



**Escuela Superior  
de Ingeniería y Tecnología**  
Universidad de La Laguna

# Trabajo de Fin de Grado

Grado en Ingeniería Informática

---

## Herramienta de visualización de datos

*Data visualization tool*

Nelson Noda Pérez

---

La Laguna, 5 de julio de 2023

D. **Carina Soledad González González**, con N.I.F. 54.064.251-Z profesora Catedrática de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutora.

D. **Jezabel Molina Gil**, con N.I.F. 78.507.682-B profesora Ayudante Doctora de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutora.

## **C E R T I F I C A ( N )**

Que la presente memoria titulada:

*“Herramienta de visualización de datos”*

ha sido realizada bajo su dirección por D. **Nelson Noda Pérez**,  
con N.I.F. 79.151.433-S.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 5 de julio de 2023

## Agradecimientos

Quiero expresar mi profundo agradecimiento a todas las personas que me apoyaron en mi camino hacia la finalización de mi TFG. En primer lugar, quiero agradecer a mi familia por estar a mi lado en cada paso del camino. Me brindaron su incondicional apoyo, lo cual me dio la fuerza y la motivación necesarias para seguir adelante cuando las cosas se pusieron difíciles.

También quiero expresar mi agradecimiento a mis compañeros de ingeniería informática, quienes siempre estuvieron ahí para ayudarme a entender temas complejos, responder mis preguntas y hacer de esta experiencia una aventura verdaderamente inolvidable. Su ayuda y amistad significan mucho para mí y me siento afortunado de tenerlos en mi vida.

# Licencia

\* Si NO quiere permitir que se compartan las adaptaciones de tu obra y NO quieres permitir usos comerciales de tu obra indica:



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional.

## **Resumen**

*El proyecto se centra en ofrecer una plataforma interactiva para visualizar y explorar estos datos de manera intuitiva y personalizada.*

*El objetivo de este trabajo ha sido crear una aplicación web utilizando React.js y Recharts con el objetivo de visualizar datos de Women@INF en gráficas dinámicas. Women@INF es una fuente de datos que proporciona información sobre diversas iniciativas y proyectos liderados por mujeres. El desarrollo de esta aplicación web es fundamental debido a la necesidad de tener una plataforma dedicada para visualizar y analizar sus datos de manera interactiva en Women@INF. La inexistencia de una herramienta específica dificulta el acceso y el aprovechamiento óptimo de la información disponible. Esta aplicación contribuye a la visibilización de las iniciativas lideradas por mujeres, promueve la igualdad de género y fomenta la colaboración entre comunidades autónomas. Al ofrecer una herramienta interactiva y accesible para la visualización y el análisis de datos, se impulsa el conocimiento y se generan oportunidades para empoderar a las mujeres y promover el cambio social hacia una sociedad más equitativa.*

*El principal objetivo de este proyecto es proporcionar a los usuarios la capacidad de seleccionar y visualizar los datos de Women@INF en diferentes tipos de gráficas utilizando React.js y Recharts. Esto les permite analizar y comprender mejor la información, identificar patrones y tendencias, y obtener conocimientos valiosos sobre las iniciativas lideradas por mujeres.*

*Para lograr este objetivo, se utilizó React.js como el framework principal para el desarrollo de la interfaz de usuario de la página web. Con la ayuda de Recharts, una biblioteca de gráficos basada en React, se implementó la funcionalidad de representación gráfica de los datos. Recharts proporciona una amplia gama de componentes de gráficos altamente personalizables y fáciles de usar, lo que me ha permitido crear visualizaciones interactivas y atractivas para los usuarios.*

*Además del uso de React.js y Recharts, también se usó Express.js para el backend y MongoDB Atlas como base de datos. Se implementó un sistema de inicio de sesión y registro de usuarios para garantizar la seguridad y la personalización de la experiencia. Asimismo, se ha desarrollado una interfaz de administrador que permite gestionar las iniciativas recibidas validándolas, actualizándolas o eliminándolas según sea necesario.*

*Las principales contribuciones de este trabajo incluyen:*

- Una página web funcional desarrollada con React.js que permite a los usuarios seleccionar y visualizar datos de Women@INF en gráficas dinámicas.*
- El uso de Recharts para implementar visualizaciones de datos interactivas y altamente personalizables, lo que facilita la exploración y el análisis de la información.*

- *Un sistema de administración que permite a los administradores gestionar las iniciativas recibidas, asegurando su validez y actualización además de administrar los datos disponibles para su representación gráfica.*
- *La posibilidad de que los usuarios publiquen sus propias iniciativas en el mapa y filtren las existentes según su estado y ubicación geográfica.*

**Palabras clave:** React, Rechart, Mongoddb,Leaflet, React-Leaflet,JWT,Express,visualización de datos

## **Abstract**

*The project focuses on providing an interactive platform for visualizing and exploring these data in an intuitive and personalized manner.*

*The aim of this work has been to create a web application using React.js and Recharts with the purpose of visualizing Women@INF data in dynamic charts. Women@INF is a data source that provides information about various initiatives and projects led by women. The development of this web application is crucial due to the need for a dedicated platform to visualize and analyze their data interactively in Women@INF. The lack of a specific tool hinders access to and optimal utilization of the available information. This application contributes to the visibility of initiatives led by women, promotes gender equality, and fosters collaboration among autonomous communities. By offering an interactive and accessible tool for data visualization and analysis, it drives knowledge and generates opportunities to empower women and promote social change towards a more equitable society.*

*The main objective of this project is to provide users with the ability to select and visualize Women@INF data in different types of charts using React.js and Recharts. This allows them to analyze and better understand the information, identify patterns and trends, and gain valuable insights into initiatives led by women.*

*To achieve this goal, React.js was used as the main framework for developing the web page's user interface. With the help of Recharts, a React-based charting library, the functionality of graphical representation of data was implemented. Recharts provides a wide range of highly customizable and easy-to-use chart components, enabling me to create interactive and visually appealing visualizations for users.*

*In addition to using React.js and Recharts, Express.js was also used for the backend, and MongoDB Atlas was used as the database. A user login and registration system was implemented to ensure security and customization of the experience. Furthermore, an administrator interface was developed, allowing administrators to manage received initiatives by validating, updating, or deleting them as needed.*

*The main contributions of this work include:*

- A functional web page developed with React.js that allows users to select and visualize Women@INF data in dynamic charts.*
- The use of Recharts to implement interactive and highly customizable data visualizations, facilitating exploration and analysis of the information.*

- *An administration system that enables administrators to manage received initiatives, ensuring their validity and updates, as well as managing the data available for graphical representation.*
- *The ability for users to publish their own initiatives on the map and filter existing ones based on their status and geographical location.*

**Keywords:** React, Rechart, Mongoddb,Leaflet, React-Leaflet,JWT, Express,data visualization



# Índice general

|  |           |
|--|-----------|
| <b>Capítulo 1 Introducción.....</b>                                  | <b>1</b>  |
| 1.1 Breve introducción.....  | 1         |
| 1.2 Antecedentes y estado actual.....                                | 2         |
| <b>Capítulo 2 Solución tecnológica.....</b>                          | <b>3</b>  |
| 2.1 Frontend.....  | 3         |
| 2.1.1 Visualización de datos.....                                    | 5         |
| 2.1.2 Mapa de iniciativas.....                                       | 6         |
| 2.1.3 Despliegue.....  | 6         |
| 2.2 Backend.....   | 7         |
| 2.3 Base de datos.....   | 9         |
| <b>Capítulo 3 Control de sesión.....</b>                             | <b>11</b> |
| 3.1 Tecnología utilizada.....  | 11        |
| <b>Capítulo 4 Creación de gráfica.....</b>                           | <b>13</b> |
| 4.1 Publicación de datos.....  | 13        |
| 4.2 Obtención de datos.....  | 15        |
| 4.2.1 Obtención de datos desde base de datos.....                    | 15        |
| 4.2.2 Procesamiento de datos para representarlos en una gráfica..... | 16        |
| 4.3 Renderizado de gráfica.....                                      | 18        |
| <b>Capítulo 5 Backend.....</b>                                       | <b>20</b> |
| 5.1 Rutas de la aplicación.....                                      | 20        |
| 5.1.1 Rutas de usuario.....  | 21        |
| 5.1.2 Rutas de iniciativas.....                                      | 22        |
| 5.1.3 Rutas de datos.....  | 22        |

|  |           |
|--|-----------|
| <b>Capítulo 6 Base De datos.....</b>                 | <b>25</b> |
| 6.1 Creación de modelos de datos.....                | 25        |
| 6.1.1 Esquema de usuario.....                        | 26        |
| 6.1.2 Esquema de Iniciativa.....                     | 27        |
| 6.1.3 Esquema de fichero de configuración.....       | 27        |
| 6.1.4 Datos sin esquema.....                         | 27        |
| <b>Capítulo 7 Mapa de iniciativas.....</b>           | <b>29</b> |
| 7.1 Obtención de iniciativas.....                    | 29        |
| 7.2 Representación de iniciativas.....               | 30        |
| 7.3 Formulario.....                                  | 32        |
| 7.3.1 Usuario común.....                             | 32        |
| 7.3.2 Administrador.....                             | 33        |
| <b>Capítulo 8 Resultados.....</b>                    | <b>34</b> |
| 8.1 Usuario común.....                               | 35        |
| 8.2 Administrador.....                               | 38        |
| <b>Capítulo 9 Conclusiones y líneas futuras.....</b> | <b>42</b> |
| <b>Capítulo 10 Summary and Conclusions.....</b>      | <b>43</b> |
| <b>Capítulo 11 Presupuesto.....</b>                  | <b>44</b> |
| 11.1 Presupuesto desglosado.....                     | 44        |
| <b>Capítulo 12 Apéndice A.....</b>                   | <b>45</b> |
| 12.1 Repositorios.....                               | 45        |

# Índice de figuras

|   |    |
|---|----|
| Figura 2.1: Popularidad de react frente a otras tecnologías.....                  | 3  |
| Figura 2.2: Popularidad de react frente a otras tecnologías.....                  | 4  |
| Figura 2.3: Uso de react frente a otras tecnologías.....                          | 5  |
| Figura 2.4: Descargas de mongoose y mongodb frente a otras bases de datos.....    | 9  |
| Figura 4.1: Barra de navegación cuando el administrador inicia sesión.....        | 13 |
| Figura 4.2: Imagen del administrador de csv.....                                  | 14 |
| Figura 4.3: Imagen del selector de campos.....                                    | 14 |
| Figura 4.4: Imagen del selector de gráficas por defecto por el administrador..... | 15 |
| Figura 4.5: Imagen del selector de ponderación.....                               | 15 |
| Figura 4.6: Imagen del gráfico con los datos a seleccionar.....                   | 16 |
| Figura 4.7: Ejemplo de estructura antes de hacer las transformaciones.....        | 16 |
| Figura 4.8: Ejemplo de estructura después de las transformaciones.....            | 17 |
| Figura 4.9: Imagen de gráfico con los campos ordenados de menor a mayor.....      | 18 |
| Figura 4.10: Imagen del código que controla qué gráfica se muestra.....           | 18 |
| Figura 4.11: Imagen que muestra cómo se hizo responsive las gráficas.....         | 19 |
| Figura 6.1: Ejemplo de un schema en mongoose.....                                 | 27 |
| Figura 7.1: Imagen del mapa de iniciativas.....                                   | 30 |

|  |    |
|--|----|
| Figura 7.2: lógica para diferenciar entre administrador y usuario común en el formulario de iniciativas..... | 31 |
| Figura 7.3: Imagen de como mostrar los datos en el popup.....  | 32 |
| Figura 7.4: Imagen de un popup.....  | 32 |
| Figura 7.5: Imagen del formulario de iniciativas.....  | 33 |
| Figura 7.6: Imagen del formulario de iniciativas para el administrador.....                                  | 33 |
| Figura 8.1: Gráfica circular.....  | 35 |
| Figura 8.2: Gráfica de barras.....   | 35 |
| Figura 8.3: Gráfica de barras apiladas.....  | 36 |
| Figura 8.4: Gráfica de barras horizontales.....  | 36 |
| Figura 8.5: Gráfica de barras horizontales apiladas.....   | 37 |
| Figura 8.6: Mapa de iniciativas para usuario común.....  | 37 |
| Figura 8.7: Formulario de publicación de iniciativa.....   | 38 |
| Figura 8.8: Selector de ejes.....  | 39 |
| Figura 8.9: Administrador de ponderación.....  | 39 |
| Figura 8.10: Selector de gráfica por defecto.....  | 39 |
| Figura 8.11: Administrador de descripción del conjunto de datos.....   | 39 |
| Figura 8.12: Administrador de ficheros csv.....  | 40 |
| Figura 8.13: Mapa de iniciativas del administrador.....  | 40 |
| Figura 8.14: Administrador de iniciativas.....   | 41 |

# Índice de tablas

|                                      |    |
|--------------------------------------|----|
| Tabla 5.1: Rutas de usuario.....     | 22 |
| Tabla 5.2: Rutas de iniciativas..... | 22 |
| Tabla 5.3: Rutas de datos.....       | 24 |
| Tabla 11.1: Presupuesto.....         | 44 |

# Capítulo 1

## Introducción

### 1.1 Breve introducción

El objetivo de este proyecto es crear una herramienta que permita a los usuarios comprender de manera más efectiva los desafíos que enfrentan las mujeres en el campo de la informática. La brecha de género es un problema significativo en esta industria, y es fundamental abordarlo para lograr una mayor equidad [1-5].

