



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE LA  
TELECOMUNICACIÓN

Curso Académico 2018/2019

Trabajo Fin de Grado

VISUALIZACIÓN DE DATOS EN REALIDAD  
VIRTUAL

Autor : Adrián Pizarro Serrano

Tutor : Dr. Jesús María González Barahona



# **Trabajo Fin de Grado**

Visualización de Datos en Realidad Virtual

**Autor :** Adrián Pizarro Serrano

**Tutor :** Dr. Jesús María González Barahona

La defensa del presente Proyecto Fin de Carrera se realizó el día                      de  
de 2019, siendo calificada por el siguiente tribunal:

**Presidente:**

**Secretario:**

**Vocal:**

y habiendo obtenido la siguiente calificación:

**Calificación:**

Fuenlabrada, a                      de                      de 2019



*Dedicado a  
mi familia*



# Agradecimientos

Aquí vienen los agradecimientos... Aunque está bien acordarse de la pareja, no hay que olvidarse de dar las gracias a tu madre, que aunque a veces no lo parezca disfrutará tanto de tus logros como tú... Además, la pareja quizás no sea para siempre, pero tu madre sí.





# Resumen

Aquí viene un resumen del proyecto. Ha de constar de tres o cuatro párrafos, donde se presente de manera clara y concisa de qué va el proyecto. Han de quedar respondidas las siguientes preguntas:

- ¿De qué va este proyecto? ¿Cuál es su objetivo principal?
- ¿Cómo se ha realizado? ¿Qué tecnologías están involucradas?
- ¿En qué contexto se ha realizado el proyecto? ¿Es un proyecto dentro de un marco general?

Lo mejor es escribir el resumen al final.



# Summary

Here comes a translation of the “Resumen” into English. Please, double check it for correct grammar and spelling. As it is the translation of the “Resumen”, which is supposed to be written at the end, this as well should be filled out just before submitting.



# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Contexto . . . . .	1
1.2. Motivación . . . . .	2
1.3. Estructura de la memoria . . . . .	3
<b>2. Objetivos</b>	<b>5</b>
2.1. Objetivo general . . . . .	5
2.2. Objetivos específicos . . . . .	5
2.3. Disponibilidad del Software . . . . .	7
<b>3. Estado del arte</b>	<b>9</b>
3.1. Sección 1 . . . . .	9
<b>4. Diseño e implementación</b>	<b>11</b>
4.1. Arquitectura general . . . . .	11
<b>5. Resultados</b>	<b>13</b>
<b>6. Conclusiones</b>	<b>15</b>
6.1. Consecución de objetivos . . . . .	15
6.2. Aplicación de lo aprendido . . . . .	15
6.3. Lecciones aprendidas . . . . .	15
6.4. Trabajos futuros . . . . .	16
<b>A. Manual de usuario</b>	<b>17</b>



# Índice de figuras

4.1. Estructura del parser básico . . . . .	12
---	----





# Capítulo 1

## Introducción

Este proyecto trata sobre la representación de datos en tres dimensiones (a partir de ahora 3D) en un navegador web utilizando como base el *framework* A-Frame<sup>1</sup>. Este *framework* nace a su vez a partir de la librería Three.js<sup>2</sup> utilizada para mostrar gráficos animados en 3D.

El objetivo principal es crear una librería capaz de visualizar datos con distintos tipos de gráficos, fácilmente escalable, mantenible y sencilla de utilizar. Además, como objetivos más específicos, podrá ser usada en cualquier navegador y dispositivo de realidad virtual.

A continuación se describe el contexto, motivación y la estructura de esta memoria.

### 1.1. Contexto

Hoy en día, prácticamente todo negocio se lleva a cabo o se promociona a través de Internet. La comunicación y la rentabilidad empresarial han mejorado enormemente gracias a la llegada de este fenómeno y de su mano surge el desarrollo web, el cual es uno de los mayores creadores de nuevos puestos de trabajo.

Para todo este paradigma surgen, casi a diario, nuevos *frameworks* para conseguir una mayor eficiencia y estandarización a la hora de crear páginas webs.

Cabe destacar que, a nivel empresarial, es necesario un estudio exhaustivo así como una mejor visualización, tratamiento y explotación de los datos. Para ello han surgido infinidad de

---

<sup>1</sup><https://aframe.io/>

<sup>2</sup><https://threejs.org/>

librerías a nuestra disposición tales como D3.js<sup>3</sup>, amcharts<sup>4</sup>, highcharts<sup>5</sup>, etc.

