viabilidad técnica en las diferentes plataformas es que el proyecto es viable. Aunque no se ha encontrado una solución para Apple Tv, el hecho de que se haya encontrado una solución para Android Tv y que las otras dos opciones permitan el desarrollo con las tecnologías escogidas, además de la gran flexibilidad que ofrece el desarrollo web, hacen que el proyecto sea viable técnicamente.

#### **4.4.2 Viabilidad económica**

La *viabilidad económica* de un proyecto es la capacidad que tiene un proyecto de generar beneficios económicos. Para estudiar la viabilidad económica de este proyecto es necesario estudiar los costes que conlleva el desarrollo del proyecto y los beneficios que se pueden obtener. A continuación se detallará el estudio realizado, sin mostrar cifras concretas ya que es información confidencial.

**Costes**

Los costes que conlleva el desarrollo de este proyecto son los siguientes:

- Costes de desarrollo: el proyecto requiere de un equipo que se encargue de desarrollar la aplicación. En estos costes se incluyen los costes de desarrollar la aplicación, los diseñadores, herramientas... Para este proyecto el único coste que se ha tenido ha sido el de los desarrolladores, en este caso 1 persona únicamente. El resto de costes o bien no han sido necesarios (las herramientas son gratuitas, el sistema de pruebas también...) o bien son gastos del cliente (diseñadores, licencias del marketplace...). o bien son gastos que ya existían (servidores, base de datos...).
- Costes de mantenimiento: el proyecto requiere de un mantenimiento continuo para asegurar su correcto funcionamiento. En este caso, la aplicación está en continuo desarrollo y mejora ya que a medida que entren nuevos clientes y se adapte a nuevos sistemas operativos y dispositivos, se irán añadiendo nuevas funcionalidades y mejoras para todos los clientes. Dentro de este desarrollo entra también el mantenimiento de la aplicación, que se encarga de solucionar los problemas que puedan surgir. Por lo tanto, los costes derivados del mantenimiento son los mismos que los costes de desarrollo.

**Beneficios**

Los ingresos previstos para este proyecto vienen derivados de la contratación de los servicios de la empresa. La empresa cobra una cantidad fija por la creación de la aplicación y una cantidad fija por el mantenimiento de la misma.

Dentro de este cobro están incluidos los costes de desarrollo y mantenimiento y se indica que toda aquella funcionalidad extra que no está contemplada ni en el desarrollo ni en el contrato, se cobrará aparte.

**Conclusión**

Uno de los beneficios de este proyecto es la reducción de costes, ya que no es necesario realizar un desarrollo para todos los clientes, evitando los gastos de desarrollo y mantenimiento de una aplicación en cada proyecto de cada cliente. Esta reducción de costes es debido a la reutilización del , reducción del tiempo de desarrollo, reducción de personal necesario...

Por lo que si el estudio y la previsión de gastos y beneficios es correcta, y esto se refleja correctamente en el contrato, el proyecto es viable económicamente.

### 4.4.3 Conclusión final

Tras el estudio de viabilidad técnica y económica, se puede concluir que el proyecto es viable. La viabilidad técnica se ha demostrado a través de la creación de la aplicación y la adaptación a los distintos sistemas operativos y dispositivos y la viabilidad económica se ha demostrado a través de los costes y beneficios previstos. Por lo tanto, se puede afirmar que el proyecto es viable y que se puede continuar con su desarrollo.

## 4.5 Estudio de las características de los distintos sistemas operativos, dispositivos y tecnologías

Cuando se realizó el primer estudio del proyecto y los distintos estudios de viabilidad se analizaron las características de los distintos sistemas operativos, dispositivos y tecnologías que se podían utilizar para el desarrollo de la aplicación. En este apartado se va a realizar un estudio de las características de estos elementos y como estos encajan entre sí para el desarrollo de la aplicación.

