



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

PROYECTO DE TITULACIÓN

CARRERA: Ingeniería en Tecnologías de la información

TEMA: Desarrollo de una aplicación web para visualización de datos espaciotemporales

AUTORES: Héctor Ismael Cedeño Zambrano

Wilmer José Solano Llano

TUTOR: Ing. Armando Ortiz

Santo Domingo de los Tsáchilas, 1 de Marzo 2024

Reporte de verificación de contenido



Plagiarism and AI Content Detection Report

HectorCedeno_SolanoWilmer_TIC.pdf

Scan details

Scan time:
March 1th, 2024 at 14:49 UTC

Total Pages:
51

Total Words:
12749

Plagiarism Detection



| Types of plagiarism | | Words |
|---------------------|-------|-------|
| Identical | 0.2% | 22 |
| Minor Changes | 0% | 0 |
| Paraphrased | 0.6% | 77 |
| Omitted Words | 10.9% | 1393 |

AI Content Detection



| Text coverage | | Words |
|---------------|------|-------|
| AI text | 0% | 0 |
| Human text | 100% | 11356 |

[Learn more](#)

Firma:

Ing. Luis Armando Ortiz Delgado

C. C. 1311722191



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Certificación

Certifico que el trabajo de integración curricular: “**Desarrollo de una aplicación web para visualización de datos espaciotemporales**” fue realizado por el señor **Cedeño Zambrano, Héctor Ismael** y el señor **Solano Llano Wilmer José**, el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizada en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Santo Domingo, 1 de marzo del 2024

Firma:

Ing. Luis Armando Ortiz Delgado

C. C. 1311722191



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Responsabilidad de Autoría

Nosotros, **Cedeño Zambrano, Héctor Ismael**, con cédula de ciudadanía N.º 2300533193 y **Solano Llano, Wilmer José**, con cédula de ciudadanía N.º 2300071608, declaramos que el contenido, ideas y criterios del trabajo de integración curricular: **Desarrollo de una aplicación web para visualización de datos espaciotemporales** es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Santo Domingo, 1 de marzo del 2024

Firma:

Firma:

Cedeño Zambrano, Héctor Ismael

Solano Llano, Wilmer José

C.C: 2300533193

C.C: 2300071608



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Autorización de Publicación

Nosotros, **Cedeño Zambrano, Héctor Ismael** con cédula de ciudadanía N.º 2300533193 y **Solano Llano, Wilmer José**, con cédula de ciudadanía N.º 2300071608, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de integración curricular: **Desarrollo de una aplicación para visualización de datos espaciotemporales** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Santo Domingo, 1 de marzo del 2024

Firma:

Firma:

Cedeño Zambrano, Héctor Ismael

Solano Llano, Wilmer José

C.C: 2300533193

C.C: 2300071608

ÍNDICE DE CONTENIDO

| | |
|--|----|
| Dedicatoria | I |
| Agradecimiento | II |
| Resumen | 1 |
| Abstract | 2 |
| I. Introducción y estado del arte | 3 |
| <i>A. Introducción</i> | 3 |
| <i>B. Antecedentes</i> | 5 |
| <i>C. Justificación</i> | 6 |
| <i>D. Alcance</i> | 7 |
| <i>E. Estados del arte</i> | 8 |
| <i>F. Objetivos</i> | 10 |
| 1) <i>Objetivo general:</i> | 10 |
| 2) <i>Objetivos específicos</i> | 10 |
| II. Marco teórico | 11 |
| <i>A. Metodologías de desarrollo de software</i> | 11 |
| 1) <i>Design Thinking:</i> | 11 |
| 2) <i>Kanban:</i> | 12 |
| <i>B. Datos espacio-temporales</i> | 13 |
| <i>C. Three.js</i> | 14 |
| 1) <i>Características:</i> | 15 |
| 2) <i>WebGL:</i> | 16 |
| a) <i>Implementación:</i> | 16 |
| b) <i>Configuración de la escena:</i> | 16 |
| c) <i>Agregar objetos:</i> | 17 |

| | |
|-----------------------------------|----|
| d) Agregar cámara: | 17 |
| e) Agregar luces: | 17 |
| f) Renderizar la escena: | 18 |
| D. Tecnologías | 18 |
| 1) Node.js: | 18 |
| 2) React: | 18 |
| 3) OrbitControls: | 18 |
| E. Herramientas | 19 |
| 1) Visual Studio Code: | 19 |
| III. Metodología | 20 |
| A. Diseño del proyecto | 20 |
| B. Estructura del proyecto | 21 |
| C. Design thinking | 21 |
| 1) Empatizar: | 21 |
| 2) Definición: | 23 |
| 3) Ideación: | 25 |
| 4) Prototipo: | 26 |
| 5) Pruebas: | 30 |
| D. Kanban | 34 |
| 1) Roles: | 34 |
| 2) Tarjetas: | 35 |
| 3) Flujo de trabajo: | 42 |
| 4) Backlog: | 43 |
| 5) Criterios de aceptación: | 44 |
| 6) Tablero: | 46 |

| | |
|---|----|
| IV. Resultados..... | 48 |
| <i>A. Análisis de resultados</i> | 48 |
| V. Conclusiones y recomendaciones | 63 |
| <i>A. Conclusiones</i> | 63 |
| <i>B. Recomendaciones</i> | 64 |
| <i>C. Trabajos a futuro</i> | 64 |
| VI. Referencias bibliográficas..... | 65 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Fig. 1 Eje Espacio-Tiempo | 14 |
| Fig. 2. Interfaz principal | 27 |
| Fig. 3. Interfaz grafica | 28 |
| Fig. 4. Opciones | 28 |
| Fig. 5. Cubo base..... | 29 |
| Fig. 6. visualización 2D..... | 30 |
| Fig. 7. Visualización 3D | 30 |
| Fig. 8. Prueba de carga en JMeter | 31 |
| Fig. 9. Ingreso de archivos JSON..... | 32 |
| Fig. 10. Formulario..... | 33 |
| Fig. 11. Opciones disponibles | 33 |
| Fig. 12. Tarjeta de creación del proyecto en Visual Studio Code..... | 35 |
| Fig. 13. Verificar el tipo de JSON que se debe de manejar..... | 36 |
| Fig. 14. Crear la escena, cámara, luces y el cubo de la aplicación..... | 36 |
| Fig. 15. Colocar una imagen de un mapa dentro del cubo | 36 |
| Fig. 16. Obtener los datos del archivo JSON a través de un botón..... | 37 |
| Fig. 17. Crear botones para ocultar líneas y puntos | 37 |
| Fig. 18. Mostrar la información de los puntos en un mensaje flotante | 37 |
| Fig. 19. Estandarizar los datos en el tiempo para implementarla dentro del cubo..... | 38 |
| Fig. 20. Colocar el valor de z en horas en la etiqueta que muestra los puntos..... | 38 |
| Fig. 21. Tener la opción de pantalla completa..... | 38 |
| Fig. 22. Mejorar la presentación de las líneas | 39 |
| Fig. 23. Hacer que la información de los puntos se vea cuando no se está en pantalla completa.. | 39 |
| Fig. 24. Hacer los puntos redondos | 39 |
| Fig. 25. Hacer las opciones para el zoom y para el gráfico 2D..... | 40 |
| Fig. 26. Dar la opción de descargar reporte | 40 |
| Fig. 27. Letras/Cubo cara de arriba transparente, y Axes Helper..... | 40 |
| Fig. 28. Hacer un filtro de hora | 41 |
| Fig. 29. Mejorar el estilo de la presentación de la aplicación en la parte gráfica | 41 |
| Fig. 30. Hacer una página de presentación de la aplicación y de los autores..... | 41 |

| | |
|---|----|
| Fig. 31. Hacer filtro trayectorias | 42 |
| Fig. 32. Realizar pruebas de funcionalidad, y probar con datas reales | 42 |
| Fig. 33. Entrega del producto | 42 |
| Fig. 34. Flujo de trabajo | 43 |
| Fig. 35. Backlog | 44 |
| Fig. 36. Tablero Kanban | 47 |
| Fig. 37. Resultados de prueba de carga | 48 |
| Fig. 38. Archivo 1 | 49 |
| Fig. 39. Mensaje fuera de rango | 49 |
| Fig. 40. Archivo 2 | 50 |
| Fig. 41. Mensaje de hora inválida | 50 |
| Fig. 42. Archivo 3 | 51 |
| Fig. 43. Mensaje de formato de hora inválida | 51 |
| Fig. 44. Archivo 4 | 52 |
| Fig. 45. Mensaje de formato incorrecto | 52 |
| Fig. 46. Formulario resultados información | 53 |
| Fig. 47. Preguntas de funcionalidad en opciones | 53 |
| Fig. 48. Preguntas filtrado | 54 |
| Fig. 49. Preguntas de facilidad de uso, velocidad de cargar y visualización | 54 |
| Fig. 50. Calificación de visualización en 2D y 3D | 55 |
| Fig. 51. Respuesta de descarga de visualización | 55 |
| Fig. 52. Datos ejemplo | 56 |
| Fig. 53. Ocultar puntos | 57 |
| Fig. 54. Ocultar líneas | 57 |
| Fig. 55. Perspectiva 2D | 58 |
| Fig. 56. Pantalla completa | 59 |
| Fig. 57. Zoom + | 59 |
| Fig. 58. Zoom - | 60 |
| Fig. 59. Varias trayectorias | 61 |
| Fig. 60. Filtro de trayectorias | 61 |
| Fig. 61. Filtro | 62 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla I. Requerimientos funcionales..... | 23 |
| Tabla II. Requerimientos no funcionales..... | 24 |
| Tabla III. Roles y responsabilidades..... | 34 |
| Tabla IV. Criterios de Aceptación..... | 44 |

DEDICATORIA

Dedico este trabajo a mis padres Ismael Cedeño y Yesenia Zambrano, por apoyarme en seguir adelante con mis estudios. A mi padre por enseñarme a trabajar duro para conseguir las cosas, y a mi madre por aconsejarme en cada paso de mi camino. Ofreciéndome su amor incondicional y realizando un gran sacrificio para llegar a este momento. También quiero dedicar este trabajo a mis hermanas que han sido una fuente de inspiración para culminar mis estudios.