Por otro lado, con el nacimiento de Canvas 3D en 2006 y su posterior evolución en WebGL se abre la puerta a la implementación de visualizaciones 3D en páginas webs. Como veremos más adelante, WebGL nos proporciona un API en javascript para renderizar visualizaciones 2D y 3D sobre HTML.

Paralelamente, nace el motor gráfico Unity el cual también ha ayudado a impulsar la fama y utilización de componentes 3D en múltiples plataformas. Ha sentado varias bases y conceptos reutilizados en muchas otras plataformas tales como el concepto de escena, cámaras, materiales, luces, sombras, formas geométricas y parámetros para cada componente.

Más tarde, y concerniente a este proyecto, nace Three.js como librería javascript basada en WebGL para crear y mostrar visualizaciones 3D. Esta librería ya nos proporciona todas las especificaciones necesarias para trabajar con escenas, efectos, animaciones, cámaras, etc. Para un uso más sencillo y estándar de esta librería y para su adaptación a la realidad Virtual nace A-frame en 2015. Gracias a esta última librería podemos utilizar todas nuestras escenas 3D y representarlas en lo que conocemos como WebVR, la cual es una interfaz que nos provee del soporte necesario para visualizar todo este contenido HTML 3D en un aparato de realidad virtual (Oculus Rift, HTC Vive y Google Cardboard) o un navegador.

Finalmente me gustaría hacer una mención especial a toda la comunidad que hay detrás de todas estas plataformas. Sin su aportación diaria, intercambio de conocimiento y ayuda nada de esto sería posible.

La unión de estos dos mundos, representación de datos y la llegada de visualizaciones 3D en navegadores web, nos lleva a nuestro siguiente apartado la motivación.

## 1.2. Motivación

Muchas veces se escucha la frase atribuida al filósofo inglés Francis Bacon “La información es poder”, la cual es cierta, pero sin una correcta interpretación de la misma podemos caer en el caso de la desinformación. Como se menciona en el apartado anterior la visualización y explotación de datos es un campo en auge del cual se requiere mucha ingeniería para poder

---

<sup>3</sup><https://d3js.org/>

<sup>4</sup><https://www.amcharts.com/>

<sup>5</sup><https://www.highcharts.com/>

sacarle partido. Es un campo apasionante, lleno de retos, con muchas salidas laborales y en constante evolución.

Hasta ahora, tenemos infinidad de librerías para representar datos en 2D pero ¿Qué ocurre con la llegada del 3D a nuestros navegadores? ¿Existen librerías para visualizar datos utilizando realidad virtual o realidad aumentada? Lo cierto es que estas preguntas nos llevan a una de las motivaciones principales de este proyecto. Este mundo es relativamente nuevo y hasta ahora se ha apostado más por realizar visualizaciones complejas en 3D o videojuegos. Por lo que la representación de datos en 3D es un nicho aun por explotar.

Este último punto ha sido muy motivador para realizar una de las primeras librerías capaz de visualizar datos en 3D tanto en un navegador como en un dispositivo de realidad virtual. Además, la intención de este proyecto es que sea escalable, mantenible y el resto de la comunidad pueda contribuir a su uso y desarrollo.

Respecto a la motivación personal podemos decir que se ha pasado por distintas fases. En primer lugar, hubo un intento fallido de realizar una aplicación de posible utilidad médica. Siempre se puede interpretar como poner excusas, pero el tiempo, obligaciones y el querer dedicar tu tiempo libre a otros menesteres son factores que impidieron la consecución de este primer proyecto el cual, a pesar de todo, era muy interesante. Todo esto llevó a volver a comenzar con una búsqueda exhaustiva y motivadora donde por suerte pude contactar con Jesús, tutor del proyecto, el cual me ofreció toda la ayuda y un trabajo de fin de grado por el cual no me importara dedicar todo el tiempo libre posible.

Es así como gracias a todos estos factores se ha podido finalizar este proyecto y escritura de esta memoria. De la cual procedemos a describir su estructura en la siguiente sección.

## **1.3. Estructura de la memoria**

En esta sección se describe la estructura de la memoria para una mejor comprensión de la misma:

- En el primer capítulo se hace una breve introducción al proyecto. Describiremos el marco y contexto del trabajo para un mejor aterrizaje del lector. Acto seguido se habla de la motivación del proyecto tanto a nivel personal como profesional. Por último se describe brevemente la estructura de la memoria.