Una de las problemáticas que se aborda en este proyecto es la visualización de datos. Actualmente, muchos informes y datos relacionados con la brecha de género en la informática se presentan de manera estática, lo que dificulta su comprensión y análisis adecuados. La falta de herramientas que aborden cómo entender y visualizar de manera efectiva estos datos ha sido un obstáculo para comprender plenamente la magnitud del problema [6].

Con el desarrollo de esta página web, se busca superar estas limitaciones y proporcionar una plataforma interactiva y atractiva para visualizar los datos recopilados por el Proyecto de Investigación "La investigación en Informática en España: Visualizando y visibilizando la labor de las mujeres (WOMEN@INF)", Ref. 21/1ACT/21. 2022-2023. Instituto de las Mujeres del Ministerio de Igualdad (<https://www.women-inf.eu/>). Esto permitirá a las personas usuarias explorar y comprender mejor la brecha de género en el campo de la informática, identificar patrones y tendencias, y obtener conocimientos más profundos sobre la situación actual.

Además, se busca fomentar la participación ciudadana y hacer que los usuarios sean parte activa de la solución. Para ello, se creará un mapa de iniciativas disponibles donde las personas podrán informarse sobre las diferentes propuestas y también contribuir añadiendo aquellas que no estén recogidas. Esto promoverá la colaboración y la difusión de acciones que busquen cerrar la brecha de género en la informática.

Este proyecto no solo busca abordar la problemática de la brecha de género en la informática, sino también mejorar la visualización de datos y promover la participación ciudadana.

## 1.2 Antecedentes y estado actual

Para el desarrollo de la aplicación se han observado otros sistemas de visualización como el de la [Universidad de Salamanca](#) [7] en el ámbito de los datos gráficos y el mapa de [en negro contra las violencias](#) [8].

Antes de la creación de la herramienta que se presenta en este trabajo, no existía una solución completa y eficiente para la visualización de los datos recogidos por Women@INF. La falta de una plataforma para presentar y analizar estos datos puede haber dificultado la toma de decisiones informadas sobre la igualdad de género y la discriminación de género. La creación de una herramienta de visualización de datos personalizada y adaptable como la presentada en este trabajo puede ser una solución importante para abordar esta necesidad.

Además, la falta de una plataforma de mapeo que permita la publicación y visualización de iniciativas relacionadas con la igualdad de género y la discriminación de género puede haber obstaculizado la participación ciudadana y la divulgación de información importante sobre estos temas. La creación de un mapa interactivo donde los usuarios pueden agregar iniciativas y visualizarlas, puede ayudar a promover la participación ciudadana y fomentar el compromiso con estos temas.

Para abordar estos desafíos, se pueden emplear herramientas consolidadas como Microsoft Power BI. Esta solución es utilizada en la Universidad de Salamanca para la visualización de gráficos. Power BI ofrece una amplia gama de capacidades, incluyendo la creación de informes interactivos y la generación de paneles de control.

Para resolver el problema relacionado con los mapas interactivos, una opción popular es la API de Google Maps. Esta herramienta permite la integración de mapas interactivos en aplicaciones web y ofrece funciones como la presentación de ubicaciones, trazado de rutas y personalización de marcadores.

Ambas soluciones mencionadas anteriormente son ampliamente utilizadas y cuentan con documentación detallada y una comunidad de desarrollo activa.

Como se puede observar, hay herramientas adecuadas para la visualización de datos y la publicación de iniciativas en el ámbito de la igualdad de género y la discriminación de género. La implementación de una herramienta de visualización bien diseñada puede tener un impacto significativo en la conciencia y la acción en torno a la igualdad de género y la discriminación de género, ya que permite resaltar las disparidades existentes, las áreas problemáticas y las tendencias a lo largo del tiempo.

# Capítulo 2

## Solución tecnológica

En este apartado, analizaremos la opción tecnológica que hemos seleccionado para desarrollar nuestro proyecto basado en el stack MERN como se muestra en la figura 2.1. El stack MERN (MongoDB, Express.js, React.js, Node.js) es ampliamente reconocido y valorado en el ámbito de las aplicaciones web modernas debido a sus numerosas ventajas y las sinergias existentes entre las tecnologías que lo conforman. Exploraremos en detalle las razones por las cuales hemos optado por este stack y cómo cada una de sus componentes contribuye al éxito de nuestro proyecto.

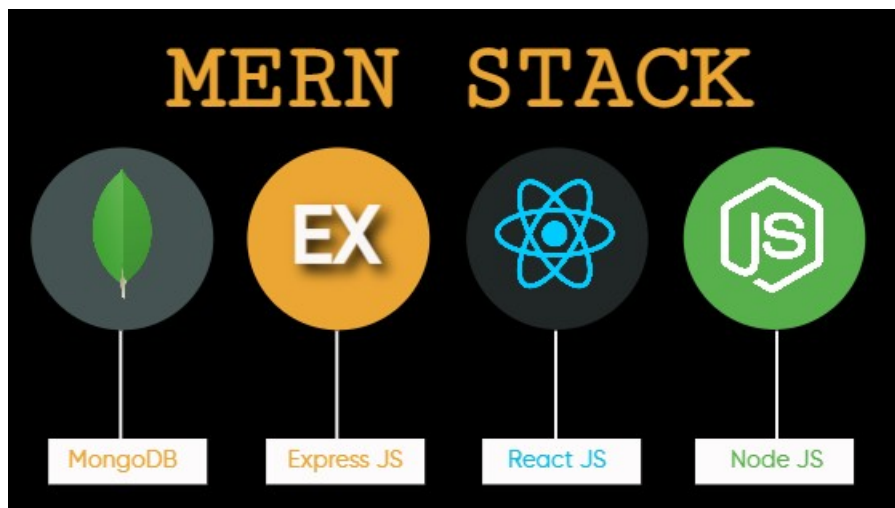


Figura 2.1: Popularidad de react frente a otras tecnologías

### 2.1 Frontend

La elección de React.js como la tecnología principal para nuestro proyecto se basó en varios factores clave.

En primer lugar, se trata de una de las bibliotecas más populares y ampliamente utilizadas como podemos observar en la figura 2.2 y la figura 2.3.



Además, cuenta con una comunidad activa, lo que facilita el aprendizaje de la tecnología, la resolución de problemas y el desarrollo eficiente.

Un aspecto destacado de React.js es su enfoque en la construcción modular de interfaces de usuario mediante el uso de componentes. Esto permite la creación de interfaces independientes y reutilizables, lo que a su vez facilita el mantenimiento y la escalabilidad del código.

React.js utiliza un enfoque de Virtual DOM (DOM virtual), lo que implica que las actualizaciones en la interfaz de usuario se realizan de manera eficiente. El Virtual DOM minimiza las actualizaciones en el DOM real, lo que se traduce en un mejor rendimiento y una mayor velocidad de la aplicación. Esta característica resulta especialmente valiosa en proyectos con aplicaciones complejas que manejan grandes cantidades de datos y cambios frecuentes en la interfaz de usuario como nuestro caso.

Otro punto a favor de React.js es su capacidad de integración con herramientas y bibliotecas externas. Esto brinda la posibilidad de ampliar las funcionalidades de React.js según las necesidades específicas del proyecto.

Además, React.js cuenta con un sólido respaldo tanto de Facebook, su creador, como de la comunidad de desarrolladores. Esta constante atención y apoyo aseguran que React.js se mantenga actualizado con nuevas versiones y mejoras continuas. El hecho de que Facebook utilice React.js en sus propias aplicaciones demuestra su capacidad para abordar proyectos a gran escala y de alto rendimiento.

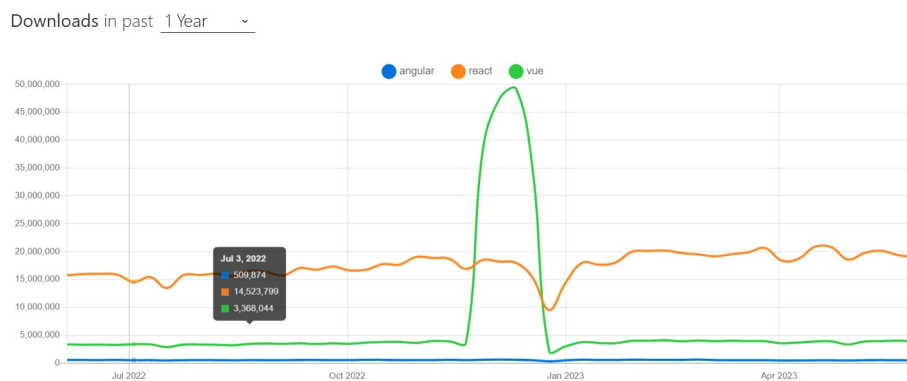


Figura 2.2: Popularidad de react frente a otras tecnologías

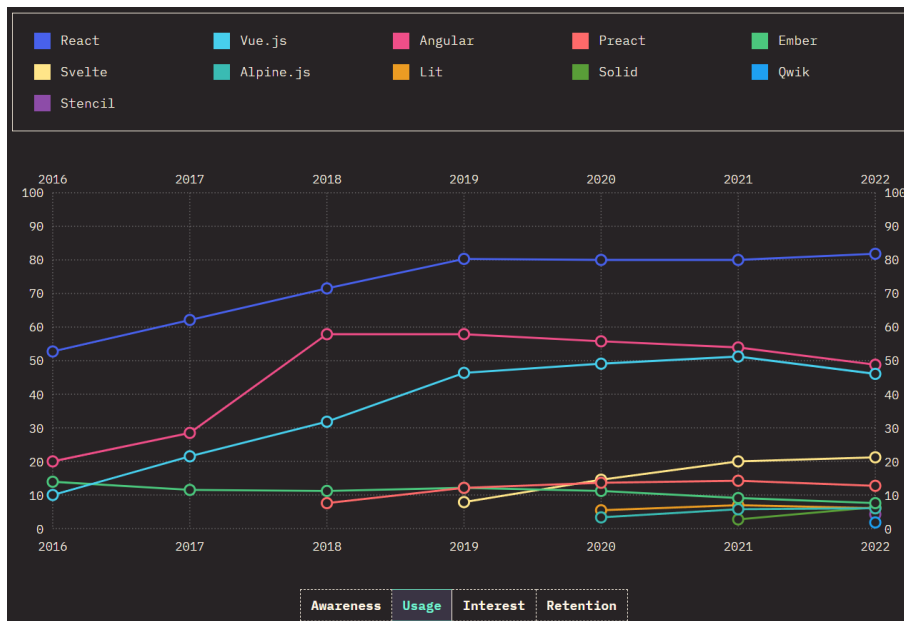


Figura 2.3: Uso de react frente a otras tecnologías

### 2.1.1 Visualización de datos

Para la representación gráfica se ha utilizado Rechart porque permite crear gráficos y visualizaciones de datos de manera fácil y rápida. Ofrece una variedad de tipos de gráficos y personalización para adaptarse a las necesidades del trabajo a realizar y funciona bien con Reactjs que es la base de nuestra aplicación web.

Una de sus principales ventajas es que no se requiere un conocimiento profundo para comenzar a utilizarla y obtener resultados visualmente atractivos.

Otra ventaja significativa es la amplia variedad de tipos de gráficos que ofrece. Con opciones que incluyen gráficos de líneas, barras, pastel, radar y más, brinda la flexibilidad necesaria para adaptarse a diversos requisitos de visualización de datos.

La personalización es otro aspecto destacado de esta herramienta. Los gráficos generados se pueden ajustar y personalizar según las necesidades específicas. Se pueden modificar los colores, agregar etiquetas y leyendas, y elegir estilos de fuente, lo que permite adaptar los gráficos al estilo y la identidad visual de cada proyecto.

Además, dado que está construida con React, esta herramienta se integra de manera fluida con aplicaciones web desarrolladas utilizando esta tecnología. Esto significa que es altamente compatible y permite una fácil integración con proyectos existentes basados en React, aprovechando las ventajas y funcionalidades adicionales que esta combinación ofrece.