También hago honor a mi abuelita, que me mira desde el cielo y me ha estado apoyando desde ahí, además de mis familiares y amigos que siempre me han apoyado y motivado para la realización de este proyecto.

Héctor Ismael Cedeño Zambrano

DEDICATORIA

Dedico el trabajo concluido a mi familia por siempre apoyarme, haciendo el esfuerzo para que pueda seguir estudiando. Entre ellos mi mami Rosario Llano que siempre me pregunto cómo me iba, a mi papá Juan Solano por acompañarme en las madrugadas peligrosas. A mis hermanas Erika y Mary por apoyarme en su manera, dándome a mis sobrinas para desestresarme.

Por último, a Lionel Andrés Messi Cuccittini que con su simple existencia me ha alegrado los días, me motivo a seguir adelante sin importar las veces que cayera. Demostrando que tarde o temprano el tiempo pone todo en su lugar.

Wilmer José Solano Llano

AGRADECIMIENTO

Agradecemos al Ing. Luis Ortiz por su orientación y dedicación que nos brindó para el desarrollo de este trabajo, compartiendo sus altos conocimientos sobre el tema que nos permitió realizarlo. Además, de reconocer a los demás docentes que nos fueron impartiendo las asignaturas a lo largo de la carrera, entregándonos sus vastos conocimientos para nuestro aprendizaje. De la misma manera, se agradece a nuestros compañeros de estudio, por hacernos compañía y servirnos como apoyo en lo que fue el trayecto desde el inicio de la carrera hasta este momento.

Cedeño Héctor y Solano Wilmer

RESUMEN

En este trabajo se ha desarrollado una aplicación web para la visualización de datos espacio-temporales, mediante la técnica de “space-time cube” ofreciendo perspectivas tanto como en 2D como en 3D. Se enfoca en proporcionar una representación visual de las trayectorias por estos datos, utilizando ejemplos reales para ilustrar las funciones de la aplicación. La aplicación permite visualizar trayectorias a través de un formato JSON en específico, con opciones interactivas como ocultar/mostrar puntos y líneas, zoom, captura en PDF, filtrado en horas y trayectorias. La implementación se realizó utilizando la librería Three.js para gráficos 3D. React para el funcionamiento y presentación, integrada junto con Nodejs para su ejecución. Se adoptaron metodologías como Design Thinking para gestión del proyecto y Kanban para visualizar el flujo de trabajo. Destacando la importancia de la visualización de datos para el análisis y comprensión de la información manejada.

Palabras claves: datos espacio-temporales, visualización, representación, space-time cube, trayectorias.

ABSTRACT

In this work, a web application has been developed for the visualization of spatio-temporal data, using the “space-time cube” technique, offering perspectives in both 2D and 3D. It focuses on providing a visual representation of the trajectories through this data, using real examples to illustrate the functions of the application. The application allows you to view trajectories through a specific JSON format, with interactive options such as hiding/showing points and lines, zoom, PDF capture, filtering in hours and trajectories. The implementation was carried out using the Three.js library for 3D graphics. React for operation and presentation, integrated with Node.js for execution. Methodologies such as Design Thinking were adopted for project management and Kanban to visualize the workflow. Highlighting the importance of data visualization for the analysis and understanding of the information handled.

keywords: space-temporal data, visualization, representation, space-time cube, trajectories.

I. INTRODUCCIÓN Y ESTADO DEL ARTE

A. Introducción

En la actualidad, es de suma importancia destacar la generación constante de datos en el mundo, como lo son la distribución de internet, IoT, sistemas de transacción, redes sociales, entre otras. Donde se debe realizar un análisis para obtener los beneficios de esos datos, y la mejor forma de representarlos es a través de las visualizaciones. Esto se debe a que es algo fundamental para el análisis y forma en la que se da a entender la información. Para lograrlo, es fundamental contar con una representación visual que brinde a los usuarios una vía efectiva para interpretar la información. Permitiendo la revelación de patrones y tendencias de una manera más fácil y accesible, debido a que se trabaja con conjuntos de datos extensos. Esto facilita la interpretación de datos complejos que ayudan a obtener conclusiones más acertadas, con el objetivo de lograr una comprensión más eficiente basada en los datos presentados. Los datos pueden tener distintas formas de representación, con el fin de otorgar un significado a los resultados que se trata de dar a conocer. Esta representación se realiza a través de ciertos gráficos, debido a que otorgan una representación visual más detallada gracias al avance computacional.

Dentro de la información que se puede encontrar existen algunos tipos de datos complejos los cuales pueden ser: Datos jerárquicos, de red, multimedia, textuales estructurales, financieros, redes neuronales, voz y audio, genómicos, biométricos, y espacio-temporales. Para este proyecto se basará en los datos espacio-temporales, debido a que van registrando la evolución de objetos o fenómenos a lo largo del tiempo y el espacio. Algunos ejemplos de los datos espacio-temporales son: el seguimiento de vehículos, registros climáticos, eventos geofísicos, entre otros que van trazando una trayectoria. Esto se debe a que este tipo de datos representa el recorrido o colocación de varios puntos representados dentro de una zona geográfica, eje de coordenadas, entre otros ejemplos de un espacio que van creando trayectorias a través del tiempo. La generación de las trayectorias a través de los datos espacio-temporales son importantes para su gestión, con el fin de identificar patrones y tendencias. Además de tener una mejor representación visual al presentarse desde una perspectiva 2D y en 3D.

Para presentar de manera visual las trayectorias de este tipo de datos se pueden implementar distintas técnicas. En la representación 2D se pueden utilizar gráficos de dispersión y línea de trayectoria. Por otro lado, la 3D se puede utilizar la técnica llamada “space-time cube”. Los gráficos

de dispersión y líneas de trayectoria en 2D ofrecen una representación clara de las rutas formadas por las distintas posiciones. Mientras que el "space-time cube" en 3D proporciona una visión holística al incorporar dimensiones espacio-temporales. Estas se implementan con el fin de dar una visualización detallada de los datos incorporados. El presente proyecto trata de proveer estas 2 opciones de visualización mediante una aplicación web, con el objetivo de representar los datos espacio-temporales. A través de la aplicación web se podrá ingresar, visualizar, interactuar y filtrar con los datos de forma eficaz, utilizando la técnica de "space-time cube" para su representación.

La aplicación web se desarrollará principalmente a través de los lenguajes y librerías de javascript, nodejs, react y three.js. Esta última permitirá la representación y visualización de los datos que se encuentran representados en el tiempo y el espacio. De esta manera se logra obtener una forma de poder interpretar los datos espacio-temporales que se poseen. Es decir, el uso de la librería de Three.js ayudará a la creación de una gráfica para la visualización de los datos espacio-temporales de manera dinámica. Para representar los datos dentro de la aplicación estos son importados desde un archivo JSON con su respectivo formato. Y esta muestra la visualización de estos datos desde las perspectivas 2D y 3D por medio de la técnica de "space-time cube". Para el desarrollo de este trabajo se utilizó React para la interfaz y el funcionamiento de la aplicación, y Nodejs para la ejecución del mismo.

En este trabajo se propone utilizar la técnica de "space-time cube" debido a que la aplicación trata de representar los datos espacio-temporales, y a través de esta se visualizan las trayectorias que se forman de manera tridimensional. El espacio se encuentra representado en dos dimensiones, es decir es un plano (x, y) , y la tercera dimensión se encuentra representada por el tiempo. Esto con la finalidad de analizar la información de las trayectorias a través del tiempo en el que se desplaza. La técnica de "space-time cube" sirve para presentar dentro de un escenario la información de los datos espacio-temporales que se quiera visualizar a través de la creación de un ambiente dinámico tridimensional [11]. Esta aplicación web implementa esta técnica con la finalidad de presentar a los usuarios una perspectiva 2D y 3D de las trayectorias y puntos que se forman a través de estos datos. Con la finalidad de analizar y comprender eventos complejos por medio de este tipo de visualización.

La información de los datos espacio-temporales hace referencia a componentes espaciales como temporales. Sus registros normalmente representan puntos en zonas geográficas que se van

capturando a través del tiempo. En otras palabras, tratan de dar a conocer sobre los datos de fenómenos u objetos que se van moviendo en el transcurso del tiempo. Es decir, va registrando la posición en la que se encuentra un objeto o suceso, tanto en el espacio o zona, como en el tiempo, lo que forma una trayectoria. Algunas de las formas en la que se pueden mostrar este tipo de información son por medio de mapas, tipos de superficies, diagramas y plano de coordenadas como ejemplos. Este tipo de información se puede guardar en diferentes tipos de archivos como lo pueden ser JSON, XML, HDF, etc [13]. Para el funcionamiento de esta aplicación web se usará el archivo tipo JSON para cargar la información de los datos espacio-temporales y poder visualizarlos.

El desarrollo de esta aplicación web es aplicada a través del criterio de las metodologías ágiles, tales como Design Thinking y Kanban. Esto con la finalidad de implementar una mayor gestión en el desarrollo del proyecto, adaptándose a las necesidades que se presenten y las mejores que se deben implementar dentro del sistema. A través del Design Thinking se trata de entender las necesidades del usuario, con el fin de mostrar una solución innovadora para la visualización de información. Por otro lado, la metodología Kanban se integra a través del manejo del flujo de trabajo y el control de los procesos.

B. Antecedentes

El avance tecnológico en las últimas décadas ha impulsado la generación masiva de datos espacio-temporales provenientes de diversas fuentes, como sensores remotos, dispositivos GPS, redes sociales y sistemas de información geográfica. Incluso se pueden aplicar modelos que se encargan de generar información espacio-temporal [1]. Este crecimiento exponencial de la información geoespacial y temporal ha creado la necesidad de herramientas innovadoras que faciliten la comprensión y análisis de estos datos complejos.