- En el capítulo 2 se muestran los objetivos del proyecto. Se comienza hablando del objetivo principal que perseguimos con este trabajo. Acto seguido, pasaremos a los específicos donde se profundizará y se ampliará los conceptos tratados en el objetivo principal. Además en este apartado se hará una descripción de la planificación temporal que se ha seguido para llevar a cabo este proyecto
- A continuación se presenta el estado del arte. Aquí haremos un repaso así como una breve descripción de todas las tecnologías que se han utilizado.
- En el cuarto capítulo se desarrolla el diseño e implementación. Se entrará al detalle de como está construida la aplicación, su arquitectura y sus distintos componentes.
- En este quinto capítulo, Resultados, se realiza un análisis del funcionamiento y rendimiento de esta aplicación. Además se podrá ver un amplio abanico de casos de uso donde se verán los resultados y capacidades del proyecto.
- Por último, tendremos las conclusiones. Aquí se presentará un resumen de los distintos conceptos aprendidos y aplicados. Podremos ver si se han alcanzado los objetivos propuestos. Además se expondrán líneas futuras de investigación y mejora de esta aplicación.

# Capítulo 2

## Objetivos

### 2.1. Objetivo general

Mi trabajo fin de grado consiste en crear una librería para la visualización de datos en 3D compatible con cualquier navegador. La cual además pueda ser usada en un dispositivo de realidad virtual.

### 2.2. Objetivos específicos

En este apartado se realiza una breve descripción de los objetivos específicos.

1. La librería debe proporcionar varios tipos de visualizaciones. Deberá implementar al menos cinco tipos de visualización tales como gráfico de burbujas, de tarta, de barras, de cilindros o tarta en forma de rosquilla.
2. Se debe permitir una gran cantidad de parametrización. Por ejemplo la posición, color, longitud y separación entre las marcas de medición de los ejes. Permitir que los ejes puedan ser negativos o estén en forma de rejilla para 2 o más dimensiones. Poder configurar tamaño y color de una leyenda, un pop-up o ambas.
3. La librería debe ser capaz de añadir a un gráfico una leyenda y/o un pop-up para una mejor interpretación de los datos. Con ello se persigue obtener más información o facilitar la lectura de datos.

4. La librería proporcionará una herramienta para filtrar o seleccionar distintas fuentes de datos. Además podrá refrescar las visualizaciones dinámicamente.
5. Debe ser capaz de interpretar ficheros con formato JSON y lanzar una excepción si no fuera capaz de leerlo correctamente.
6. El rendimiento es un factor clave que se desea cuidar en este proyecto. Por ello, esta librería debe mostrar un buen comportamiento cargando grandes volúmenes de datos. Se debe comprobar que es capaz de mostrar visualizaciones con una gran cantidad de datos
7. Se debe poder utilizar en cualquier navegador sin necesidad de instalar un plugin. Además se debe poder utilizar y visualizar en dispositivos de realidad virtual como las Oculus Rift.
8. La librería se debe distribuir y poner al servicio de la comunidad. Se debe dar fácil acceso y ser subida a un repositorio de código donde se puedan recibir sugerencias para implementar y resolver posibles incidencias.
9. Debe ser fácilmente escalable, es decir, deberá estar desarrollada de tal manera que sea sencillo implementar nuevos desarrollos o modificaciones en el código.
10. Un objetivo importante es que el código deberá ser mantenible tanto por su autor como por la comunidad. Se debe publicar dicho software en una plataforma donde, como se menciona anteriormente, se pueda contribuir a resolver incidencias y mantener actualizado de manera sencilla las versiones de los framework y librerías que utilizará el proyecto.
11. Aprender Webpack para empaquetar y distribuir nuestra librería.
12. Aprender el framework de A-frame. Se debe mirar en profundidad su documentación y estándares para un uso óptimo de la misma. Esta librería de realidad virtual está basada en Three.js por lo que también se deberá realizar un estudio de la misma, crear una web en github
13. Crear una web con ejemplos de uso de la librería.
14. Crear el proyecto con Angle. Una herramienta hecha por los creadores de Aframe para comenzar nuestro componente con un arquetipo que sigue los estándares de Aframe

15. Una vez creado el proyecto siguiendo los estándares y después de realizar los primeros avances se deberá publicar nuestro proyecto en el registry de Aframe donde aparecen todos los componentes de la comunidad de A-frame. Esto proporcionará visibilidad al proyecto pudiendo salir en el blog oficial llamado Week of a- Frame
16. Se deberá cuidar y refactorizar el código para una lectura sencilla del mismo siguiendo distintas directrices que podemos encontrar en el libro Clean Code [2].