En la visualización de datos, se decidió no utilizar Power BI debido a nuestra preferencia de tener un sistema altamente personalizable. Además, se tomó en cuenta que varias características de Power BI tienen un costo asociado. Optamos por desarrollar un sistema propio para tener un mayor control y no depender de terceros en este aspecto.

### **2.1.2 Mapa de iniciativas**

Para obtener los datos de las distintas localizaciones que se seleccionen en el mapa se eligió la api de de OpenStreetMap.

Uno de los aspectos más destacados es que los datos son de código abierto, lo que significa que están disponibles de forma gratuita y abierta para todos. Esto facilita el acceso a los datos y fomenta la colaboración y el desarrollo de soluciones innovadoras basadas en ellos.

La herramienta proporciona una gran flexibilidad, permitiendo a los desarrolladores llevar a cabo una amplia gama de operaciones en los datos geoespaciales. Esto incluye las operaciones específicas que necesitábamos para este proyecto, lo que brinda la libertad de trabajar con los datos de la manera más adecuada para nuestros requerimientos.

Además, esta herramienta se integra fácilmente con otras herramientas y servicios, en nuestro caso con Leaflet. La integración sin problemas con herramientas y servicios externos amplía aún más las capacidades y funcionalidades disponibles, permitiéndonos aprovechar al máximo el potencial de nuestro proyecto geoespacial.

Para la representación del mapa se eligió la antes comentada Leaflet pero se utiliza React-Leaflet, ya que está específicamente diseñado para trabajar de manera nativa con React, lo que facilita la integración de mapas interactivos en aplicaciones existentes sin necesidad de aprender una nueva sintaxis o tecnología. Esto agiliza el proceso de desarrollo y permite aprovechar la familiaridad y eficiencia de React.

Además, esta herramienta proporciona una amplia variedad de funciones para interactuar con los mapas. Esto incluye la capacidad de agregar marcadores para nuestras iniciativas, lo que nos permite visualizar y resaltar ubicaciones específicas en el mapa de manera fácil y personalizada.

Es importante destacar que se descartó Google Maps como opción debido a sus restricciones y limitaciones en el uso gratuito, mientras que esta herramienta, basada en OpenStreetMap, ofrece una solución gratuita y de código abierto sin las restricciones asociadas.

### **2.1.3 Despliegue**

Para el despliegue del fronted se eligió Netlify por su facilidad de uso. Con una experiencia de despliegue simplificada, es posible conectar rápidamente un repositorio de código y configurar el proceso de implementación con solo

unos pocos clics. Esto te permite tener tu aplicación lista para ser desplegada de forma rápida y sin complicaciones.

Otra característica interesante es la integración continua y el despliegue continuo (CI/CD). Al conectar tu repositorio de código, puedes automatizar el proceso de construcción y despliegue de tu aplicación. Cada vez que realices cambios en tu repositorio, Netlify se encargará de construir y desplegar automáticamente la última versión de tu aplicación. Esto agiliza el ciclo de desarrollo y garantiza que tus usuarios siempre tengan acceso a las últimas actualizaciones.

Netlify también ofrece soporte para Vite y React.js, lo que lo hace compatible con las tecnologías que estás utilizando en este proyecto. Esto facilita la integración y el desarrollo de tu aplicación sin problemas.

En términos de costos, Netlify cuenta con un plan gratuito que te permite comenzar sin incurrir en gastos adicionales. Esto es beneficioso para proyectos de pequeña escala o para aquellos que están en etapas iniciales de desarrollo.

Además, Netlify proporciona un despliegue global y escalabilidad automática. Su red global de servidores distribuidos en diferentes ubicaciones geográficas garantiza que tu aplicación esté disponible para usuarios de todo el mundo, con tiempos de carga rápidos. Además, la capacidad de escalabilidad automática de Netlify te permite gestionar incrementos en el tráfico sin preocuparte por problemas de rendimiento.

## **2.2 Backend**

Consideramos como alternativa para el desarrollo de nuestro backend el uso de Python debido a su amplia popularidad en el ámbito de desarrollo. Sin embargo, tras analizar los requisitos y objetivos de nuestro proyecto, tomamos la decisión de utilizar Node.js como tecnología principal.

Una de las razones clave para elegir Node.js es su rendimiento y escalabilidad. Node.js está construido sobre el motor V8 de Google Chrome, lo que lo convierte en una opción altamente eficiente en términos de velocidad de ejecución. Esto es especialmente importante para nuestra aplicación, ya que nos permite manejar un gran número de solicitudes y procesar datos de manera rápida y eficiente.

Otro factor que influyó en nuestra elección fue la sinergia con React. Al utilizar Node.js, podemos aprovechar el hecho de que ambos utilizan JavaScript como lenguaje de programación. Esto nos brinda la ventaja de compartir código y recursos entre el backend y el frontend.

Para ayudarnos al desarrollo hemos utilizado Expressjs, un marco de aplicación web para Node.js que simplifica la creación de aplicaciones web y APIs. El enfoque minimalista de esta herramienta brinda la capacidad de crear aplicaciones web de manera rápida y sencilla. Su diseño centrado en la

simplicidad permite un desarrollo ágil y eficiente, sin cargar con funcionalidades innecesarias.

La modularidad es otra ventaja clave de esta herramienta. Permite elegir los módulos específicos que necesitan para su aplicación, evitando la carga de funcionalidades innecesarias. Esto proporciona una mayor flexibilidad y optimización en el uso de recursos.

Además, esta herramienta se integra fácilmente con otras bibliotecas y marcos de trabajo de Node.js. Esta capacidad de integración facilita la construcción de proyectos grandes y complejos, permitiendo aprovechar las ventajas y funcionalidades adicionales que ofrecen otras herramientas y bibliotecas.

La incorporación de middleware es otra característica destacada. Permite agregar funcionalidades personalizadas a la aplicación de manera fácil y eficiente. Esto ofrece una mayor flexibilidad para adaptar la aplicación a requisitos específicos y agregar características personalizadas sin complicaciones.

Para el despliegue del backend utilizamos render por su facilidad de uso. Ofrece una experiencia de despliegue sencilla y fácil de configurar. Al conectar tu repositorio de código, puedes configurar el despliegue de tu aplicación backend en unos pocos pasos. Render se encarga de la configuración del servidor y del escalado automático, lo que te permite centrarte en el desarrollo de tu aplicación sin preocuparte por la infraestructura.

La integración continua y el despliegue continuo (CI/CD) son características clave de Render. Puedes conectar tu repositorio de código y configurar la integración continua para que tu backend se construya y despliegue automáticamente en Render cada vez que realices cambios en tu repositorio. Esto agiliza el ciclo de desarrollo y garantiza que siempre estés trabajando con la última versión de tu aplicación.

Render también ofrece soporte para Node.js y Express.js, lo que lo hace compatible con las tecnologías que se están utilizando en tu proyecto. Además de la compatibilidad, Render proporciona características adicionales como balanceo de carga, enrutamiento personalizado y monitoreo de rendimiento. Estas características te permiten gestionar y optimizar tu aplicación de manera eficiente, asegurando un rendimiento óptimo.

En términos de costos, Render ofrece un plan gratuito, lo que permite comenzar sin incurrir en gastos adicionales. Esto es clave para este proyecto que está en la etapa inicial.

La escalabilidad automática es otra ventaja importante de Render. Proporciona escalabilidad automática para el backend, lo que significa que la aplicación se ajustará automáticamente según la demanda de tráfico. A medida que el tráfico aumenta, Render aumentará automáticamente el número de instancias de tu aplicación para mantener un rendimiento óptimo. Esto

permite manejar picos de tráfico sin preocupaciones y garantizar una experiencia fluida para los usuarios.

## 2.3 Base de datos

Para la base de datos hemos utilizado mongoose junto con MongoDB Atlas. MongoDB Atlas nos brinda una plataforma en la nube confiable y fácil de usar para alojar la base de datos MongoDB.

MongoDB destaca por su capacidad de escalabilidad. Permite escalar la base de datos de manera fácil y rápida para manejar grandes volúmenes de datos y tráfico web. Esto es especialmente importante en entornos en crecimiento donde la demanda de almacenamiento y rendimiento puede aumentar rápidamente.

La disponibilidad es otra fortaleza de MongoDB Atlas. Garantiza una alta disponibilidad y tolerancia a fallos a través de su arquitectura distribuida y múltiples réplicas. Esto asegura que los datos estén siempre accesibles y protegidos, incluso en caso de fallos o interrupciones.

El modelo de datos de MongoDB se basa en documentos, que se asemejan a la estructura de los objetos en JavaScript. Esto facilita la interacción con la base de datos utilizando el mismo lenguaje que se utiliza en toda la aplicación. Esta familiaridad y consistencia en el lenguaje de programación simplifican el desarrollo y agilizan la manipulación de los datos.

La flexibilidad es otra ventaja clave de MongoDB Atlas. Su esquema flexible permite almacenar datos sin una estructura predefinida. Esto es especialmente útil cuando no se conoce de antemano la estructura de los archivos de datos. Permite almacenar información heterogénea y cambiar el esquema de manera dinámica sin afectar la integridad de los datos. Esto brinda libertad para modelar los datos de forma ágil y eficiente, sin limitaciones impuestas por una estructura rígida.

En la figura 2.4 podemos observar las descargas de las herramientas utilizadas en nuestro proyecto con otras soluciones.

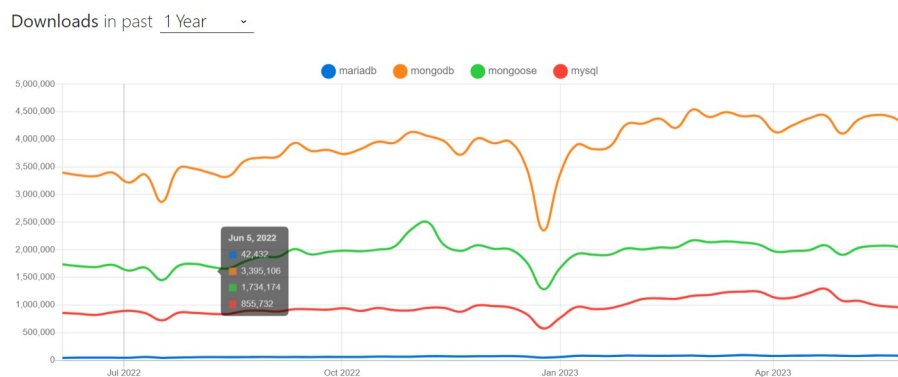


Figura 2.4: Descargas de mongoose y mongodb frente a otras bases de datos



# Capítulo 3

## Control de sesión

Para restringir el acceso a la funcionalidad de publicar iniciativas en el mapa únicamente a los usuarios registrados, se ha implementado un control de inicio de sesión en la aplicación. De esta manera, se busca asegurar que solo aquellos usuarios que poseen una cuenta y han iniciado sesión puedan realizar esta acción.

Además, el control de inicio de sesión también contempla un enfoque específico para el administrador de la aplicación. El administrador cuenta con funciones y privilegios adicionales que le permiten realizar tareas propias de su rol, como gestionar y moderar las iniciativas publicadas, realizar ajustes en la configuración de la aplicación, y llevar a cabo otras acciones administrativas relevantes.

Al requerir el inicio de sesión para publicar iniciativas en el mapa, se asegura que las contribuciones realizadas sean atribuibles a usuarios autenticados, lo cual facilita la trazabilidad y responsabilidad en el contexto de la aplicación.

### 3.1 Tecnología utilizada

Para el control de usuario utilizamos JSON Web Token (JWT), un estándar abierto basado en JSON que se utiliza para transmitir información entre partes como un objeto de forma segura. JWT (JSON Web Tokens) ofrece ventajas significativas en términos de seguridad y autenticación. Este formato de token utiliza algoritmos de cifrado para garantizar la seguridad de la información transmitida. Además, los tokens JWT son utilizados para autenticar a los usuarios en aplicaciones web y móviles, proporcionando una capa adicional de seguridad.

Una de las ventajas de JWT es su ligereza, ya que su formato de datos compacto facilita su transmisión a través de la red. Asimismo, los tokens JWT pueden configurarse con un tiempo de caducidad, lo que incrementa la seguridad al limitar su uso a un período determinado.

Otra característica destacada de JWT es la integridad de los datos. Los tokens están firmados digitalmente, lo que garantiza que la información transmitida no haya sido modificada durante su transporte.



En diferentes secciones de la aplicación, se requiere que el usuario inicie sesión para acceder a determinadas funcionalidades. Es necesario que el usuario tenga una sesión iniciada para poder agregar nuevas iniciativas y acceder a la sección de administrador, que permite editar iniciativas y cargar datos para su visualización en la gráfica.