En los últimos años se ha identificado un crecimiento en las investigaciones relacionadas con la visualización de datos espacio-temporales. Las cuales se centran mayoritariamente en los métodos de visualización con sus respectivas técnicas y los campos de aplicación [18]. También se ha visto la tendencia de usar bibliotecas de lenguajes de programación para una visualización interactiva.

Como lo es la biblioteca ST_VISIONS en Python, la cual permite coloración y filtrado interactivo de los datos [19]. Sin embargo, a pesar de la creciente importancia de estos datos, las

herramientas disponibles para su visualización suelen carecer de la flexibilidad y la capacidad de personalización necesarias para adaptarse a las diversas necesidades de los usuarios. En este contexto, el desarrollo de aplicaciones web para la visualización de datos espacio-temporales emerge como una solución prometedora.

En las últimas décadas, se ha pasado de depender principalmente de gráficos bidimensionales y tablas estáticas a explorar visualizaciones dinámicas y tridimensionales que ofrecen una comprensión más profunda de los datos. La aparición de software especializado, como Tableau, Power BI y D3.js, ha mejorado la creación de visualizaciones. También se han creado herramientas como Mapyzer y SpatialEpiApp [2] que permiten la carga y visualización de datos espacio-temporales. Ambas herramientas incluyen funciones como datos estadísticos, visualizaciones interactivas e información.

El objetivo central de este proyecto es la creación de una aplicación web innovadora que posibilite la visualización intuitiva y eficaz de datos espacio-temporales. El enfoque se centra en la capacidad de la aplicación para integrarse fluidamente con diversas fuentes de información geográfica, como los archivos JSON. Y lograr la representación visual de los datos tanto en entornos 2D como 3D. Este proyecto no solo enfrenta el desafío técnico de fusionar datos complejos en un formato accesible.

Sino que también responde a la creciente relevancia de comprender y comunicar patrones espacio-temporales en diversas disciplinas, desde la planificación urbana hasta la investigación científica. La aplicación resultante no solo llenará un vacío tecnológico, también proporcionará una herramienta valiosa para profesionales y académicos que buscan explorar y analizar de una manera más eficaz datos espacio-temporales.

La comprensión y gestión eficaz de datos espacio-temporales son esenciales para informar decisiones estratégicas, optimizar procesos operativos, anticipar patrones y tendencias en diversos campos. Este estudio busca contribuir a la evolución de las herramientas disponibles, proporcionando a los usuarios una plataforma accesible y potente para explorar y comprender la complejidad de los datos espacio-temporales en diversos contextos.

C. Justificación

La visualización de datos espacio-temporales es una técnica que permite representar información espacial y temporal de manera visual [3]. Esta técnica se utiliza en una amplia gama de aplicaciones, como la ciencia, la ingeniería, la planificación y la gestión. Por ejemplo, se emplea la visualización espacio-temporal para analizar las alteraciones de los casos COVID-19 generando gráficos de globos asociados al tiempo [4]. En este contexto, el desarrollo de una aplicación web para la visualización de datos espacio-temporales surge como una respuesta esencial para abordar las limitaciones actuales en la representación y análisis de información geoespacial.

La relevancia de esta investigación radica en la capacidad de la aplicación propuesta para transformar la manera en que interactuamos con datos espacio-temporales. La mayoría de las herramientas existentes tienden a enfocarse en la representación en 2D, limitando la percepción y comprensión de la información en entornos tridimensionales. La incorporación de la tercera dimensión no solo amplía las posibilidades de análisis, sino que también permite una interpretación más precisa de la información. Especialmente en campos críticos como la planificación urbana, la gestión ambiental y la monitorización de fenómenos climáticos.

La importancia del desarrollo de una aplicación web para visualizar datos espacio-temporales se destaca en su potencial para empoderar a diversos sectores. Desde profesionales en geociencias hasta planificadores urbanos y tomadores de decisiones gubernamentales. La posibilidad de visualizar datos espacio-temporales en un entorno 3D y 2D integrado no solo facilita la identificación de patrones y tendencias, sino que también mejora la comunicación y comprensión de los resultados, siendo esencial para la toma de decisiones informadas.

D. Alcance

El alcance de este trabajo presenta una aplicación web que muestra las trayectorias de los datos espacio-temporales a través de la importación de un archivo JSON. Para su implementación se utiliza la técnica de visualización de “space-time cube”, con la finalidad de que represente gráficamente los puntos y trayectorias que se forman en 2D y 3D. De esta manera se gestiona la visualización de los datos espacio-temporales a través de esta aplicación para su respectivo análisis. Para utilizarla el usuario debe cargar el archivo con los datos del formato antes mencionado.

El funcionamiento de la aplicación web trata sobre presentar los datos espacio-temporales a través de la técnica de “space-time cube” para visualizarlos desde las perspectivas de 2D y 3D.

Esto ayuda a que el usuario pueda identificar patrones entre los datos otorgados que forman puntos y trayectorias en el tiempo y espacio. En otras palabras, la aplicación permite que los usuarios que quieran tener una visualización de las trayectorias que se forman en su data. Y para ello deben cargar un archivo JSON con un formato en específico, lo que facilita obtener un mayor análisis sobre la complejidad de los datos.

Para la visualización de los datos desde la perspectiva 2D se representará a través de un cuadro (el cual es la cara superior del cubo) bajo las coordenadas (x, y). Por otro lado, desde la perspectiva 3D se mostrará un cubo interactivo que muestra los puntos y trayectorias de las coordenadas a través del tiempo. Para lograr estas representaciones gráficas se utiliza la técnica de “space time cube”, para mostrar las diferentes perspectivas en la que se pueden visualizar las trayectorias que se forman a través de los datos otorgados.

La implementación de trabajo aspira ofrecer una forma de visualización y comprensión de los datos espacio-temporales, a través de una aplicación web que se encarga de presentar de manera gráfica y desde las perspectivas 2D y 3D este tipo de información. Además, esta aplicación tiene otras funcionalidades como lo son ocultar y mostrar puntos y líneas, descarga del escenario, zoom del cubo, pantalla completa, volver a perspectiva 2D, el filtrado de los datos a través de un rango de horas y uno seleccionando la trayectoria que se quieren visualizar. Lo que permite una mayor flexibilidad en la exportación de los conjuntos de datos, potenciando la experiencia del usuario. Estos usuarios pueden ser analistas de datos, profesionales de logística y transporte, investigadores de ciencia espaciales y temporales, o algún interesado en la visualización de datos espacio-temporales.

Además, toca resaltar que el proyecto no generará un archivo JSON, sino que este vendrá por parte del usuario según los parámetros establecidos. De la misma forma no ofrecerá análisis avanzados como predicciones o modelado estadístico. Excluyendo otras formas complejas de datos como genómicos, biométricos u otros tipos no relacionados a lo que son los datos espacio-temporales. Ya que el proyecto se enfocará específicamente en la visualización eficiente de datos espacio-temporales y sus trayectorias.

E. Estados del arte

1) Visualización de datos Espacio-Temporales de predicción de eventos de emergencia en Ecuador:

Se centra en el desarrollo de una herramienta que se encarga de la visualización de datos espacio-temporales. Dicha visualización sería esencial para el análisis, captura y predicción en tiempo real de los eventos de emergencia que suceden en Ecuador. Se llevaron a cabo cuatro etapas en el desarrollo de la herramienta [16].

Existen estudios que, en la primera etapa, se adquirieron y transformaron datos desde Twitter y el registro de llamadas de emergencia del ECU 911. En la segunda etapa, se diseñó la arquitectura del prototipo con un enfoque en el Modelo Vista Controlador (MVC), resaltando los módulos de almacenamiento, predicción y visualización de datos. La tercera etapa implicó el desarrollo de la herramienta utilizando diversas bibliotecas para generar visualizaciones de datos espacio-temporales. En la última etapa, se demostró la funcionalidad de la herramienta web, destacando su capacidad para adquirir, almacenar, visualizar y analizar predictivamente eventos de emergencia.

Dicho estudio obtuvo como resultados gráficos en relación con los datos espacio-temporales. Entre las gráficas que se destacan se encuentra el mapa de calor de Ecuador, que señala las zonas con mayor emergencia. También se puede encontrar otro gráfico que hace referencia a las emergencias por meses en relación con las emergencias principales del país. Como lo son la emergencia de tránsito, municipal, sanitarias, entre otros.

2) Comparativa de visualización de trayectorias espaciotemporales 2D vs 3D:

Este trabajo presenta cómo realizar un análisis de datos espacio-temporales que se muestran desde una perspectiva 2D y 3D. Esto se aplica a través de la visualización de los datos que pueden estar representados mediante la técnica de “space-time cube”, el cual es un cubo que representa las trayectorias de manera tridimensional. Este cubo contiene un mapa en su base para indicar cómo se va trazando la trayectoria desde ambas perspectivas, es decir, la 2D y 3D [21].

Para el desarrollo de este trabajo se utilizó la librería de Three.js para la visualización 3D donde presenta los datos de la trayectoria con respecto al tiempo. Para la 2D se usó D3.js, la cual solo va a representar los espacios, y se explica las características y uso de estos dos tipos de librerías. Un ejemplo es el trabajo de [21] donde en la creación de los gráficos 2D y 3D se

seleccionaron los datos de un parque de diversiones ficticio. Los cuales son obtenidos a partir de una aplicación que otorga un id que se asignan a los usuarios, para registrar el camino que van recorriendo. Se obtiene la información colocada en diferentes campos como son el tiempo, ejes (x,y), el tipo y el id.

Para visualizar en 2D los datos espacio-temporales se toma principalmente los ejes (x,y) para trazar los caminos representados en un mapa. Donde el mapa, los puntos, ejes y rutas se encuentran separadas por capas. En el caso de 3D usará la técnica de “space-time cube” para mostrar los datos desde un plano lateral para observar sus medidas con el tiempo. Además de colocar un mapa en el plano inferior, y sobre estas una serie de líneas que representan la trayectoria de una persona con respecto al tiempo.