### 4.5.1 Sistemas operativos

Cuando se comenzó a hablar del proyecto, el objetivo principal era el de comenzar con el desarrollo enfocado en televisores, pero sin perder la perspectiva del resto de dispositivos. Además, también se tenía una idea más o menos clara de los sistemas operativos con los que comenzar: Android TV, Tizen, webOs y Apple TV. Como ya se ha mencionado en la sección 4.4.1, Apple Tv tuvo que ser descartado, pero los otros tres sistemas operativos siguen siendo válidos. Se estudiaron las características de estos sistemas operativos a través de la documentación oficial y de la (escasa) información que se podía encontrar en la red. Las características tecnológicas básicas ya han sido comentadas en la sección de fundamentos tecnológicos 2.3.3. En este apartado vamos a extender un poco más estas características, a través del análisis realizado para el proyecto.

**características de los sistemas operativos** Como ya se ha comentado las tecnologías que se querían utilizar para el desarrollo de la aplicación eran JavaScript, HTML y CSS. El primer análisis que se realizó fue el de la compatibilidad de estas tecnologías con los distintos sistemas operativos. Aquí se descubrió que tanto Tizen como webOs son compatibles con estas tecnologías, por lo menos con las funcionalidades más básicas y necesarias, ya que existen funcionalidades más recientes o complejas que, aunque poco a poco se van integrando, todavía no están disponibles en estos sistemas operativos. Por otro lado, Android TV propuso un reto, ya que, por defecto, no es compatible con estas tecnologías. Sin embargo, estudian-

do un poco más a fondo, se descubrió que Android TV permite el desarrollo de aplicaciones utilizando tecnologías web a través de una herramienta llamada Apache Cordova [2.3.3](#).

**Empaquetado y distribución de aplicaciones** Otro punto importante que fue objeto de estudio fue como se debían configurar y empaquetar las aplicaciones para poder probar y distribuir en los distintos sistemas operativos. De nuevo, Tizen y webOs son muy similares en este aspecto, ya que aunque tienen formatos de empaquetado diferentes (wgt e ipk respectivamente), la estructura de los archivos son muy parecidos: ambas plataformas requieren un archivo HTML principal, un archivo de configuración (config.xml y package.json respectivamente) principal de la aplicación como nombre, version, etc. , el código de la aplicación (JavaScript, HTML y CSS) y los recursos necesarios (imágenes, fuentes, etc.). Por otro lado, la estructura de Android Tv es distinta ya que la estructura principal de la aplicación es la de Apache Cordova y todo el código de este necesario para convertir la aplicación a un formato compatible con Android TV. Dentro de esta estructura encontramos:

- **www**: Directorio que contiene el código de la aplicación (JavaScript, HTML y CSS).
- **config.xml**: Archivo de configuración de la aplicación.
- **platforms**: Directorio que contiene las plataformas en las que se ha empaquetado la aplicación.
- **plugins**: Directorio que contiene los plugins de Cordova que se han utilizado en la aplicación.

**Herramientas de desarrollo** Por último, se estudiaron las herramientas de desarrollo que se podían utilizar para el desarrollo de la aplicación. El desarrollo llevado a cabo ha sido a través de un código común desarrollado en Visual Studio Code. Sin embargo, la ejecución de las pruebas y la generación de los archivos de empaquetado no se podía realizar de la misma manera. Para cada SO se han creado unas plantillas con la estructura necesaria en cada caso y en las que se adjuntaba el código común y se añadía el código específico para cada SO (un archivo de configuración básico necesario para unas pocas funcionalidades específicas de cada SO).

Cada SO ofrece distintas opciones para la ejecución, pruebas y empaquetado de las aplicaciones.

- **Tizen**: Tizen Studio es la herramienta oficial de Samsung para el desarrollo de aplicaciones para Tizen. Esta herramienta ofrece muchas funcionalidades para el desarrollo como la gestión de dispositivos, emuladores, etc. Además, permite la generación de los

archivos de empaquetado y la instalación de la aplicación en un televisor Samsung con Tizen OS.