Es importante destacar que el inicio de sesión garantiza la seguridad y la personalización de la experiencia del usuario. Al requerir que el usuario esté autenticado, se evita el acceso no autorizado y se asegura que solo las personas adecuadas tengan la capacidad de agregar iniciativas y realizar cambios en el sistema. Además, el inicio de sesión también permite almacenar y asociar la información correctamente, lo que facilita la gestión y el seguimiento de las iniciativas y los datos cargados en la aplicación.

Para la utilización del token primero lo obtenemos del inicio de sesión en la aplicación. Junto con el inicio de sesión, también se almacena el rol del usuario en el *sessionStorage* para preservarlo en caso de que la página se recargue.

Es importante destacar que el token de sesión es un elemento fundamental para la autenticación y autorización del usuario en la aplicación. Una vez que el usuario ha iniciado sesión exitosamente, se genera un token único que se utiliza para identificar y verificar su identidad en las solicitudes posteriores.

Al almacenar el rol del usuario en el *sessionStorage*, se asegura que esta información esté disponible incluso si la página se refresca o se recarga. Esto garantiza una experiencia de usuario continua y evita que el usuario tenga que volver a iniciar sesión cada vez que se produce una actualización de la página.

Una vez que el usuario ha iniciado sesión y se ha obtenido su rol, se puede utilizar esta información para mostrar determinadas secciones de la página exclusivamente al administrador. Además, se realiza una comprobación de validez del token al recibir solicitudes en rutas restringidas que solo el administrador puede acceder, como por ejemplo la ruta para agregar nuevos datos destinados a la representación gráfica.

La utilización del token de sesión permite implementar un nivel de seguridad adicional en la aplicación. Al asignar roles a los usuarios, se puede controlar de manera precisa qué funcionalidades y secciones de la página están disponibles para cada tipo de usuario. Esto asegura que solo el administrador tenga acceso a las secciones y acciones específicas reservadas para su rol.

Asimismo, al realizar la comprobación de validez del token en rutas restringidas, nos aseguramos de que solo los usuarios con los permisos adecuados puedan acceder a esas funcionalidades. Esto evita cualquier intento de acceso no autorizado y garantiza la integridad y seguridad de los datos.

# Capítulo 4

## Creación de gráfica

Se ha implementado un apartado en la aplicación que utiliza la biblioteca Recharts para visualizar gráficas dinámicas basadas en los datos almacenados en la base de datos. Estos datos provienen de archivos CSV subidos por el administrador a nuestra plataforma.

La funcionalidad de gráficos personalizables permite a los usuarios seleccionar entre diferentes tipos de gráficos, como gráficos, barras o circulares, para representar los datos de manera visualmente atractiva y comprensible. Además, los usuarios tienen la opción de elegir las variables específicas que desean mostrar en la gráfica, lo que brinda flexibilidad y adaptabilidad para explorar diferentes aspectos de los datos almacenados.

Este enfoque proporciona una experiencia interactiva y personalizada para los usuarios al permitirles explorar y analizar los datos relevantes para sus necesidades. Al ofrecer la posibilidad de visualizar y comparar diferentes variables y tipos de gráficos, se fomenta la comprensión y la identificación de patrones y tendencias en los datos subyacentes.

### 4.1 Publicación de datos

Una vez que el administrador inicia sesión en su cuenta, podrá acceder al apartado de administrador que se encuentra en la barra de navegación, como podemos observar en la figura 4.1.



Figura 4.1: Barra de navegación cuando el administrador inicia sesión

En el panel de administrador, tendrás la opción de realizar diversas acciones, como publicar nuevos datos, editar la ponderación de los datos, eliminar datos y editar los datos ya publicados.

Para agregar un nuevo archivo CSV con los datos, se deberá proporcionar un

nombre que será visible para los usuarios y luego importar el archivo CSV desde tu ordenador. También es obligatorio añadir una pequeña descripción del conjunto de datos que se están añadiendo para mejorar la experiencia de usuario cuando consulte estos.

En la figura 4.2 se puede ver el panel de administrador comentado anteriormente.

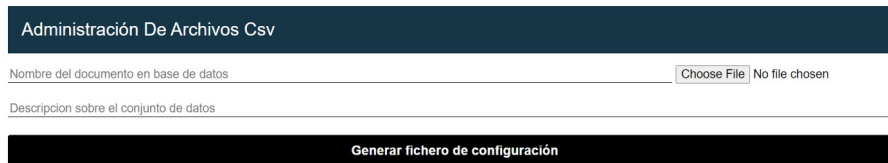


Figura 4.2: Imagen del administrador de csv

Una vez que se haga clic en "Generar fichero de configuración", se podrá seleccionar los campos que desea mostrar en cada eje para la representación gráfica, como se puede observar en la figura 4.3.

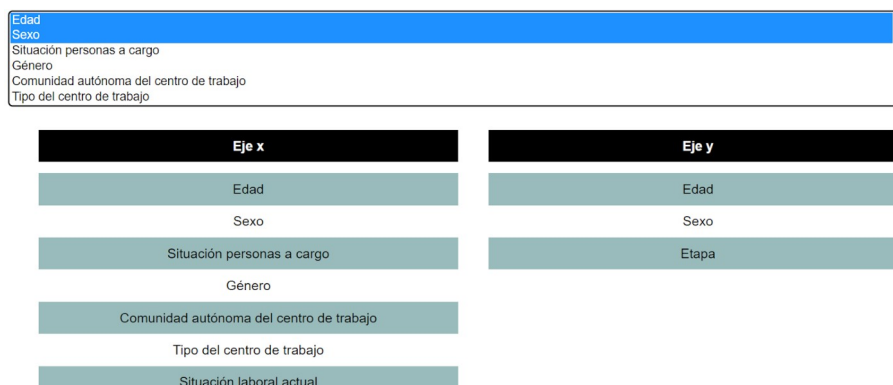


Figura 4.3: Imagen del selector de campos

Después de elegir los ejes, se generarán automáticamente todas las combinaciones únicas posibles entre esos campos para asignar la gráfica predeterminada que desees utilizar.

La siguiente figura es un ejemplo del selector de gráfica por defecto a cada par de ejes.

| Campo 1 | Campo 2                                  | Gráfica Recomendada | Descripción |
|---------|--|---------------------|-------------|
| Sexo    | Situación personas a cargo               | Gráfica circular    | opcional    |
| Sexo    | Edad                                     | Gráfico de barras   | opcional    |
| Sexo    | Género                                   | Gráfico de barras   | opcional    |
| Sexo    | Comunidad autónoma del centro de trabajo | Gráfico de barras   | opcional    |
| Sexo    | Tipo del centro de trabajo               | Gráfico de barras   | opcional    |
| Sexo    | Situación laboral actual                 | Gráfico de barras   | opcional    |

Figura 4.4: Imagen del selector de gráficas por defecto por el administrador

Por defecto, todas las gráficas se mostrarán como gráficas de barras para facilitar la tarea de asignación al administrador. También se le ofrece al administrador la capacidad de cambiar la descripción de la grafica en caso de que no quiera la que se le asigna por defecto, que sería los ejes seleccionados como por ejemplo “Edad - Sitiacion de personas a cargo”.

Después de enviar la configuración, se podrá ver el nuevo archivo disponible para su visualización al recargar la página en unos segundos.

Para asignar una ponderación a los datos, se necesita acceder a la ruta de asignación de ponderación en el panel de administrador. La ponderación se asignará y se mostrará en la tabla que podemos ver en la figura 4.5.

| Ponderacion De Datos |        |       |  |
|----------------------|--------|-------|--|
| Etapa                | Hombre | Mujer |  |
| Etapa A              | 1.0001 | 1     |  |
| Etapa B              | 1      | 1     |  |
| Etapa C              | 1      | 1     |  |
| Etapa D              | 1      | 1     |  |

**Guardar cambios**

Figura 4.5: Imagen del selector de ponderación

Por defecto estarán todos los valores a cero hasta que el administrador les asigne uno.

## 4.2 Obtención de datos

### 4.2.1 Obtención de datos desde base de datos

Al cargar el componente, se realiza una solicitud a la base de datos utilizando el nombre del archivo que se quiere buscar. Este nombre se extrae de la ruta de acceso (pathname) cuando el usuario hace clic en el archivo que desea utilizar para mostrar los datos.

También se hace una solicitud para obtener la ponderación que se utilizará en los datos. Esta llamada se realiza cada vez que la ruta de acceso (pathname) cambia en la ruta "/" y su valor coincide con una de las rutas

dinámicas, excluyendo las rutas estáticas como "/Iniciativas".

Las rutas dinámicas se generan mediante una solicitud a la base de datos para obtener todos los nombres de archivos disponibles, y luego se crean las rutas a partir de "/".

Una vez obtenidos los datos, se realiza otra solicitud a la base de datos para obtener el archivo de configuración que contiene información sobre qué datos estarán disponibles en cada eje. En el archivo de configuración también se encuentra la descripción del conjunto de datos que se está mostrando y la descripción de cada eje. Después de obtener esta información, se configuran todos los campos de cada eje en "true" para que se muestren en la gráfica la primera vez, como podemos ver en el desplegable que aparece en la parte izquierda de la imagen de la figura 4.6.

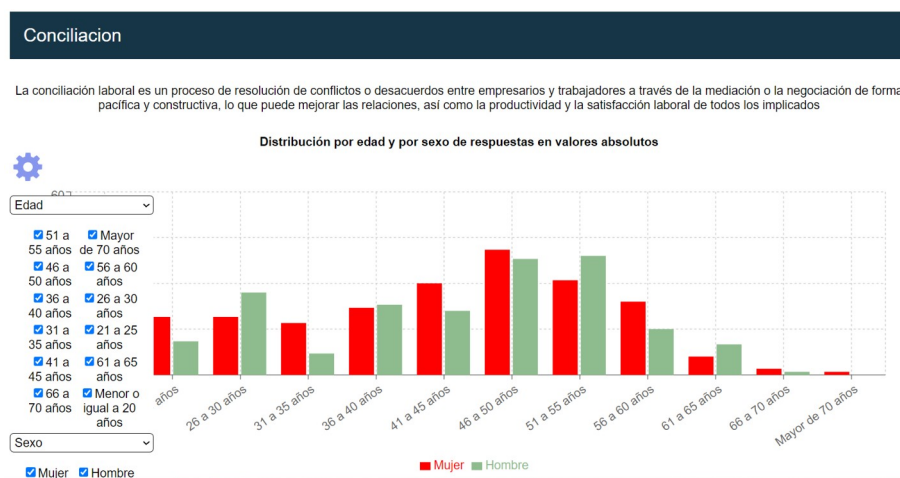


Figura 4.6: Imagen del gráfico con los datos a seleccionar

#### 4.2.2 Procesamiento de datos para representarlos en una gráfica

Todos los datos los tenemos que pasar de esta estructura que nos llega desde la base de datos que es la que podemos apreciar en la figura 4.7.

```
[{ Edad: '51 a 55 años', // Cada objeto es una fila del csv
  Sexo: 'Mujer',
  'Género': 'Femenino',
  ...Más campos
}...Más objetos]
```

Figura 4.7: Ejemplo de estructura antes de hacer las transformaciones

a esta la que podemos ver en la figura 4.8 con la información que queremos. En este caso "Edad" es el eje X y "Sexo" es el eje Y.

```
[{ Edad: "51 a 55 años", Sexo: "Mujer" },  
{ Edad: "51 a 55 años", Sexo: "Hombre" }, ...Mas objetos]
```

Figura 4.8: Ejemplo de estructura después de las transformaciones

Este paso es crucial, ya que para utilizar la biblioteca Recharts es necesario proporcionar los datos en un formato específico.

Una vez obtenido el vector de objetos, se puede seleccionar qué variables se desea mostrar en cada eje, utilizando la clave correspondiente en el objeto, como "Edad" o "Sexo". Además, cada objeto también contiene la altura de esa variable, lo cual es necesario para su visualización. Para calcular esta altura, se cuenta las coincidencias multiplicadas por la ponderación asignada por el administrador y eliminamos aquellos objetos con altura 0 para que no se muestren en la gráfica.

Además, se realiza algunas transformaciones en los datos. Por un lado, se elimina los valores nulos y, por otro lado, se ordenan para que los campos como la edad tengan un orden coherente en la representación gráfica. Para lograr esta ordenación, dado que los datos son cadenas de caracteres, se sigue la siguiente lógica: las palabras que comienzan con "menor" se colocan al principio, las que comienzan con "mayor" al final y, para el resto, extraemos la cadena hasta el primer espacio en blanco y la convertimos a un número. De esta manera, se puede ordenar los datos utilizando el método "sort" de los arrays en JavaScript.