F. Objetivos

1) Objetivo general:

Desarrollar una aplicación web que permita visualizar datos espacio-temporales en 2D y 3D utilizando la técnica de “space-time cube”.

2) Objetivos específicos

- Utilizar la metodología de desarrollo de software Design Thinking para planificar la ejecución del proyecto, la metodología Kanban para optimizar el flujo de trabajo.
- Desarrollar una aplicación web para visualizar los datos espacio-temporales a través de las librerías de React de JavaScript y Three.js.
- Demostrar el funcionamiento correcto de la aplicación web realizando las pruebas.

II. MARCO TEÓRICO

A. Metodologías de desarrollo de software

1) Design Thinking:

La metodología Design Thinking se centra primordialmente en el proceso de diseño en lugar de poner un énfasis exclusivo en el resultado final. Este método se distingue por su capacidad para integrar conocimientos técnicos provenientes del ámbito del diseño, las ciencias sociales, la gestión empresarial y la ingeniería. Se busca no solo la resolución de problemas de manera eficaz, sino también la generación de soluciones innovadoras y centradas en las necesidades y experiencias de los usuarios [5].

La metodología Design Thinking inicia su proceso mediante la definición clara del problema antes de emprender el desarrollo de cualquier solución, poniendo un énfasis particular en la comprensión profunda de las necesidades del cliente. Al enfocarse en el problema del cliente, se posibilita la creación de un prototipo que representa una solución tentativa. Este prototipo se somete a pruebas, lo que brinda la oportunidad de continuar aprendiendo y perfeccionando la solución [6]. Este enfoque secuencial y centrado en el usuario no solo asegura la resolución efectiva de los problemas planteados, sino que también fomenta un proceso de mejora continua basado en la retroalimentación constante. Y la adaptabilidad a medida que evoluciona la comprensión del contexto y las necesidades del usuario.

A continuación, se presentan las etapas que se emplean en la metodología Design Thinking:

- **Empatizar:** Comprender las necesidades y experiencias del usuario a través de la observación, la conversación y la empatía profunda.
- **Definir:** Establecer claramente los requisitos del problema después de la fase de empatía, proporcionando enfoque y claridad a los parámetros del diseño.
- **Idealizar:** Generar una amplia gama de ideas para abordar los problemas definidos, fomentando la creatividad y explorando soluciones innovadoras.
- **Prototipado:** Crear representaciones tangibles, desde notas hasta productos reales, para obtener retroalimentación y mejorar el diseño de manera iterativa

- Prueba: Obtener retroalimentación del usuario mediante la prueba de prototipos, ajustar y perfeccionar continuamente el diseño para abordar eficazmente los problemas identificados.

2) Kanban:

La metodología de desarrollo de software Kanban se encarga de la gestión de proyectos a través del control de los procesos de un flujo de trabajo, los cuales son representados a través de un tablero. Esta metodología se encarga de mejorar la calidad de desarrollo, debido a que se mantiene un control sobre la carga de trabajo y se logra visualizar el tablero de trabajo [10]. Se encarga de mostrar el flujo de trabajo que se encuentra organizado por columnas, las cuales normalmente se encuentran compuestas por nombres de “Por Hacer”, “En Proceso” y “Hecho”.

Kanban se centra en mostrar una ágil y sencilla gestión de trabajo, de manera que no presente problemas y responda de manera más efectiva a los cambios que se presenten. La metodología utiliza lo que son tableros para representar visualmente las tareas, con el fin de mejorar la eficiencia del trabajo en equipo y sus procesos. Estos tableros son los encargados de mostrar el avance del proyecto, de manera que cada uno de los integrantes del proyecto pueden observar cómo va yendo este. De esta forma se comprueba el cumplimiento de cada una de las tareas asignadas según el tiempo planificado [14].

a) Elementos de Kanban:

Tablero: Es una tabla que indica la representación gráfica del flujo de trabajo del proyecto, la cual se encuentra divididas por columnas que representan las diferentes etapas que lleva cada proceso. Estas columnas pueden adaptarse según las necesidades del proyecto.

Tarjetas: Contienen la información sobre la tarea encomendada en el proyecto, es decir representa los trabajos individuales que debe hacer cada integrante del proyecto. Contiene información clave como descripción, estado, el encargado y otros detalles. Cada una de estas se pueden colocar en las diferentes columnas para indicar en que etapa del proceso se encuentran.

Columnas: Se encargan de representar las etapas de la tabla del flujo de trabajo o como se divide las fases o estados específicos del proceso, las cuales pueden ser “Por Hacer”, “En Proceso”, “Pruebas” y “Hecho”. Es decir, que cada columna indica en que etapa del proyecto se encuentra cada tarea, desde su inicio hasta su finalización.

Límites de Trabajo en Proceso (WIP): Son las restricciones establecidas para controlar el número de tareas asignadas que se encuentren activas en las columnas. Esto con el fin de evitar la sobrecarga de trabajo en cada una de las fases del proyecto, de manera que ayuda a mantener un flujo constante en el desarrollo de este.

Priorización: A pesar de que no es exactamente un elemento de Kanban, es importante priorizar cada una de las tareas. Ya que esto permite realizar en primer lugar las tareas más importantes para el desarrollo del proyecto.

b) Reglas de Kanban:

Según Rodríguez [17], la metodología de Kanban se compone principalmente de cinco reglas las cuales son:

1. Organizar visualmente las diferentes partes del proyecto, utilizando un tablero el cual se compone por columnas que separan cada una de las fases y tarjetas que representan las tareas. De esta manera se logra obtener una visión clara del progreso del trabajo, comprendiendo las responsabilidades de cada miembro del equipo.
2. Se debe completar una tarea antes de iniciar con otra con el fin de garantizar el éxito del proyecto. Para ello se debe limitar la cantidad de tareas que se pueden realizar en simultáneo. Su número debe ser bastante alto para que el personal esté ocupado, pero lo suficientemente reducido como para evitar priorizar una tarea tempranamente.
3. Establecer políticas para definir los criterios de aceptación de cada fase del proyecto, con el fin de dar a entender los procesos de trabajo. De manera que se cumplan las condiciones establecidas para el avance de las tareas en las diferentes etapas.
4. Medir y administrar el flujo de trabajo para evaluar su velocidad y estado en los equipos para las diferentes etapas del proyecto.
5. Realizar una mejora continua al tener el tablero actualizado que facilita la toma de decisiones. Debido a que se evita el retraso de las tareas o la acumulación de estas, por ende, se realizan reuniones periódicas.

B. Datos espacio-temporales

Los datos espacio-temporales se refieren a la información que se representa a través de las dimensiones del espacio y el tiempo. Este tipo de datos representan los registros del cambio de

ubicación geográfica e identificación de eventos a lo largo del tiempo. Es decir, son datos de posicionamiento que se incorpora a la dimensión temporal para que se pueda clasificar como datos espacio-temporales. Esto con el fin de proporcionar una representación de como los fenómenos, objetos o eventos cambian o pasan por diferentes lugares a través en cierto momento o tiempo [13].

Estos datos se pueden representar por a través de coordenadas (x, y, z) como se muestra en la Figura 1, donde (x, y) representan la ubicación o posicionamiento geográfico del fenómeno u objeto, lo que vendría a ser el espacio. El eje “z” representa el tiempo en que se van ubicando cada una de esas posiciones o puntos, y la unión de cada uno de estos crean lo que es una trayectoria. Una de las formas en la que se puede representar es a través de un cubo que se componga de los ejes (x, y, z) que integran el tiempo y el espacio. Donde el eje “z” determina la compresión de un evento en un determinado tiempo, bajo un sistema de granularidad, como lo son, años, meses, días, horas, minutos, entre otros [22].

A través de la visualización de los datos espacio-temporales se logra obtener una mayor comprensión de los fenómenos complejos que dan intersección en el espacio y el tiempo. Lo que permite crear una trayectoria que se mueve a través de las tres dimensiones que representa esta información. Donde una de las formas en la que se puede representar esta visualización es a través de la técnica de “space-time cube”, el cual es un cubo cuya tercera dimensión representa el tiempo en el que pasa la trayectoria. Esto con la finalidad de comprender el movimiento o patrón de un fenómeno a través del tiempo.

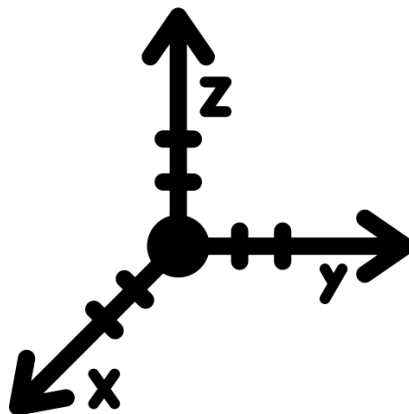


Fig. 1 Eje Espacio-Tiempo

C. Three.js

Three.js trata sobre una librería de JavaScript la cual permite la creación elementos o gráficos en 3D en tiempo real en el navegador. Esta realiza un trabajo más simple y sencillo utilizando WebGL, el cual es una API algo más compleja y difícil de utilizar que permite renderizar gráficos en 3D. De manera que reduce su complejidad y permite que los desarrolladores incorporen elementos 3D a sus aplicaciones web. Se compone de tres elementos que son la cámara para indicar los elementos que se muestran, el motor de renderizado que pinta los gráficos, y la escena la cual se encuentra todos los elementos. Además, esta librería permite la interacción de los usuarios con los modelos 3D a través del ratón o el teclado [12].

1) Características:

En el mundo actual centrado en la web, la presentación de información va más allá de simples imágenes, texto, y otros elementos bidimensionales. La demanda de experiencias visuales más inmersivas ha llegado a un auge de tecnología como WebGL, en la que three.js emerge como gran librería que permite la creación de gráficos 3D en tiempo real en aplicación web. Lo que permite proporcionar una plataforma sólida para la visualización de este los datos otorgados, permitiendo un trabajo más accesible a comparación de WebGL que presenta un trabajo más complicado. El uso de la librería three.js ha tenido una gran aceptación debido a que presenta características como:

- Facilita la creación de escenas tridimensionales en la que se pueden colocar objetos, cámaras y luces para construir entornos visuales.
- Simplifica el desarrollo de gráficos 3D en la web.
- Controla la visualización y la iluminación de las escenas.
- Representación dinámica de objetos y escenas.
- Permite a los usuarios interactuar con objetos 3D y escenas mediante eventos.
- Es adaptable a diferentes tamaños de pantallas y dispositivos.
- Carga modelos 3D en formatos comunes.
- Librería ligera y un equilibrio notable entre eficacia y capacidad.