- **WebOs:** LG ofrece una CLI (Command Line Interface) para la ejecución de las funcionalidades necesarias para la creación, empaquetado y distribución de aplicaciones para WebOS. Además, también está disponible una extensión para Visual Studio Code que facilita aún más el desarrollo.
- **Android TV:** Las aplicaciones para Android Tv se pueden empaquetar y distribuir a través de Android Studio, que ofrece todas las funcionalidades necesarias para el desarrollo de las aplicaciones, desde escribir código hasta depurar e inspeccionar la aplicación. Sin embargo, como en este proyecto no se está utilizando Android de forma nativa, si no que se está adaptando a través de Cordova, Android Studio deja de tener todas las funcionalidades disponibles, debido a que ni tiene la estructura ni el lenguaje de programación que se utiliza por defecto en Android. La funcionalidad más afectada por esto es la de ejecución ya que no permite crear el apk correctamente porque no reconoce el código. Por este motivo, el empaquetado e instalación de la aplicación en un televisor Android se ha realizado a través de un terminal utilizando los comandos ofrecidos por Cordova.

#### 4.5.2 Dispositivos

Para el estudio de los dispositivos, los puntos en los que se centró el análisis fueron la capacidad de los dispositivos para ejecutar la aplicación, la resolución de pantalla y el tamaño de la pantalla.

El segundo punto arrojó una respuesta muy clara: la aplicación debía ser adaptativa. La aplicación debía ser capaz de adaptarse a cualquier resolución de pantalla, ya que se iba a ejecutar en televisores de distintas marcas y modelos, en móviles, en Pc, etc.

La filosofía de diseño que se tomó desde un principio fue la filosofía de diseño responsivo. Esta filosofía permite adaptarse automáticamente al tamaño de la pantalla del dispositivo en el que se visualiza. Para ello, se utilizan unidades relativas como porcentajes o vh y vw en lugar de unidades absolutas como px. De esta manera, la aplicación se adapta a cualquier resolución de pantalla, ya sea un televisor, un móvil o una tablet. Por ejemplo, para este proyecto las pruebas se han realizado en dos televisores (Samsung y Lg) que tienen una resolución de 1920x1080 ambas, y con una TCL(Android TV) que tiene una resolución de 920\*540. La aplicación se ha adaptado correctamente a los tres dispositivos, mostrando la misma información y funcionalidades en todos ellos.

Por otro lado, el primer punto fue un estudio clave, ya que la capacidad actual varía enormemente entre los distintos dispositivos sobre los que queremos trabajar, incluso entre dispo-

sitivos de la misma familia como los televisores, existen diferencias de eficiencia y capacidad entre los distintos modelos y SO. Unas primeras pruebas de eficiencia demostraron que las funcionalidades que en un portátil funcionaban sin ningún tipo de problema ni sobrecarga para el dispositivo, a la hora de probar en las televisiones los movimientos se veían pesados y lentos. Como no es posible estudiar todos los modelos de los diferentes dispositivos de las distintas marcas y la aplicación busca llegar al mayor número de dispositivos posible, el estudio se realizó sobre las herramientas y funcionalidades del desarrollo más eficientes.

Gracias a este estudio se consiguió mejorar la eficiencia y fluidez de la aplicación en los dispositivos de menor capacidad buscando cual era la manera más eficiente de realizar las distintas funcionalidades. Por ejemplo, se demostró que utilizar la propiedad `transform: translate()` en lugar de la propiedad `top` o `left` para mover los elementos de la aplicación era mucho más eficiente y fluido.

Se encontró otra manera de aumentar la eficiencia, por ejemplo, con una propiedad de CSS que permite acelerar el renderizado de la aplicación. Esta propiedad es la de `will-change`. Esta propiedad permite al navegador saber que un elemento va a cambiar y que debe prepararse para ello. De esta manera, el navegador puede optimizar el renderizado de la aplicación y mejorar la experiencia del usuario.