En caso de que se encuentren cadenas que no se pueden convertir en números (NaN, por sus siglas en inglés, "not a number"), se dejan al principio de la ordenación, considerándolas siempre como el valor más bajo en la comparación, para que los demás números puedan ser comparados correctamente. En la figura 4.9 podemos ver un ejemplo de la ordenación comentada.



Figura 4.9: Imagen de gráfico con los campos ordenados de menor a mayor

### 4.3 Renderizado de gráfica

Cada par de ejes seleccionados tiene una gráfica recomendada por el administrador que se mostrará. Esta información está en el fichero de configuración que se obtiene únicamente cuando se produce un cambio en el pathname.

La configuración se consulta cada vez que se modifican los ejes seleccionados. Una vez obtenido el tipo de gráfica elegido, evaluamos una condición para determinar cuál gráfico debemos renderizar. En la imagen adjunta, se puede observar que utilizamos solo tres componentes para representar los cinco tipos de gráficas. Por ejemplo, para el gráfico circular, se utiliza un componente especial que solo requiere un dato, como el sexo.

En el caso de las gráficas de barras horizontales y apiladas, se emplea otro componente que, al recibir un valor para el atributo "stack", se convierte en una gráfica apilada. Esta misma estrategia se aplica para el gráfico de barras en general. Esto se puede observar en la figura 4.10.

```

{dataset.length === 0 && <>Selecciona como mínimo un campo en cada eje</>}
{chartSelected === 'Gráfico de barras' && <Chart dataset={dataset} keyFields={selectedFields[1].yfield} />}
{chartSelected === 'Gráfico de barras apiladas' && <Chart dataset={dataset} stackFlag='a' keyFields={selectedFields[1].yfield} />}
{chartSelected === 'Gráfica circular' && <CircularChart dataset={dataset} keyFields={selectedFields[1].yfield} />}
{chartSelected === 'Gráfico de barras horizontal' && <HorizontalBarChart dataset={dataset} keyFields={selectedFields[1].yfield} />}
{chartSelected === 'Gráfico de barras horizontal apiladas' && <HorizontalBarChart stackFlag='a' dataset={dataset} keyFields={selectedFields[1].yfield} />}

```

Figura 4.10: Imagen del código que controla qué gráfica se muestra

Para el renderizado de la gráfica, fue necesario envolver el componente en un <div> adicional como se aprecia en la figura 4.11. Esto se debe a que se desea lograr proporciones diferentes según el tamaño de pantalla, ya que se visualizan mejor cuando se ajusta el aspect ratio. Simplemente utilizar el contenedor responsive no es suficiente, ya que, aunque sea responsive por sí mismo según su nombre, no nos permite cambiar el aspect ratio de forma independiente.

```
<div className='chart-container'>
  <ResponsiveContainer
    width='100%' height='100%'
  >
    <BarChart
      margin={responsiveMargin}
      data={dataset}
    >
      <CartesianGrid strokeDasharray='3 3' />
      <XAxis dataKey={dataset[0].xField} tick={CustomTick} interval={0} />
      <YAxis />
      <Tooltip />
      <Legend wrapperStyle={{ top: '90%', right: 0, textAlign: 'right' }} />
      {keys.map((key, i) : Element => <Bar key={key} stackId={stackFlag} dataKey={key} fill={COLORS[i]} />)}
    </BarChart>
  </ResponsiveContainer>
</div>
```

Figura 4.11: Imagen que muestra cómo se hizo responsive las gráficas

Para abordar el problema de las etiquetas de los ejes que son demasiado grandes para mostrarlas completamente en pantalla, se crea un componente personalizado. Este componente renderiza las etiquetas con un ángulo de 35 grados si la pantalla tiene un ancho mayor a 600 píxeles, y un ángulo de 60 grados si es más pequeña.

Es importante destacar que esta solución se aplica específicamente al gráfico de barras normal. En el caso del gráfico de barras horizontal, no necesita rotar las etiquetas. En su lugar, se utiliza un enfoque diferente basado en el margen disponible dentro del gráfico para mostrar las etiquetas. La cantidad de margen utilizada varía dependiendo del tamaño de la pantalla.

Además, se ha implementado una funcionalidad para cortar las etiquetas cuando superan un determinado tamaño. Aunque se ha hecho todo lo posible para mostrar las etiquetas completas, en algunos casos no es factible debido a restricciones de espacio. En estos casos, optamos por truncar las etiquetas y mostrar solo una parte de ellas. Además, en pantallas más pequeñas, se reduce el tamaño de la fuente para proporcionar más espacio para la renderización de las etiquetas. Si el usuario desea ver el nombre completo, puede hacerlo pasando el cursor sobre la barra correspondiente en el gráfico.



# Capítulo 5

## Backend

El backend desarrollado en Express.js es una parte fundamental de esta aplicación. Es el encargado de gestionar las solicitudes y respuestas del servidor, así como de manejar la lógica de negocio y la comunicación con la base de datos. Este backend proporciona un entorno robusto y seguro para respaldar las funcionalidades de nuestra aplicación.

Se integra sin problemas con la base de datos, permitiendo el almacenamiento y recuperación eficiente de datos mediante consultas y operaciones en la base de datos. Esto brinda la capacidad de interactuar con los datos necesarios para la funcionalidad de la aplicación, ya sea recuperando, actualizando registros o creando nueva información.

### 5.1 Rutas de la aplicación

En la aplicación, hemos establecido diferentes rutas para realizar diversas acciones relacionadas con los usuarios. Tomando como ejemplo las rutas de usuarios, llevamos a cabo las siguientes operaciones:

- Recuperar un usuario por su nombre de usuario y contraseña.
- Crear un nuevo usuario.
- Obtener el rol de un usuario solicitado.
- Ruta para borrar un usuario.

Para definir el propósito y la funcionalidad de cada una de estas rutas, se utiliza los métodos HTTP estándar, tales como GET, PUT, POST, PATCH y DELETE. Estos métodos se emplean para establecer la forma en que se interactúa con los recursos en el servidor. Cada uno de estos métodos tiene una semántica y un propósito específico que determina la operación a realizar en el servidor.

- GET: se utiliza para solicitar y obtener datos de un recurso en el servidor. Por ejemplo, obtener los datos de la gráfica explicada anteriormente.

- PUT: se utiliza para actualizar completamente un recurso en el servidor. Es decir, el recurso se reemplaza por completo con los nuevos datos proporcionados. Por ejemplo al actualizar la configuración de gráficas.
- POST: se utiliza para enviar datos al servidor para crear un nuevo recurso. Por ejemplo, crear una nueva cuenta de usuario.
- PATCH: se utiliza para realizar modificaciones parciales en un recurso existente. A diferencia de PUT, PATCH no reemplaza completamente el recurso, sino que actualiza solo las partes específicas que se proporcionan en los datos de la solicitud. Lo utilizamos a la hora de actualizar las iniciativas.
- DELETE: se utiliza para eliminar un recurso del servidor. Por ejemplo, para eliminar un usuario existente de una base de datos.

En las rutas de usuario, se cuenta con una ruta protegida, que es la de "obtener rol". Esta ruta utiliza dos middlewares para llevar a cabo su funcionamiento. El primer middleware se encarga de extraer el usuario del token que se pasa al backend. Esta medida de seguridad es necesaria para verificar la autenticidad del usuario y garantizar que solo los usuarios registrados y autenticados puedan acceder a esta ruta.

El segundo middleware se utiliza para verificar si el usuario es un administrador. Al implementar esta verificación, se asegura que solo los usuarios con privilegios de administrador tengan acceso a la funcionalidad de obtener el rol. Esto es especialmente relevante en un sistema con múltiples roles de usuario, donde es necesario limitar el acceso a ciertas acciones y recursos a usuarios específicos.

La inclusión de estos middlewares en la ruta "obtener rol" proporciona una capa adicional de seguridad y control de acceso. Al verificar tanto la autenticidad del usuario como su rol, se garantiza que solo los administradores puedan realizar la acción de obtener el rol. Esto contribuye a la integridad y seguridad del sistema, evitando accesos no autorizados y asegurando que las funcionalidades sensibles estén protegidas adecuadamente.

### 5.1.1 Rutas de usuario

En la siguiente tabla se muestran todas las rutas que usa en esta aplicación para funcionar respecto a los usuarios.

| <b>Auth?</b> | <b>Exclusiva de Administrador</b> | <b>Método</b> | <b>Ruta</b>            | <b>Acción</b>  |
|--------------|-----------------------------------|---------------|------------------------|--|
| No           | No                                | GET           | /user/:email/:password | Buscar si hay un usuario que coincida con el nombre y contraseña |

|    |    |        |       |   |
|----|----|--------|-------|---|
|    |    |        |       | proporcionados                            |
| No | No | POST   | /user | Crear el usuario con la contraseña pasada |
| Si | No | DELETE | /user | Eliminar el usuario de la base de datos   |

**Tabla 5.1: Rutas de usuario**

### 5.1.2 Rutas de iniciativas

En la siguiente tabla se muestran todas las rutas que usa en esta aplicación para funcionar respecto a las iniciativas.

| <b>Auth?</b> | <b>Exclusiva de Administrador</b> | <b>Método</b> | <b>Ruta</b>  | <b>Acción</b>   |
|--------------|-----------------------------------|---------------|--|---|
| No           | No                                | GET           | /initiative  | Devuelve un array con todas las iniciativas guardadas en la base de datos                   |
| No           | No                                | GET           | /initiative/:ComunidadAutonoma/:active?/:validated | Devuelve un array con las iniciativas que cumplan los filtros que le pasemos en la petición |
| Si           | No                                | POST          | /initiative  | Crear una iniciativa en la base de datos con la información que ha rellenado el usuario     |
| Si           | Si                                | PATCH         | /initiative  | Actualizar los campos que se le pasen a la iniciativa seleccionada                          |
| Si           | Si                                | DELETE        | /initiative  | Elimina la iniciativa seleccionada  |

**Tabla 5.2: Rutas de iniciativas**

### 5.1.3 Rutas de datos

En la siguiente tabla se muestran todas las rutas que usa en esta aplicación para funcionar respecto a los datos que sube el administrador para la visualización en la gráfica interactiva.

| <b>Auth?</b> | <b>Exclusiva de Administrador</b> | <b>Método</b> | <b>Ruta</b>                      | <b>Acción</b>   |
|--------------|-----------------------------------|---------------|----------------------------------|---|
| No           | No                                | GET           | /data/weighing                   | Trae los valores de ponderación configurados por el administrador   |
| No           | No                                | GET           | /data/fileNames                  | Devuelve un array con los nombres de todos los ficheros disponibles para visualizar                         |
| No           | No                                | GET           | /data/configFiles                | Devuelve Todos los ficheros de configuración  |
| No           | No                                | GET           | /data/configField/:name/:idFlag? | Devuelve el fichero de configuración que tiene el nombre pasado o la id                                     |
| No           | No                                | GET           | /data/axes/:name                 | Devuelve los campos que tendrá cada eje según el nombre de fichero que le pasemos                           |
| No           | No                                | GET           | data/dataFile/:name              | Devuelve los datos del fichero que hemos seleccionado para su representación                                |
| No           | No                                | GET           | data/description/:id             | Devuelve la descripción del conjunto de datos correspondiente a la id                                       |
| Si           | Si                                | POST          | data/:collectionName             | Creamos un nuevo documento con los datos del csv y su fichero de configuración con el nombre que le pasamos |
| Si           | Si                                | POST          | data/weighing/data               | Crear el documento donde se guarda los  |

|    |    |        |  |  |
|----|----|--------|--|--|
|    |    |        |  | datos de ponderación   |
| Si | Si | PUT    | data/collectionName/:name/:id/:newName | Actualiza el nombre y descripción del fichero que coincide con la id                 |
| Si | Si | PUT    | data/configFiles/:id                   | Actualiza el fichero de configuración con los nuevos datos que pasa el administrador |
| Si | Si | PUT    | data/weighing                          | Actualiza la ponderación   |
| Si | Si | DELETE | data/dataFile/:name/:id                | Elimina el documento seleccionado por el administrador                               |

**Tabla 5.3: Rutas de datos**

# Capítulo 6

## Base De datos

La base de datos implementada en MongoDB Atlas desempeña un papel fundamental, al proporcionar un sistema de almacenamiento eficiente y escalable para gestionar la información clave de las cuentas de usuarios, las iniciativas y los datos utilizados para generar las gráficas.

Aprovechando la flexibilidad y el rendimiento que ofrece MongoDB, se ha diseñado una estructura de datos adecuada para nuestro modelo de negocio, permitiéndonos almacenar y organizar la información de manera óptima.