- Ayuda a la creación de animaciones suaves y fluidas a través de la manipulación de objetos en la escena.
- Funciona en la mayoría de los navegadores web.
- Utiliza WebGL para la renderización, con la finalidad de obtener gráficos de alta calidad que se muestren dentro del navegador.

2) WebGL:

WebGL es una API (Application Programming Interface) que posibilita la generación de gráficos tridimensionales, con la función de operar interactivamente con el usuario a través de un navegador. Esta API se fundamenta a través de la tecnología OpenGL, siendo la base para la creación de los gráficos 2D y 3D en una variedad de plataformas y dispositivos. Utiliza el lenguaje de programación javascript para acceder al GPU y renderizar gráficos 3D en tiempo real, creando experiencias visuales avanzadas, sin la necesidad de plugins adicionales. Para eso trabaja con el elemento Canvas de HTML5 para mostrar los gráficos que se generan a través de la programación. De esta manera presenta un rendimiento visual optimizado para ofrecer experiencias dinámicas de los gráficos 3D que presenta la web [20].

a) Implementación:

Para desarrollar un proyecto se debe de importar la librería de Three.js a este, donde se puede descargar el archivo desde el sitio oficial e incorporarlo a través de una etiqueta. Otra opción es utilizar un enlace en la etiqueta de script del HTML para acceder a las funciones y clases proporcionadas por Three.js. También se puede descargar mediante el comando correspondiente para su incorporación dentro del proyecto. De esta manera se proporciona un conjunto complejo de herramientas y funcionalidades para desarrollar experiencias visuales de manera dinámica.

b) Configuración de la escena:

La escena actúa como el principal contenedor para todos los elementos 3D como lo son la cámara, luces y objetos. Es decir, donde se colocan y manipulan los objetos tridimensionales, determinando la visualización y la interacción dentro de este entorno. Para ello se debe de crear el renderizado, utilizando el WebGL, Canvas u otra opción disponible, donde se van a especificar las

dimensiones del lienzo tridimensional y agregarlos a la página. Esto se realiza adaptándose a los requisitos del proyecto.

c) Agregar objetos:

Se especifica la geometría que se puede implementar utilizando Three.js para dar forma al objeto, y se configura el material para mostrar cómo reacciona con la luz. Estos elementos se unen para formar un objeto tridimensional e incluirlo dentro de la escena que posteriormente será renderizada. Al crear ese objeto se le dará un tamaño y forma el cual por ejemplo puede ser un cubo, y en su textura se le definirá un color.

d) Agregar cámara:

Para definir la perspectiva y el punto de vista de una escena es necesario crear una cámara, para visualizar los elementos que se quieren mostrar. La cámara a su vez tiene la capacidad de actuar como cualquier otro elemento, es decir, puede realizar acciones como rotar, mover, entre otros. Es decir, que su configuración es esencial para definir la visualización y perspectiva de la escena en el espacio tridimensional.

Three.js permite utilizar diferentes tipos de cámara, cada uno con sus respectivas funcionalidades, no obstante, la más utilizada es el tipo perspectiva. Esto se debe a que proporciona un enfoque más realista debido a su ángulo de visión, y como esta se implementa con las dimensiones y el rango de visualización. De esta manera se hace que la cámara entre en la escena y enfoque algún objeto de esta.

Para crear la cámara de la escena primero se tiene que definir el campo de visión, ancho/alto y el rango de visión, luego se posiciona y orienta la cámara, para después crear los controles de la cámara que permitan el movimiento y rotación de los objetos. Esto se realiza para interacción de los objetos con el usuario, y finalmente se actualiza la cámara para reflejar cualquier cambio dentro de la escena.

e) Agregar luces:

Se necesita iluminación para poder visualizar los colores y formas de los objetos que se presentan dentro de la escena, ya que si no se aplican estos no se logran apreciar. Al aplicar estas luces se logra iluminar y observar cada uno de los objetos que se encuentran en la escena.

Estableciendo la claridad de visualización a través de un formato hexadecimal, además de la posición en la que se va a ubicar. De manera que la luz que se proyecta apunta a todas las direcciones y muestran los elementos de la escena.

f) Renderizar la escena:

El renderizado se encarga de mostrar la escena junto a todos sus elementos en el lienzo HTML para observar sus resultados. Esta misma permite realizar las animaciones personalizadas según las necesidades que se tenga, además de otras funcionalidades que otorga Three.js para la creación de modelos 3D. Esto indica su importancia para convertir la representación 3D de la escena en imágenes visuales que se muestran dentro del navegador. Para su configuración se utiliza WebGL para gestionar el proceso de renderizado en la escena, se define el tamaño del lienzo, para así añadirlo al documento HTML, y luego se renderiza la escena pasando esta y la cámara como sus argumentos y aplicarlo en el bucle de animación.

D. Tecnologías

1) Node.js:

Node.js es un entorno de ejecución de JavaScript, el cual trabaja en el lado del servidor y permite a los desarrolladores crear aplicaciones web escalables. Se caracteriza por estar orientado a eventos, lo que es muy eficiente para aplicaciones en tiempo real. Además, cuenta con el gestor de paquetes npm, que es la mayor colección de biblioteca de software de código abierto disponible [7].

2) React:

React es una librería de JavaScript utilizada con el fin de construir interfaces de usuario y aplicaciones web interactivas y reutilizables. Una de sus características es que además de poder trabajar como la vista de una aplicación, ayuda a complementar algunas de las funciones de esta. También, facilita la creación de aplicaciones complejas debido a su desarrollo en el Front-End y su combinación con otras librerías, implementando diseños de interfaces simples e interactivas para el usuario [15].

3) OrbitControls:

OrbitControls es un controlador el cual se encarga de la interacción entre el usuario y una escena 3D. Gracias a ello permite la navegación y manipulación de la cámara. Este controlador específicamente está diseñado para crear controles de órbita. Lo que le permite al usuario hacer zoom, rotar, y desplazarse alrededor de una escena 3D de manera intuitiva y fácil [8].

E. Herramientas

1) Visual Studio Code:

Visual Studio Code es un editor de código fuente desarrollado por Microsoft creado para varios sistemas operativos populares como Microsoft, Linux y MacOS. Es un entorno de desarrollo ligero y altamente personalizable. Incluye varias funciones como depuración, sangrías automáticas, finalización inteligente de código, entre otros aspectos. Se destaca por permitir agregar diferentes extensiones que facilitan el desarrollo de aplicaciones.

III. METODOLOGÍA

A. Diseño del proyecto

El uso de la técnica de space-time cube representa la evolución de los datos dentro de un área definida con respecto al tiempo, y su integración en una aplicación web para la visualización espacio-temporal es el enfoque principal en el que se basa este proyecto. Se demuestra una plataforma interactiva que permite demostrar las trayectorias de los datos en el espacio-tiempo. Con la finalidad de que cualquier usuario puede acceder a esta, visualizando de este tipo de datos por medio de un navegador web. Haciendo énfasis en la experiencia de usuario para garantizar la accesibilidad y la comprensión de la información presentada en el gráfico. Por ende, esto significa que el desarrollo de este proyecto se enfoca más al ámbito cualitativo debido a que se basa al análisis visual.

El objetivo de esto es demostrar de manera visual el trazo de las trayectorias de los datos espacio-temporales a través de la técnica de space-time cube. Mostrar la información de los datos que conforman cada uno de los puntos que trazan las rutas dentro de un mapa o zona geográfica a través del tiempo. Permitiendo mostrar la o las trayectorias que forman los puntos desde una perspectiva bidimensional (2D) y tridimensional (3D). Además de otras funcionalidades como la captura del escenario mostrando las trayectorias, mostrar la información de cada uno de los puntos, ocultar puntos y trayectorias, aumento y disminución de tamaño, pantalla completa, volver a perspectiva 2D, el filtrado de datos en horas y trayectorias.

Debido a esto se espera que la aplicación web muestre las trayectorias que se forman en las diferentes posiciones a través del tiempo. Esto con el objetivo de llevar un análisis de los datos que se ingresan dentro de la aplicación web de manera visual. Lo que lleva a mejores conclusiones, e indica un enfoque más cualitativo sobre la información y el tema que se está tratando.

Para la descripción y el proceso de desarrollo de este proyecto se toma en cuenta las metodologías de Design Thinking y Kanban respectivamente. Esto con el fin de abordar el problema o inconveniente que se busca resolver, e indicar la ejecución y seguimiento de las tareas específicas para el desarrollo del proyecto. Es decir, una sirve para entender las necesidades del problema que se busca resolver dentro del proyecto. El otro para facilitar la gestión de las tareas a través de tableros visuales que permitan un seguimiento de estas en tiempo real.

B. Estructura del proyecto

La implementación de este proyecto es un ejemplo de lo que es una investigación descriptiva. Debido a que se enfoca a describir las características de un fenómeno, que es este caso, da a conocer como pueden ser presentados de manera efectiva. Por ende, al aplicar este tipo de investigación se centra en proporcionar una representación detallada y precisa de la realidad. Proporcionado una visión clara y detallada de los datos espacio-temporales, para analizar la estructura de los conjuntos de datos. Presentado un tipo de visualización más efectiva e informativa para los usuarios.

Para el funcionamiento de este proyecto implicó definir la estructura y la forma en la que se van a obtener los datos espacio-temporales. El cual va a ser cargando un archivo JSON con un formato en específico, que va a tener la información de las coordenadas (x,y) y el tiempo. Donde los máximos valores del plano (x, y) van a ser de (-15, 15) y el campo del tiempo o valor en “z” con el formato en horas de “hh:mm:ss”. Además, de con campo externo o adicional que va a tener el enlace de la imagen del mapa que se quiere mostrar dentro del cubo para indicar por donde pasan las trayectorias.