## **2.3. Disponibilidad del Software**

Aqui ponemos la web, y el repo de github etc





# Capítulo 3

## Estado del arte

Descripción de las tecnologías que utilizas en tu trabajo. Con dos o tres párrafos por cada tecnología, vale. Se supone que aquí viene todo lo que no has hecho tú.<sup>o</sup>

Puedes citar libros, como el de Bonabeau et al. sobre procesos estigmérgicos [1].

También existe la posibilidad de poner notas al pie de página, por ejemplo, una para indicarte que visite la página de LibreSoft<sup>1</sup>.

### 3.1. Sección 1

Hemos hablado de cómo incluir figuras. Pero no hemos dicho nada de tablas. A mí me gustan las tablas. Mucho. Aquí un ejemplo de tabla, la Tabla 3.1.

---

<sup>1</sup><http://www.libresoft.es>

1	2	3
4	5	6
7	8	9

Cuadro 3.1: Ejemplo de tabla



# Capítulo 4

## Diseño e implementación

Aquí viene todo lo que has hecho tú (tecnológicamente). Puedes entrar hasta el detalle. Es la parte más importante de la memoria, porque describe lo que has hecho tú. Eso sí, normalmente aconsejo no poner código, sino diagramas.

### 4.1. Arquitectura general

Si tu proyecto es un software, siempre es bueno poner la arquitectura (que es cómo se estructura tu programa a “vista de pájaro”).

Por ejemplo, puedes verlo en la figura 4.1.

Si utilizas una base de datos, no te olvides de incluir también un diagrama de entidad-relación.

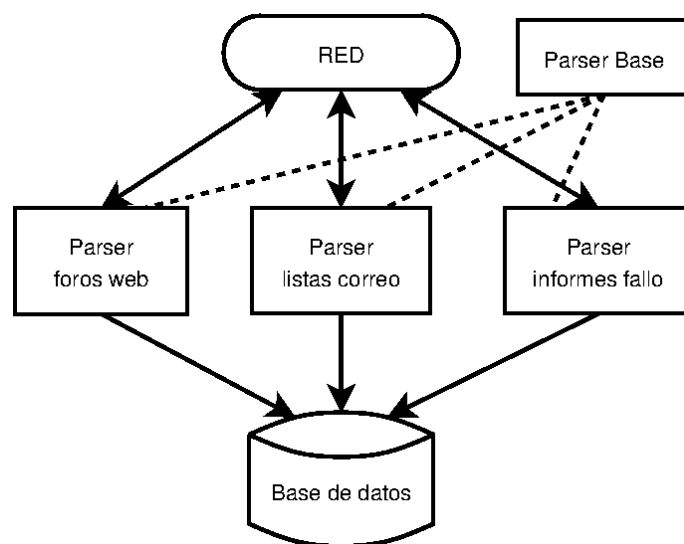


Figura 4.1: Estructura del parser básico

# Capítulo 5

## Resultados

En este capítulo se incluyen los resultados de tu trabajo fin de grado.

Si es una herramienta de análisis lo que has realizado, aquí puedes poner ejemplos de haberla utilizado para que se vea su utilidad.



# Capítulo 6

## Conclusiones

### 6.1. Consecución de objetivos

Esta sección es la sección espejo de las dos primeras del capítulo de objetivos, donde se planteaba el objetivo general y se elaboraban los específicos.

Es aquí donde hay que debatir qué se ha conseguido y qué no. Cuando algo no se ha conseguido, se ha de justificar, en términos de qué problemas se han encontrado y qué medidas se han tomado para mitigar esos problemas.

### 6.2. Aplicación de lo aprendido

Aquí viene lo que has aprendido durante el Grado/Máster y que has aplicado en el TFG/TFM. Una buena idea es poner las asignaturas más relacionadas y comentar en un párrafo los conocimientos y habilidades puestos en práctica.

1. a

2. b

### 6.3. Lecciones aprendidas

Aquí viene lo que has aprendido en el Trabajo Fin de Grado/Máster.

1. a

2. b

## **6.4. Trabajos futuros**

Ningún software se termina, así que aquí vienen ideas y funcionalidades que estaría bien tener implementadas en el futuro.

Es un apartado que sirve para dar ideas de cara a futuros TFGs/TFM.



# **Apéndice A**

## **Manual de usuario**



# Bibliografía

- [1] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, Inc., 1999.
- [2] R. C. Martin. *Clean Code: A Handbook of Agile Software Craftsmanship*. Prentice Hall, 2008.