Con la base de datos en MongoDB Atlas, se puede almacenar de manera segura y eficiente los detalles de las cuentas de usuarios, incluyendo información como nombres, correos electrónicos y contraseñas encriptadas. Esto permite gestionar el registro, inicio de sesión y autenticación de los usuarios de forma segura y confiable.

Además, se utiliza la base de datos para almacenar y administrar los datos relacionados con las iniciativas. Esto incluye detalles como título, descripción, ubicación y correo de contacto. Gracias a la flexibilidad de MongoDB, se puede adaptar el esquema de la base de datos según nuestras necesidades, lo que nos permite agregar nuevos campos o modificar la estructura de los datos sin interrupciones en la funcionalidad de la aplicación.

Asimismo, MongoDB Atlas proporciona la capacidad de almacenar y consultar los datos necesarios para generar las gráficas. Esto incluye variables, valores y otros detalles relevantes para la visualización de los datos. Se puede realizar consultas eficientes y obtener los datos requeridos para crear gráficas personalizables y actualizadas en función de los datos almacenados.

### 6.1 Creación de modelos de datos

Mongoose se utiliza como una biblioteca de modelado de objetos de MongoDB para Node.js. Esta proporciona una capa de abstracción sobre la base de datos NoSQL MongoDB, para simplificar la interacción con esta y proporcionar estructura y validación a los datos almacenados. Los esquemas

en Mongoose permiten definir la forma en que los datos deben ser organizados y qué tipos de datos se esperan en cada campo. Esto ayuda a mantener la coherencia en la estructura de los documentos almacenados y proporciona una capa de validación para garantizar que los datos cumplan con los requisitos especificados.

Al definir un esquema en Mongoose, se pueden establecer propiedades como el tipo de dato esperado (por ejemplo, String, Number, Date, etc.), requisitos de longitud, valores por defecto, índices y restricciones de unicidad, entre otros. También se pueden definir métodos personalizados en el esquema para realizar operaciones específicas en los documentos.

Una vez definido el esquema, se puede crear un modelo basado en ese esquema, que actúa como una interfaz para interactuar con la base de datos. Los modelos de Mongoose permiten realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en la base de datos MongoDB de manera más fácil y consistente, abstrayendo las complejidades de las consultas y manipulación de datos.

MongoDB es una base de datos NoSQL que almacena los datos en formato de documentos JSON sin esquema fijo. Sin embargo, en nuestro caso es beneficioso tener una estructura y validación de datos por lo que se utilizará los esquemas de mongoose.

### **6.1.1 Esquema de usuario**

Para la creación de usuario se declara el siguiente esquema que vemos en la figura 6.1. Con el que se consigue que por defecto todos los usuarios creados son clientes, que la contraseña, el email y el rol tienen que estar para que se pueda ingresar en la base de datos además de una validación para comprobar que es un email lo que se va a insertar.

```

const userSchema = new mongoose.Schema({
  email: {
    type: String,
    required: true,
    unique: true,
    validate: {
      validator: (email: string) => validator.isEmail(email),
      message: 'Invalid email address'
    }
  },
  password: {
    type: String,
    required: true
  },
  role: {
    type: String,
    required: true,
    default: 'client'
  }
})

```

Figura 6.1: Ejemplo de un schema en mongoose

### 6.1.2 Esquema de Iniciativa

Este esquema se encarga de comprobar que las iniciativas que se inserten en la base de datos, tengan como mínimo la localización, coordenadas, un nombre, si están validadas y si están activas. Por defecto `validated` y `active` están a `false` en caso de que en la iniciativa no se indiquen estos campos.

### 6.1.3 Esquema de fichero de configuración

Este esquema se asegura que tiene el nombre del fichero con el cual está relacionado esta configuración, la descripción del conjunto de datos, la configuración en sí que llevaría los dos campos y la gráfica asociada: ejemplo: [edad, sexo, grafico de barras] y por último los campos de de cada eje. Para el par de ejes tiene un campo opcional que corresponde con la descripción para ese par.

### 6.1.4 Datos sin esquema

Como se mencionó anteriormente, MongoDB es una base de datos NoSQL que no requiere un esquema fijo, lo que nos brinda ventajas al guardar datos provenientes de archivos CSV con estructuras imprevisibles.

Cada archivo CSV se guarda en una nueva colección con un nombre asignado desde la parte del cliente. Este nombre se utiliza para establecer la



relación con el archivo de configuración correspondiente.

# Capítulo 7

## Mapa de iniciativas

Un mapa de iniciativas es una representación visual de las diferentes acciones y proyectos relacionados con un tema específico, en este caso, la igualdad de género y la discriminación de género. En este tipo de mapa, se utilizan marcadores geográficos para mostrar la ubicación de cada iniciativa en un contexto geoespacial.

A través del mapa, los usuarios pueden explorar y descubrir las iniciativas que se están llevando a cabo en distintas regiones, obteniendo información detallada sobre cada una de ellas. Esto incluye datos como el nombre de la iniciativa, enlace a la página de esta, y un correo electrónico de contacto.

El mapa de iniciativas no solo permite visualizar la distribución geográfica de las acciones, sino que también proporciona herramientas interactivas para filtrar y buscar iniciativas según diversos criterios, como el tipo de proyecto, la fecha de inicio, el estado de desarrollo, entre otros. Esto facilita la exploración y la búsqueda de información específica dentro del mapa.

### 7.1 Obtención de iniciativas

Los datos necesarios para la representación se obtienen de la base de datos, incluyendo la posición en latitud y longitud, así como información detallada sobre las iniciativas. Esta información se muestra cuando el usuario hace clic en uno de los marcadores en el mapa.

Los datos se obtienen utilizando un filtro, que por defecto muestra las iniciativas activas y validadas como se observa en la figura 7.1. Solo el administrador tiene la capacidad de utilizar y ver el filtro para mostrar las iniciativas no validadas.

Cada vez que el filtro se modifica, se realizan consultas a la base de datos para obtener los datos solicitados por el usuario y actualizar la visualización en consecuencia.

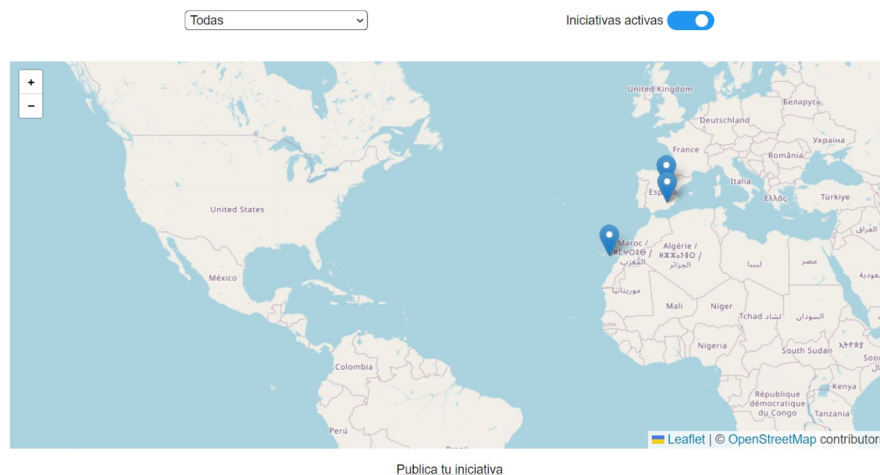


Figura 7.1: Imagen del mapa de iniciativas

## 7.2 Representación de iniciativas

Una vez obtenido el vector de iniciativas, se renderiza un marcador `<Marker>` por cada una en el que le indicamos el color del marcador la posición de este, y que tiene que hacer cuando se abre o cierra el pop up. En la figura 7.4 se puede ver un ejemplo visual del pop up mencionado.

Sin embargo, hay una condición en la primera línea, que se muestra en la figura 7.2, que controla si eres un usuario normal, en cuyo caso no se mostrarán las iniciativas al seleccionar la ubicación en el modo formulario, solo se te mostrará un marcador del sitio que estás seleccionando para tener una referencia visual. Esta decisión se tomó como una medida de precaución, considerando que se puede tener un gran número de iniciativas en nuestro sistema. Para evitar que sea complicado para el usuario seleccionar la ubicación en un mapa con muchos marcadores, se ha optado por ocultar las iniciativas al seleccionar la ubicación, especialmente para usuarios normales.

Esta práctica se basa en la experiencia de otras páginas que manejan marcadores geográficos similares a la nuestra, donde la cantidad de marcadores puede dificultar la interacción del usuario. Al mostrar únicamente los marcadores relevantes para la edición y administración de iniciativas, se busca mejorar la usabilidad y facilitar la selección de ubicaciones en el mapa.

Si eres administrador, se mostrarán las iniciativas en la parte del formulario para poder editarlas, eliminarlas o publicar una nueva.

Cuando el administrador abre el pop up, se cargarán los datos de la iniciativa seleccionada en los campos correspondientes del formulario. Además, ponemos el flag de actualización a verdadero para mostrar las opciones de actualizar las iniciativas al administrador. Cuando se cierra el pop up ponemos el flag de actualizar a falso para que oculte estas opciones hasta que abra el

siguiente pop up.

```
{(formFlag && user.role !== 'admin') ||
  initiatives.map(({
    latitude, _id, longitude, initiativeName, contacto,
    validated,
    link,
    active
  }): Element => {
    return (
      <Marker
        key={_id}
        position={[latitude, longitude]}
        icon={active ? blueIcon : redIcon}
        eventHandlers={{
          popupclose: () : void => { setUpdateFlag(false) },
          popupopen: () : void => {
            if (user.role === 'admin') {
              setUpdateFlag(true)
              setLocationData(prevState => ({
                ...prevState,
                latitude,
                id: _id,
                longitude,
                initiativeName,
                contacto,
                validated,
                link,
                active
              })))
            }
          }
        }}
      )
    )
  }
}
```

Figura 7.2: Lógica para diferenciar entre administrador y usuario común en el formulario de iniciativas

```
<Popup>
  <label>Nombre: {initiativeName}</label>
  <br />
  <label>Contacto: {contacto}</label>
  <br />
  Link:<a href={link}>{link}</a>
</Popup>
</Marker>
```

Figura 7.3: Imagen de como mostrar los datos en el popup

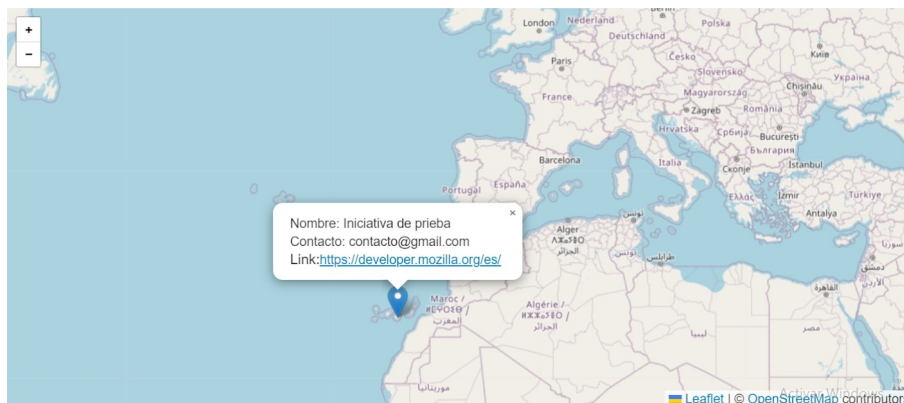


Figura 7.4: Imagen de un popup

## 7.3 Formulario

### 7.3.1 Usuario común

El formulario se mostrará una vez que se haga clic en "Publicar Iniciativa". En este formulario, se presentará un marcador en la ubicación actualmente seleccionada, lo cual proporciona una referencia visual para ayudar a elegir la ubicación de la iniciativa.

En el formulario, se solicitan los datos necesarios para agregar una iniciativa. Uno de los campos requeridos es el de ubicación, el cual se completa automáticamente con la localización seleccionada anteriormente como se demuestra en la figura 7.6.

Esto permite que el usuario tenga una experiencia más fluida al no tener que introducir manualmente la ubicación, ya que se captura automáticamente del marcador previamente seleccionado en el mapa. De esta manera, se simplifica el proceso de agregar una nueva iniciativa al ofrecer una forma intuitiva y conveniente de especificar la ubicación.



Figura 7.5: Imagen del formulario de iniciativas

### 7.3.2 Administrador

El formulario de publicación de iniciativa se muestra cuando se hace clic en "Publicar Iniciativa". La diferencia para el administrador es que puede hacer clic en un marcador existente en el mapa, lo cual activará un pop up con información sobre la iniciativa. Al hacer clic en el pop up, los datos de la iniciativa se cargarán en el formulario, lo que permite al administrador editar la iniciativa o eliminarla si lo desea como se observa en la figura 7.6.