En el desarrollo de la aplicación web se creó un cubo dentro de una escena, el cual es la base para la demostración de los datos espacio-temporales. Esta aplicación permite cargar los datos de las coordenadas (x, y, z), lo que permite a los usuarios visualizar los puntos y líneas que se crean para indicar una o más trayectorias, en caso de que sea más de una ruta. Obteniendo la información de cada uno de los puntos e interactuando con el cubo para que el usuario pueda observar desde diferentes perspectivas el trazo de las trayectorias que se forman a partir de estos datos.

Esto permite que la aplicación web sea una solución para una mejor visualización de este tipo de información, debido a que puede mostrar ambas perspectivas de las trayectorias de los datos-espaciotemporales en 2D y 3D. Convirtiéndolo en una herramienta accesible y fácil de usar para los usuarios, además de tener la posibilidad de descargar una captura de las trayectorias que se tiene dentro de la escena, junto con la forma que se quiere mostrar los datos, es decir aplicando las diferentes funcionalidades de la aplicación.

C. Design thinking

1) Empatizar:

En esta etapa del proceso de diseño, nos enfocamos en comprender a fondo las necesidades, desafíos y expectativas de los usuarios potenciales de la aplicación web para visualización de datos espacio-temporales en 2D y 3D. A partir de la investigación realizada, identificamos a dos grupos principales de usuarios: empresas que requieren visualizar datos de mercado, tendencias de consumo y patrones de comportamiento, así como empresas de transporte que desean visualizar sus rutas en relación al tiempo. Además, reconocemos que la aplicación está diseñada para servir a una variedad de industrias y no se centra en un nicho de mercado específico.

a) Necesidades y desafío:

- Las empresas necesitan comprender y analizar datos de mercado, tendencias de consumo y patrones de comportamiento para tomar decisiones estratégicas.
- Para optimizar sus operaciones, las empresas de transporte deben tener una visualización efectiva de rutas y tiempos.
- Aunque las representaciones 2D de datos espacio temporales son comunes en la práctica, la visualización 3D es un concepto mucho más útil que se puede estudiar.

b) Limitaciones y Consideraciones Técnicas:

- Actualmente, no existen aplicaciones web disponibles en el mercado para la visualización de datos espacio-temporales, lo que sugiere una necesidad no cubierta en el espacio tecnológico.
- Los datos se manejan principalmente en formato JSON, lo que implica una mayor facilidad de procesamiento y manipulación en comparación con otros formatos más complejos.
- La aplicación web está diseñada para ser accesible en términos de recursos de hardware. Por lo que no se requieren máquinas de alto rendimiento ni grandes cantidades de espacio de memoria como los necesarios para algunas soluciones de software, como Tableau.

c) Diversidad de usuarios y casos de Uso:

- La aplicación web está destinada a servir a múltiples empresas y sectores. Ya que, tiene la capacidad de representar datos de diversos casos de uso, adaptándose a las necesidades específicas de cada usuario.

- Se espera que la aplicación proporcione una experiencia intuitiva y eficiente a los usuarios.

2) *Definición:*

Se ha identificado un problema claro que es la falta de aplicaciones web dedicadas a la visualización de datos espacio-temporales. Los usuarios enfrentan dificultades para encontrar una plataforma accesible y efectiva que les permita visualizar datos en 2D y 3D de manera espacial y temporal.

Para abordar esta problemática, se propone el desarrollo de una aplicación web específicamente diseñada para la visualización de datos espacio-temporales en ambas dimensiones. Esta aplicación será versátil y adaptable, capaz de manejar una amplia gama de casos de uso y tipos de datos. La solución se fundamenta en el uso de un formato estándar de archivo JSON para la entrada de datos, lo que facilitará la integración y manipulación de los datos por parte de los usuarios.

A continuación, en la Tabla I y II se presentan los requerimientos funcionales y no funcionales que se tendrán en cuenta para el desarrollo de la aplicación web:

TABLA I

REQUERIMIENTOS FUNCIONALES

| Requerimientos funcionales | |
|-----------------------------------|--|
| Requerimiento | Descripción |
| Cargar Archivos JSON | La aplicación debe permitir a los usuarios cargar archivos JSON que contengan los datos espacio-temporales a ser visualizados. |
| Ocultar/Mostrar Puntos | Debe haber una opción para ocultar o mostrar los puntos de datos en el gráfico, lo que permite a los usuarios controlar la visibilidad de los puntos de datos en la visualización. |
| Ocultar/Mostrar Líneas | Debe haber una opción para ocultar o mostrar las líneas de conexión entre los puntos de datos, lo que permite a los usuarios controlar la visibilidad de las conexiones entre los datos en la visualización. |

TABLA I
(CONTINUACIÓN) REQUERIMIENTOS FUNCIONALES

| | |
|--|--|
| Ampliar la Escena del Gráfico | Los usuarios deben poder ampliar la escena del gráfico, lo que les permite acercarse para ver detalles específicos de la visualización. |
| Disminuir o Aumentar la Visualización | Debe haber opciones para disminuir o aumentar la visualización, lo que permite a los usuarios ajustar el tamaño de los elementos en el gráfico para adaptarse a sus necesidades. |
| Imprimir el Gráfico Generado | Se debe proporcionar una opción para que los usuarios impriman el gráfico generado, lo que les permite guardar la visualización en formato físico. |
| Filtro de Horas | Los usuarios deben poder aplicar un filtro de horas para limitar la visualización a un rango específico de tiempo, lo que les permite analizar datos en intervalos temporales específicos. |
| Mapa para el Gráfico | Se debe incluir un mapa para el gráfico, lo que permite a los usuarios tener una referencia visual de la ubicación geográfica de los datos representados en la visualización. |

TABLA II
REQUERIMIENTOS NO FUNCIONALES

| Requerimientos no funcionales | |
|--------------------------------------|---|
| Requerimiento | Descripción |
| Rendimiento | La aplicación debe ser rápida y eficiente en la carga y renderización de grandes conjuntos de datos espacio-temporales para garantizar una experiencia de usuario fluida. |
| Interfaz de Usuario Intuitiva | La interfaz de usuario debe ser fácil de usar y comprender, lo que permite a los usuarios navegar y utilizar las funcionalidades de la aplicación sin dificultad. |

TABLA II
(CONTINUACIÓN) REQUERIMIENTOS NO FUNCIONALES

| | |
|-------------------------------------|---|
| Compatibilidad del Navegador | La aplicación debe ser compatible con una amplia gama de navegadores web, incluyendo Chrome, Firefox, Safari, etc., para garantizar la accesibilidad para todos los usuarios. |
| Escalabilidad | La aplicación debe ser escalable y capaz de manejar un aumento en el número de usuarios y la cantidad de datos sin comprometer el rendimiento. |
| Mantenibilidad | El código de la aplicación debe estar bien estructurado y documentado, lo que facilita su mantenimiento y futuras actualizaciones por parte de los desarrolladores. |

3) Ideación:

Durante la fase de ideación, se buscó aprovechar al máximo la creatividad del equipo y explorar diversas perspectivas para el desarrollo de la aplicación web de visualización de datos espacio-temporales en 2D y 3D. El proceso se estructuró en una serie de pasos para garantizar una generación efectiva de ideas:

Se comenzó por asegurar que todos los miembros del equipo comprendieran completamente el problema y estuvieran al tanto de los hallazgos anteriores, incluyendo el análisis de requerimientos y el entendimiento profundo de las necesidades de los usuarios finales. Esto proporcionó una base sólida para la generación de ideas.

Durante la fase de generación de ideas, se emplearon diversas técnicas, como la lluvia de ideas y el mapa mental, para explorar diferentes enfoques y posibilidades. Se promovió la colaboración entre los miembros del equipo, permitiendo que cada uno contribuyera con sus perspectivas únicas y mejorará las ideas presentadas. Esto facilitó un proceso de refinamiento constante, donde las ideas iniciales se transformaron y se mejoraron en respuesta al feedback del equipo.

A continuación, se presentan las ideas principales que se han generado para la implementación en la aplicación web:

1. **Desarrollo de una interfaz interactiva en 2D y 3D:** Crear una interfaz intuitiva que permita a los usuarios explorar y analizar datos espacio-temporales de manera efectiva, utilizando tanto representaciones en 2D como en 3D para una visualización más inmersiva.
2. **Integración de filtrado:** Incluir funcionalidades avanzadas de filtrado y análisis que permitan a los usuarios personalizar su experiencia de visualización, facilitando la identificación de patrones y tendencias en los datos.
3. **Funcionalidades de exportación:** Incluir opciones para que los usuarios puedan exportar las visualizaciones creadas en diferentes formatos, como imágenes o archivos PDF, para su uso fuera de la plataforma.
4. **Información de uso:** Agregar una página o una sección donde se describa al usuario cómo manejar la aplicación web y el formato respectivo del archivo JSON con el que se trabaja.
5. **Implementación de opciones de personalización:** Ofrecer opciones de personalización que permitan a los usuarios adaptar la visualización a sus necesidades específicas, como los puntos y las líneas del gráfico.
6. **Facilidad de posicionamiento:** Implementar un botón que se encargue directamente de posicionar la visualización de datos desde una perspectiva de 2D.

4) Prototipo:

a) Interfaz Principal:

En la Figura 2 se puede visualizar la interfaz principal con la que se le da la bienvenida al usuario. Por un lado, en el rectángulo amarillo se ubica el nombre de la aplicación web que es TrackView. Luego en el rectángulo morado se ubican las secciones que tiene la página con información relevante. Mientras que, en el rectángulo anaranjado se ubica el contenido de la sección señalada por el usuario. En la sección de características se ubican las características de la aplicación web. En Formato se encuentran los formatos con los que se trabaja. En Uso la forma en la que el usuario debe de usar la aplicación. Se tiene además ejemplos y el equipo de trabajo. Por último, en la sección de la Gráfica el usuario podrá visualizar los datos espacio temporales.