### 4.5.3 Tecnologías

En cuanto al estudio de las tecnologías, este es constante a lo largo de todo el desarrollo del proyecto. Si bien también lo es el estudio de los sistemas operativos y dispositivos en casos concretos, el estudio de las tecnologías es algo que se realiza de manera constante, ya que no solo las tecnologías cambian y evolucionan, sino que es muy importante para un proyecto buscar, estudiar y analizar las distintas tecnologías para sacarles el máximo partido y mejorar la eficiencia y la calidad de la aplicación. Así, a lo largo de todo el desarrollo se han buscado las mejores funciones que nos ofrece JavaScript que nos pudieran ayudar con cada una de las funcionalidades a implementar. También se estudian las mejores librerías disponibles para utilizar en la aplicación de análisis de datos, a través de Node.js, una potentísima herramienta con un catálogo de librerías muy amplio, el cual si se estudia a fondo puede mejorar nuestros desarrollos de manera exponencial. Por otro lado, estudiar las mejores estructuras y elementos que nos ofrece HTML y CSS, y las mejores combinaciones entre estas tecnologías, va a ser clave tanto para la eficiencia de la aplicación como para la presentación de la interfaz.



## Capítulo 5

# Diseño

---

### 5.0.1 Introducción

# Conclusións

---

**D**ERRADEIRO capítulo da memoria, onde se presentará a situación final do traballo, as leccións aprendidas, a relación coas competencias da titulación en xeral e a mención en particular, posibles liñas futuras,...

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique

neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

# **Apéndices**

## Material adicional

---

**E**XEMPLO de capítulo con formato de apéndice, onde se pode incluír material adicional que non teña cabida no corpo principal do documento, suxeito á limitación de 80 páxinas establecida no regulamento de TFGs.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique

neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.



# Bibliografía

---

- [1] C. N. de los Mercados y la Competencia, “Dos de cada tres hogares que consumen contenidos audiovisuales online de pago usan más de una plataforma,” 2023. [En línea]. Disponible en: <https://www.cnmc.es/prensa/panel-hogares-audiovisual-20231215>
- [2] B. Comunicación, “Análisis sobre la evolución de otts en acceso población,” 2023. [En línea]. Disponible en: <https://barloventocomunicacion.es/barometrotv-ott/analisis-sobre-la-evolucion-de-otts-en-acceso-poblacion/>
- [3] —, “6ª ola junio 2024 barÓmetro ott - multidispositivo,” 2024. [En línea]. Disponible en: <https://barloventocomunicacion.es/barometrotv-ott/6a-ola-junio-2024-barometro-ott-multidispositivo/>
- [4] Wikipedia, “Servicio ott.” [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Servicio\\_OTT](https://es.wikipedia.org/wiki/Servicio_OTT)
- [5] —, “Arquitectura de software.” [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Arquitectura\\_de\\_software](https://es.wikipedia.org/wiki/Arquitectura_de_software)
- [6] —, “Microservicio.” [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Arquitectura\\_de\\_microservicios](https://es.wikipedia.org/wiki/Arquitectura_de_microservicios)
- [7] —, “Experiencia de usuario.” [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Experiencia\\_de\\_usuario](https://es.wikipedia.org/wiki/Experiencia_de_usuario)
- [8] —, “Tizen.” [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Tizen>
- [9] —, “Webos.” [En línea]. Disponible en: <https://es.wikipedia.org/wiki/WebOS>
- [10] —, “Android tv.” [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Android\\_TV](https://es.wikipedia.org/wiki/Android_TV)
- [11] —, “Apache cordova.” [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Apache\\_Cordova](https://es.wikipedia.org/wiki/Apache_Cordova)



- [12] T. C. T. M. Association, “Hisense’s vidaa is now the no. 2 smart tv os globally behind samsung tizen, trade group says (chart of the day).” [En línea]. Disponible en: <https://www.nexttv.com/news/hisenses-vidaa-is-now-the-no-2-smart-tv-os-globally-behind-samsung-tizen-trade-group-says-chart-c>