Esta funcionalidad proporciona al administrador un medio conveniente para realizar acciones específicas en relación a una iniciativa existente. Al hacer clic en el marcador y cargar los datos en el formulario, se evita la necesidad de buscar y seleccionar manualmente la iniciativa en la base de datos. Esto agiliza el proceso de edición o eliminación de iniciativas, brindando al administrador un mayor control y eficiencia en la gestión del contenido.



Figura 7.6: Imagen del formulario de iniciativas para el administrador

# Capítulo 8

## Resultados

En este capítulo mostraremos el diseño final de nuestra aplicación web. Para llegar a estos resultados se realizó un cuestionario sobre la usabilidad de la web y puntos a mejorar. Basados en los resultados de la encuesta se realizaron las siguientes modificaciones que mejoraron la visualización y corrigieron algunos fallos presentes:

- Se implementó la solución de eliminar las respuestas de valor nulo 'NA' de la visualización.
- Se corrigió el problema donde el selector no se cerraba automáticamente al salir de él.
- Se ordenaron correctamente las variables en los ejes de las visualizaciones.
- Se resolvió el problema de funcionalidad del menú de navegación en pantallas pequeñas y se vinculó el logo del proyecto a la página de inicio.
- Se agregó una explicación al menú "Gráficas" para identificar claramente qué se está mostrando.
- Se tradujo el texto de "Registro" e "Inicio de sesión" al español para que sea uniforme con el resto de la interfaz.
- Se ajustó el comportamiento de la rueda de configuración de las gráficas para que el pop-up de configuración se oculte al hacer clic fuera de él, siguiendo la expectativa de interacción habitual.
- Se incorporó la capacidad de ponderación por parte del administrador.
- Se mejoraron las etiquetas en la medida de lo posible para una mayor claridad y comprensión.

Estas modificaciones se realizaron con el objetivo de optimizar la visualización de los datos y brindar una experiencia mejorada a los usuarios. A continuación se muestra el resultado final de las diferentes partes de la aplicación.

## 8.1 Usuario común

Las siguientes imágenes están relacionadas con las distintas gráficas disponibles que se han implementado.

La figura 8.1 representa los datos que tenemos por rango de edades ya que esta gráfica solo acepta un campo.

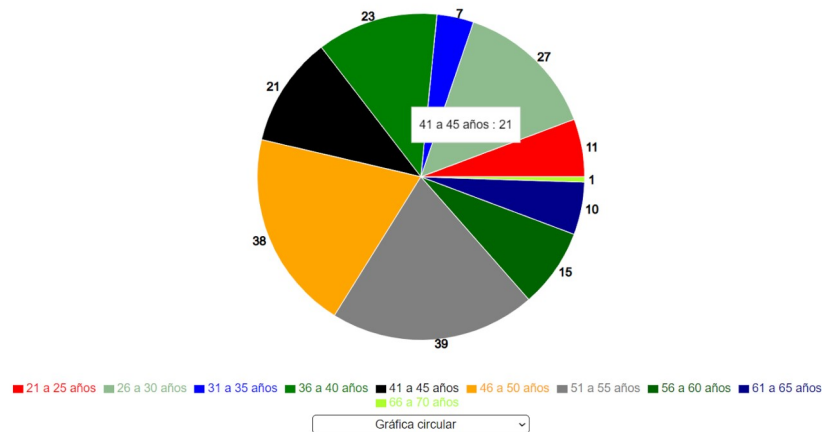


Figura 8.1: Gráfica circular

La figura 8.2 representa los datos que tenemos por rango de edades en el eje X y el sexo en el eje Y. En esta imagen están todos los campos disponibles seleccionados.



Figura 8.2: Gráfica de barras

En la figura 8.3 se muestra el mismo conjunto de datos que en la figura 8.2,



pero se ha aplicado la opción de apilar las barras. Además, se ha observado que los ejes seleccionados no tenían una descripción específica, por lo que se ha utilizado la asignada por defecto, que consiste en el nombre de cada eje separado por un guion.

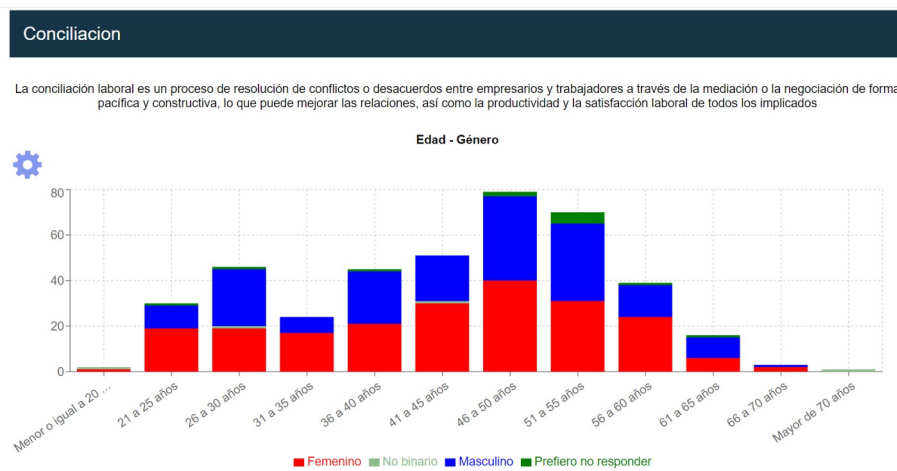


Figura 8.3: Gráfica de barras apiladas

La figura 8.4 representa los datos que tenemos en la 8.2 cambiando el tipo de representación a barras horizontales.

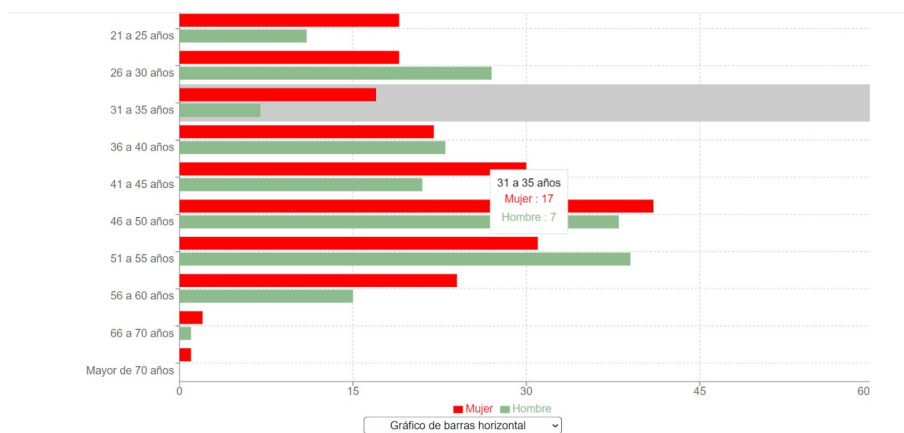


Figura 8.4: Gráfica de barras horizontales

La figura 8.5 representa los datos que tenemos en la 8.4 aplicando la opción de barras horizontales apiladas.

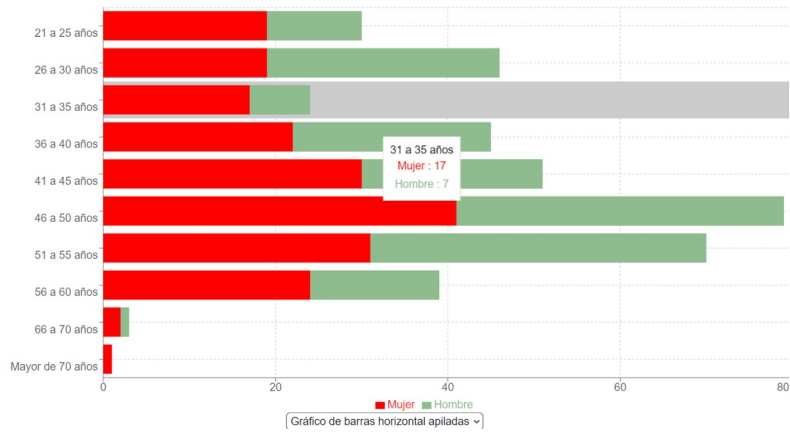


Figura 8.5: Gráfica de barras horizontales apiladas

Todos los tipos de gráficas se muestran igual para cualquier tipo de usuario, ya sea administrador o cliente.

Las siguientes imágenes son el resultado final del mapa de iniciativas y el formulario para publicar una de estas. Este diseño si difiere de lo que ves cuando eres administrador.

En la figura 8.6 se puede ver el mapa en el que los usuarios pueden consultar las iniciativas disponibles y filtrarlas. En la figura 8.7 se muestra el formulario para publicar una nueva.

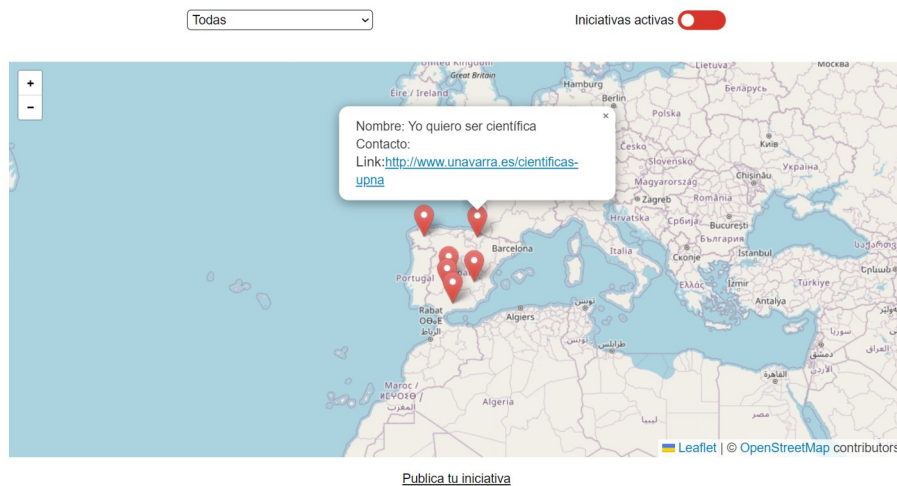


Figura 8.6: Mapa de iniciativas para usuario común

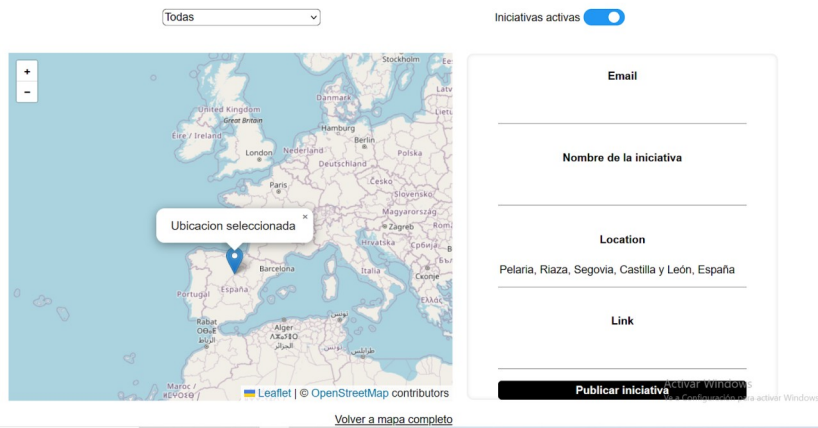


Figura 8.7: Formulario de publicación de iniciativa

## 8.2 Administrador

En la sección de administrador, tienes acceso al manejo de datos que se pueden visualizar en la aplicación. A continuación, se describen las funciones correspondientes a cada una de las imágenes:

- Figura 8.8: Selección de ejes al cargar un archivo CSV: En esta imagen, se muestra la interfaz para seleccionar los ejes (columnas) que se desean utilizar al cargar un archivo CSV en la aplicación. Esto permite personalizar qué datos se mostrarán en las visualizaciones gráficas.
- Figura 8.9: Administra la ponderación que se aplicará a los datos que vemos en las gráficas.
- Figura 8.10: Selección de gráficas: Aquí se presenta la interfaz para seleccionar el tipo de gráfico recomendado que se desea mostrar como primera opción al usuario.
- Figura 8.11: Formulario en el que se puede editar el nombre y descripción de los datos añadidos.
- Figura 8.12: Visión general de los archivos CSV en la base de datos: En esta imagen, se muestra una vista general de los archivos CSV que están almacenados en la base de datos. Esto proporciona una visión general de los datos disponibles y permite gestionarlos, como editar, eliminar o agregar nuevos archivos CSV.