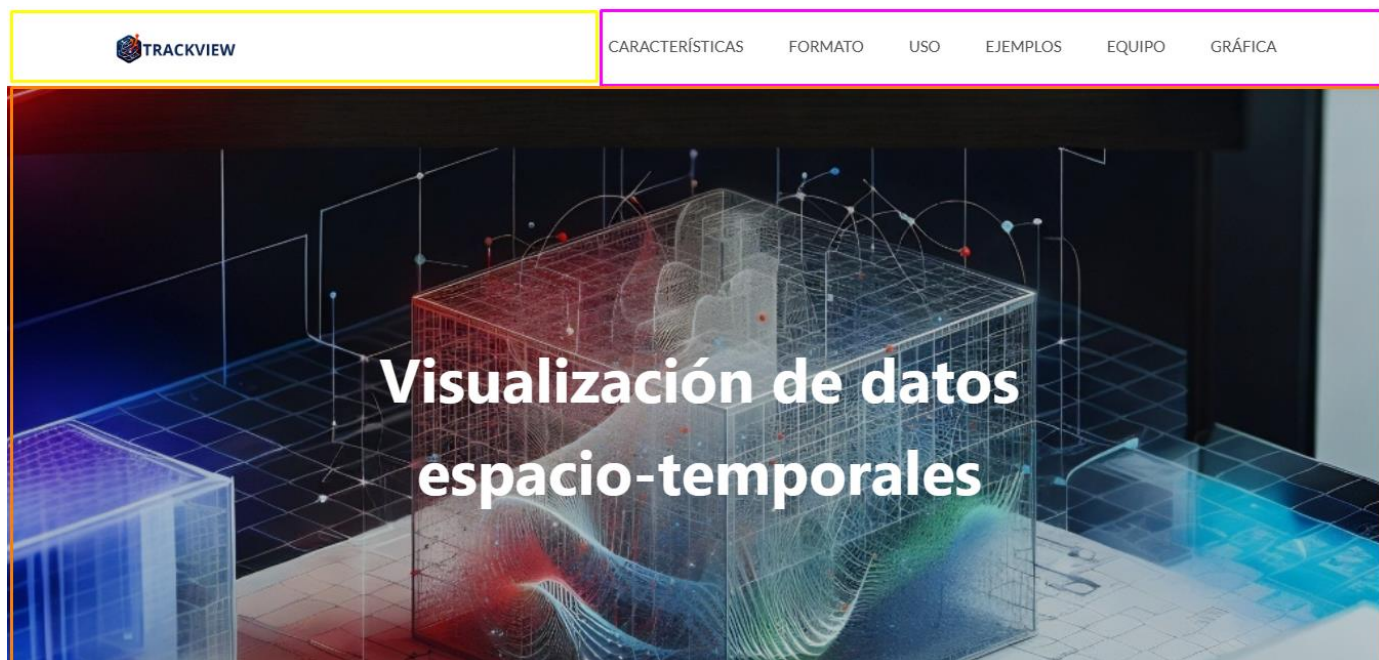


Fig. 2. Interfaz principal

b) Interfaz gráfica:

En la Figura 3 se presenta la interfaz que se encarga de visualizar los datos espacio-temporales. En el rectángulo rojo se ubican las opciones que tendrá el usuario para manejar la visualización de datos a su gusto. Y en la sección del rectángulo azul se encuentra el cubo donde se visualizarán los datos insertados por el usuario.

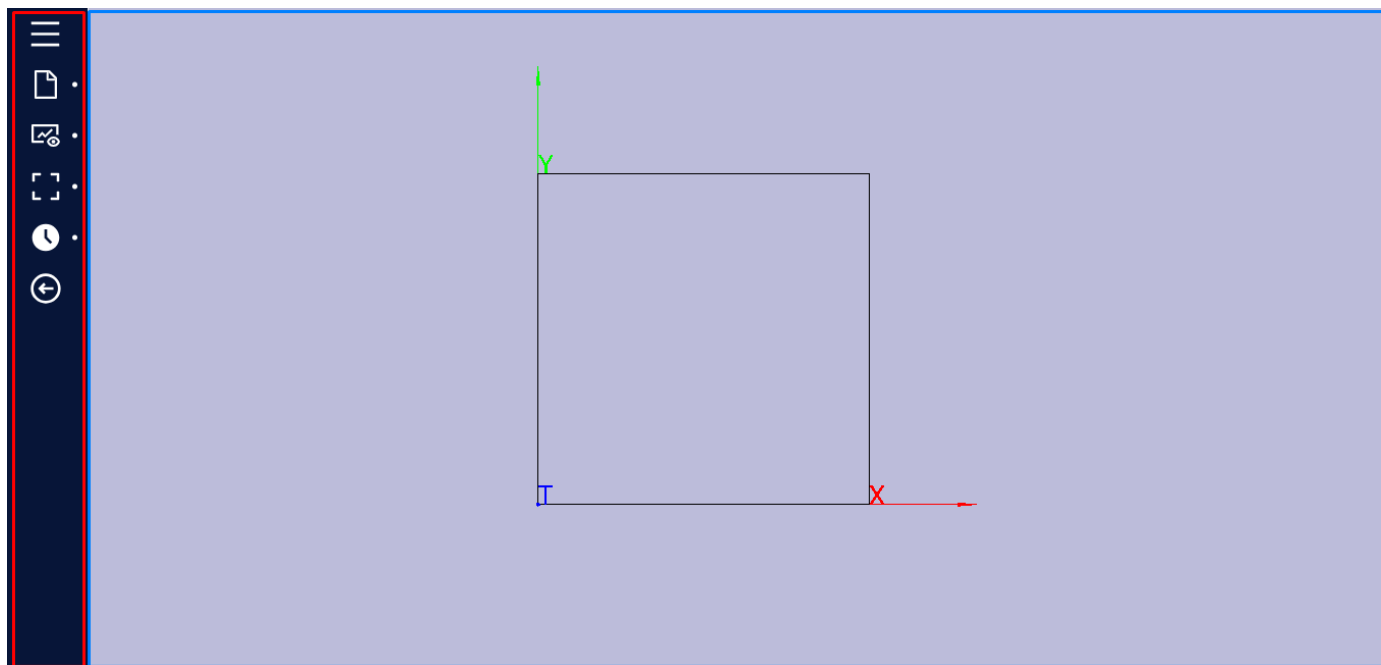


Fig. 3. Interfaz grafica

c) Opciones:

En la Figura 4 se encuentra el menú donde se manejan las opciones para la visualización de los datos. Donde el usuario se encargará de personalizar la visualización según sus necesidades. Entre las opciones podrá encontrar si desea visualizar o no los puntos, de igual forma las líneas. Y muchas opciones más que ayudarán a un mejor análisis de datos.

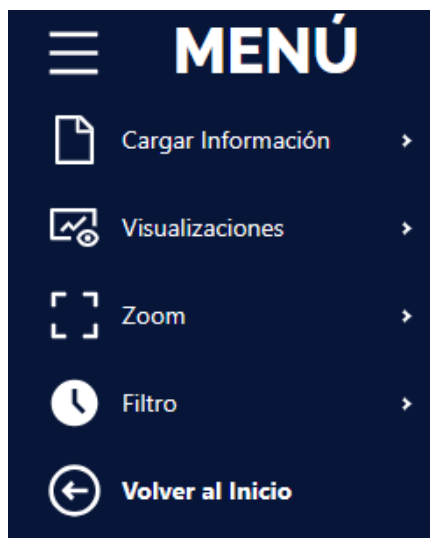


Fig. 4. Opciones

d) Visualización:

En la Figura 5 se encuentra el cubo donde se van a representar los datos espacio-temporales. Contiene un Axes Helper que servirá para identificar la altura (y) el ancho (x) y la profundidad (t) que sería el tiempo. Dicho Axes Helper también contiene sus letras correspondientes como se puede visualizar en la imagen. Lo cual ayudará al usuario a tener una mejor perspectiva al momento de mostrar los datos. Dicho cubo tiene la capacidad de tener desde 15 a -15 unidades en cada uno de los Axes Helper.

Al insertar los datos con el formato del archivo JSON correspondiente se debe de poder visualizarlos en un cubo espacio-tiempo. Así como se muestra en Figura 6, que sería la perspectiva 2D, y en la Figura 7 se encuentra la visualización de un lado que representa en forma de 3D.