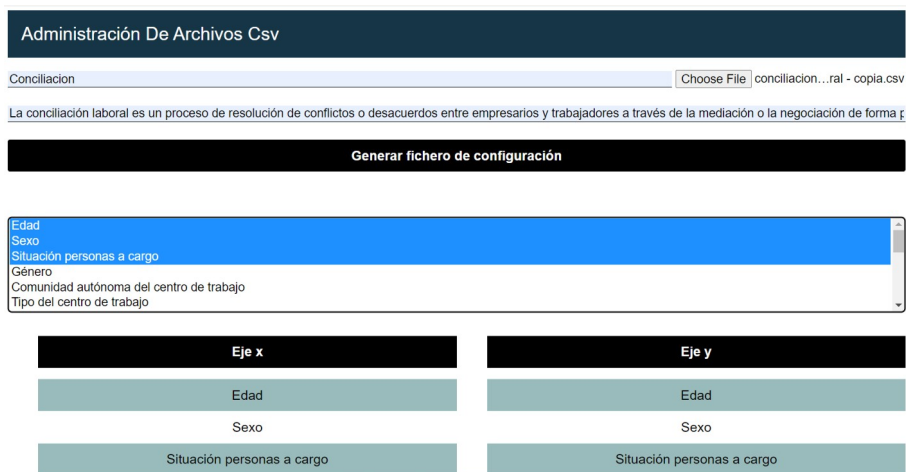


Figura 8.8: Selector de ejes



Figura 8.9: Administrador de ponderación

| Campo 1 | Campo 2                                  | Gráfica Recomendada                   | Descripción |
|---------|--|---------------------------------------|-------------|
| Edad    | Sexo                                     | Gráfico de barras                     | opcional    |
| Edad    | Situación personas a cargo               | Gráfico de barras                     |             |
| Sexo    | Situación personas a cargo               | Gráfico de barras apiladas            |             |
| Edad    | Género                                   | Gráfica circular                      |             |
| Edad    | Comunidad autónoma del centro de trabajo | Gráfico de barras horizontal          |             |
| Edad    | Tipo del centro de trabajo               | Gráfico de barras horizontal apiladas |             |
| Edad    |  | Gráfico de barras                     | opcional    |
| Edad    |  | Gráfico de barras                     | opcional    |

Figura 8.10: Selector de gráfica por defecto

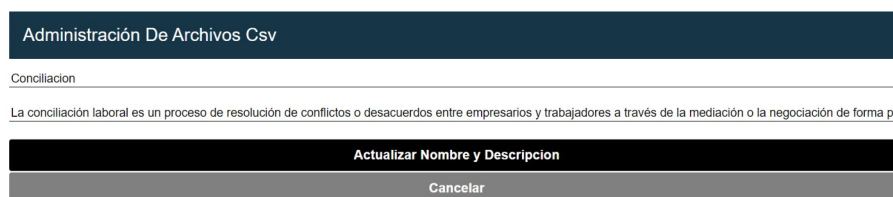


Figura 8.11: Administrador de descripción del conjunto de datos



Figura 8.12: Administrador de ficheros csv

En la sección del mapa de iniciativas, la imagen mostrada es similar a la vista de un usuario común, pero con una diferencia importante: se ha agregado un filtro adicional para mostrar solo las iniciativas que aún no han sido validadas. Esto permite al administrador tener una visión más específica y controlada de las iniciativas en el mapa.

En cuanto al formulario que se muestra en la segunda imagen, se pueden apreciar varias funcionalidades adicionales en comparación con la vista de un usuario común. Además de poder ver todas las iniciativas, el formulario ofrece opciones de actualización y eliminación de iniciativas. Esto brinda al administrador la capacidad de modificar y eliminar iniciativas existentes, lo que le otorga un mayor control sobre los datos y la gestión de la plataforma.

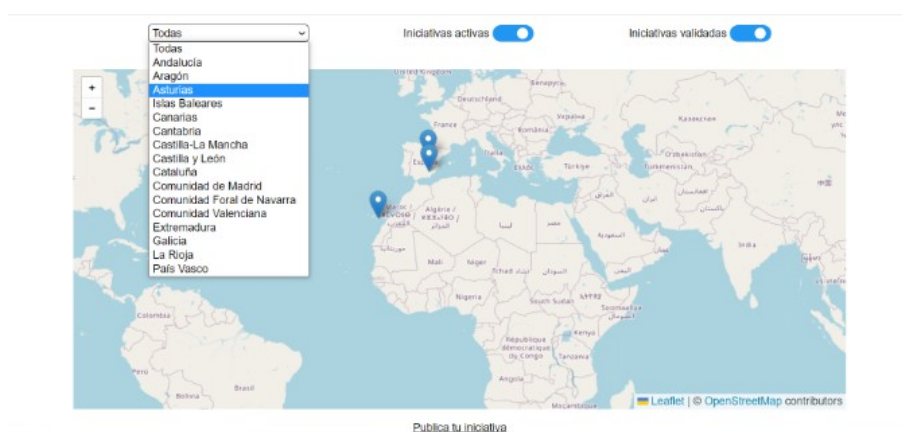


Figura 8.13: Mapa de iniciativas del administrador

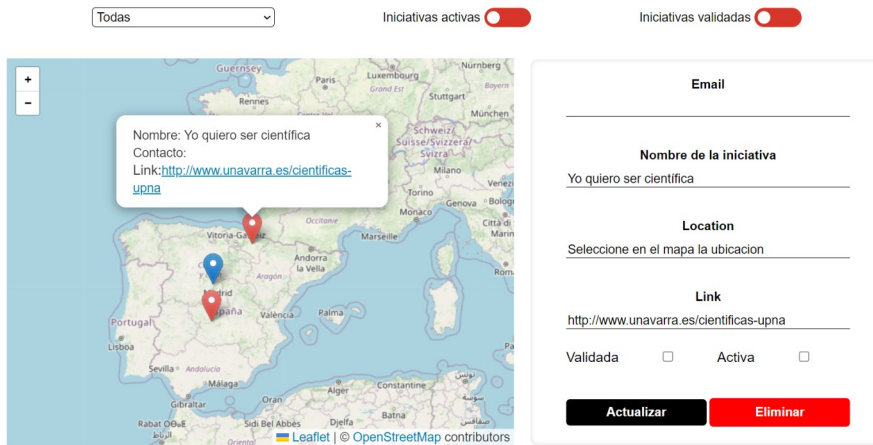


Figura 8.14: Administrador de iniciativas

# Capítulo 9

## Conclusiones y líneas futuras

A medida que aumenta la conciencia sobre los problemas relacionados con la igualdad de género y la discriminación de género, se hace cada vez más importante contar con herramientas que permiten a las personas acceder a información y recursos sobre estos temas.

La visualización de datos es una herramienta poderosa para crear conciencia sobre temas complejos. En el caso de Women@INF, la visualización de datos en gráficos puede ayudar a los usuarios a comprender mejor los problemas que enfrentan las mujeres en diferentes ámbitos y a identificar patrones y tendencias importantes.

La participación ciudadana es un aspecto importante en la lucha por la igualdad de género y contra la discriminación. La posibilidad de publicar iniciativas en un mapa es una forma de fomentar la participación ciudadana y de crear una comunidad de personas comprometidas con estos temas.

Las plataformas como Women@INF tienen un gran potencial para generar un cambio social positivo. Al permitir que las personas compartan información, recursos y experiencias, estas plataformas pueden inspirar a otros a tomar medidas concretas para abordar los problemas relacionados con la igualdad de género y la discriminación de género.

La tecnología puede ser una herramienta valiosa para fomentar la igualdad de género y la justicia social, pero también es importante tener en cuenta los posibles riesgos y desafíos. Es necesario abordar temas como la privacidad, la seguridad y la representatividad en las plataformas digitales que abordan temas sensibles y complejos como la igualdad de género.

Las futuras líneas serían añadir nuevas gráficas en caso de que los tipos de datos que se inserten necesitan de más variedad, implementar un sistema de sesión con cuenta de Google u otras redes sociales si se ve oportuno y añadir un sistema de sugerencias para seguir mejorando.

# Capítulo 10

## Summary and Conclusions

As awareness of gender equality issues and gender discrimination increases, it becomes increasingly important to have tools that enable people to access information and resources on these topics.

Data visualization is a powerful tool for raising awareness about complex issues. In the case of Women@INF, visualizing data in charts can help users better understand the challenges women face in different fields and identify important patterns and trends.

Citizen participation is an important aspect in the fight for gender equality and against discrimination. The ability to publish initiatives on a map is a way to encourage citizen participation and create a community of people committed to these issues.

Platforms like Women@INF have great potential to generate positive social change. By allowing people to share information, resources, and experiences, these platforms can inspire others to take concrete actions to address gender equality and gender discrimination issues.

Technology can be a valuable tool for promoting gender equality and social justice, but it is also important to consider potential risks and challenges. Issues such as privacy, security, and representativeness need to be addressed in digital platforms that tackle sensitive and complex topics like gender equality.

Future developments could include adding new types of charts in case the inserted data requires more variety, implementing a session system with Google or other social media accounts if deemed appropriate, and incorporating a feedback system to continue improving.



# Capítulo 11

## Presupuesto

Todas las tecnologías que se han utilizado para este proyecto no tienen ningún costo por lo que solo se tendrá en cuenta la mano de obra.

### 11.1 Presupuesto desglosado

| <b>Tarea</b>  | <b>Nº de horas</b> | <b>Precio por hora</b> |
|---|--------------------|------------------------|
| Diseño de la aplicación                                   | 20                 | 20 €                   |
| Instalación y configuración de las tecnologías necesarias | 8                  | 20 €                   |
| Desarrollo de Backend                                     | 120                | 20 €                   |
| Desarrollo de Frontend                                    | 150                | 20 €                   |
| Depuración y refactorización                              | 10                 | 20 €                   |

**Tabla 11.1: Presupuesto**

# Capítulo 12

## Apéndice A

### 12.1 Repositorios

Frontend : <https://github.com/alu0101130675/tfg-pruebas>

Backend : <https://github.com/alu0101130675/backend-TFG>

# Bibliografía

[1] Paderewski-Rodríguez, P., García-Arenas, M. I., Gil-Iranzo, R. M., González, C. S., Ortigosa, E. M., & Padilla-Zea, N. (2017). Initiatives and strategies to encourage women into engineering. *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, 12(2), 106-114.

[2] González-González, C. S., García-Holgado, A., de los Angeles Martínez-Estévez, M., Gil, M., Martín-Fernández, A., Marcos, A., ... & Gershon, T. S. (2018, April). Gender and engineering: Developing actions to encourage women in tech. In *2018 IEEE global engineering education conference (EDUCON)* (pp. 2082-2087). IEEE.

[3] Peixoto, A., González, C. S. G., Strachan, R., Plaza, P., de los Angeles Martínez, M., Blazquez, M., & Castro, M. (2018, April). Diversity and inclusion in engineering education: Looking through the gender question. In *2018 IEEE Global Engineering Education Conference (EDUCON)* (pp. 2071-2075). IEEE.

[4] García-Peñalvo, F. J., García-Holgado, A., Dominguez, A., & Pascual, J. (2022). Women in STEM in Higher Education: Good Practices of Attraction, Access and Retainment in Higher Education (p. 197). Springer Nature.

[5] Puig Puig, A., & Cirici, J. (2022). Mujeres que estudian informática y matemáticas: quiénes son y por qué son tan pocas. *The Conversation*, 2022.

[6] Ansari, B., Barati, M., & Martin, E. G. (2022). Enhancing the usability and usefulness of open government data: A comprehensive review of the state of open government data visualization research. *Government Information Quarterly*, 39(1), 101657.

[7] Universidad de Salamanca. <https://indicadores.usal.es/portal/personal/personal-docente-e-investigador/>

[8] En negro contra las violencias. <https://ennegrocontraasviolencias.gal/gl/mapa>

[9] React. <https://es.react.dev/reference/react>

[10] Express. <https://expressjs.com/>

[11] Mongoose. <https://mongoosejs.com/>

[12] Recharts. <https://recharts.org/en-US/api>

[13] React-leaflet. <https://react-leaflet.js.org/docs/api-map/>

[14] Leaflet. <https://leafletjs.com/>

[15] Nominatim. <https://nominatim.org/release-docs/develop/api/Overview/>

- [16] MDN. <https://developer.mozilla.org/es/>
- [17] Kinsta. <https://kinsta.com/blog/nodejs-vs-python/>
- [18] npm trends. <https://npmtrends.com/angular-vs-react-vs-vue>
- [19] Stateof. <https://2022.stateofjs.com/en-US/other-tools/>
- [20] Hevodata. <https://hevodata.com/learn/mongodb-vs-sq>