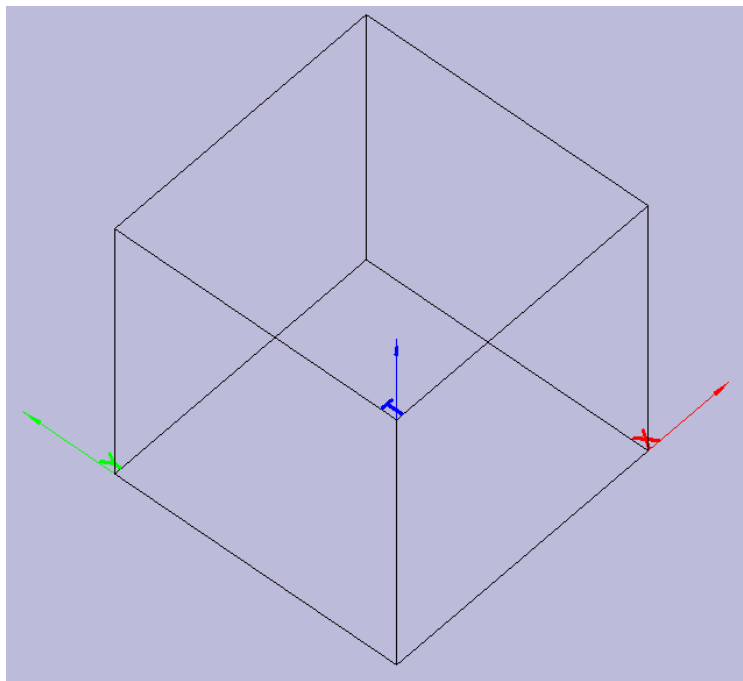


Fig. 5. Cubo base

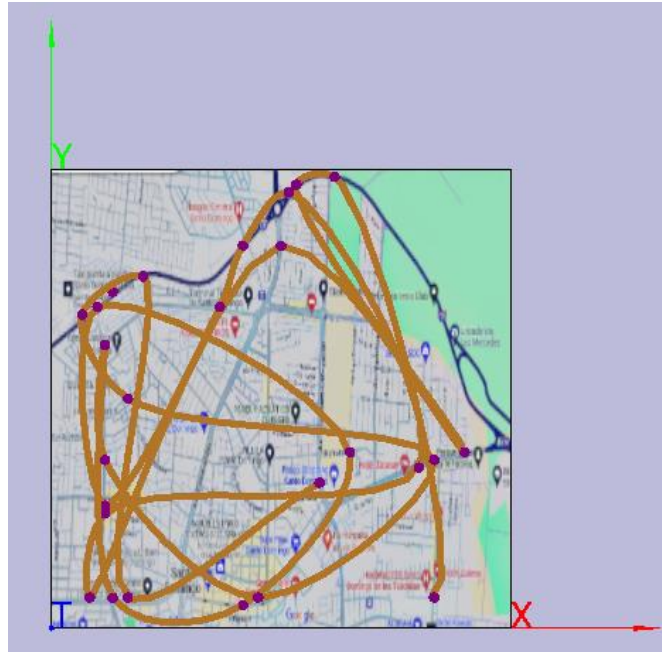


Fig. 6. visualización 2D

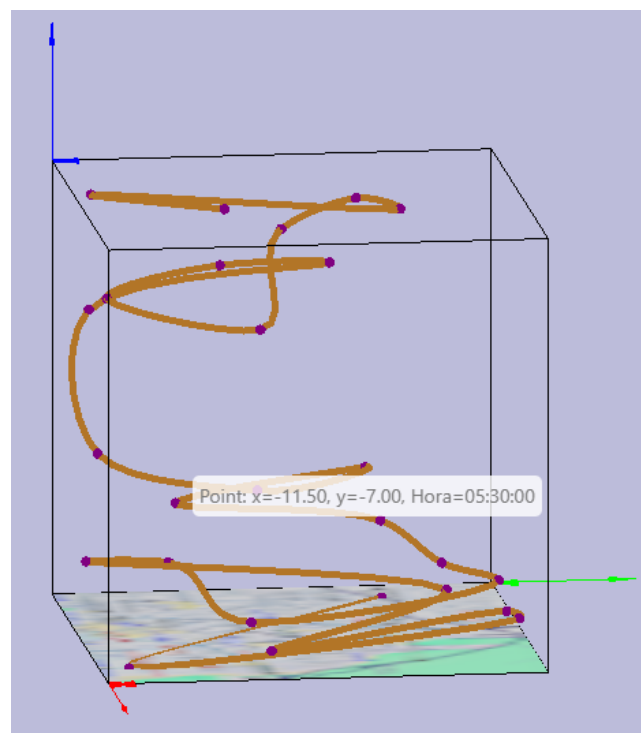


Fig. 7. Visualización 3D

5) Pruebas:

a) Prueba 1:

En la Figura 8 se realizará prueba de carga a la aplicación web, con la ayuda de la herramienta de Apache JMeter. Se realizar peticiones simulando un número grande de usuarios. Esto se realiza con el fin de asegurarse que la aplicación puede soportar y seguir funcionando correctamente.

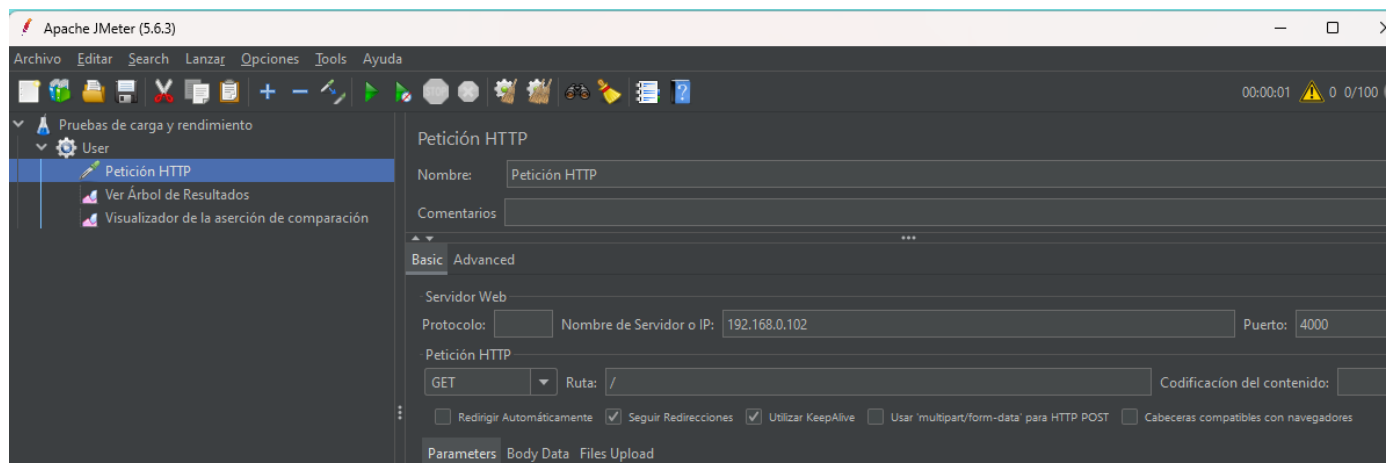


Fig. 8. Prueba de carga en JMeter

b) Prueba 2:

En la prueba 2 se tiene como objetivo probar la entrada de archivos JSON que va a permitir la aplicación web. Para ello se van a cargar diferentes archivos, buscando que la aplicación responda como se debe. En la Figura 9 se muestran los archivos con los que se harán las pruebas correspondientes.

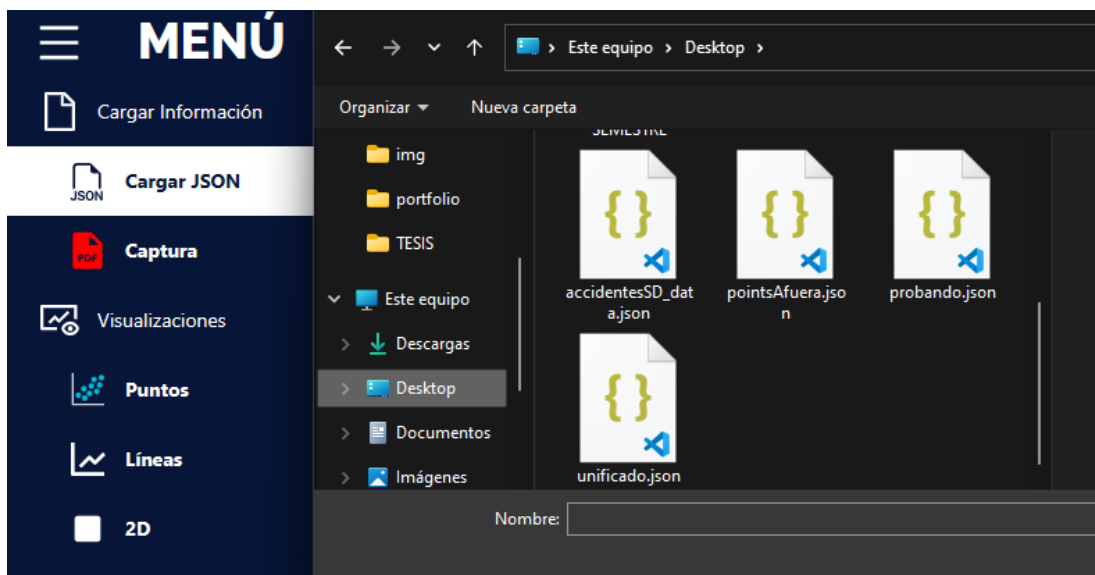


Fig. 9. Ingreso de archivos JSON

c) Prueba 3:

Se realizó un formulario de Google como se observa en la Figura 10, con preguntas sobre el funcionamiento de la aplicación web. Esto con el fin de que los usuarios que han usado la aplicación den su opinión sobre el funcionamiento del producto final.



Encuesta sobre el uso de la aplicación web Trackview

¡Gracias por utilizar nuestra aplicación de visualización de datos espacio-temporales Trackview!

Nos gustaría conocer tu opinión para asegurarnos de que nuestra plataforma cumpla con tus expectativas. Por favor, tómate unos minutos para completar esta breve encuesta.
Link de la aplicación web: [Trackview](#)

wjsolano@espe.edu.ec [Cambiar de cuenta](#)

No compartido

* Indica que la pregunta es obligatoria

Fig. 10. Formulario

d) Prueba 4:

La última prueba se encarga de evidenciar el funcionamiento de todas las opciones que contiene la aplicación web para la visualización de los datos espacio-temporales. En la Figura 11 se observan todas las opciones que se pondrán a prueba. Entre las que se encuentran la visualización, descarga, zoom y el filtro.

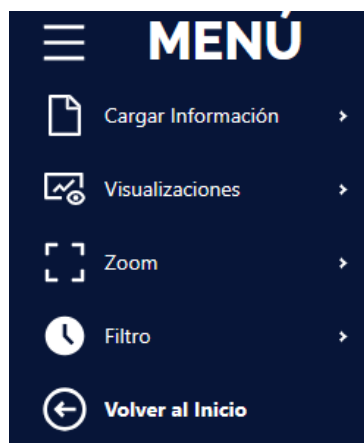


Fig. 11. Opciones disponibles

D. Kanban

1) Roles:

Para el desarrollo de la aplicación web de “Trackview” fueron asignados diferentes roles a los encargados de la implementación de este proyecto, que se muestra en la Tabla III. El Ingeniero Luis Ortiz es la persona que posee el rol de propietario de la aplicación, donde las personas encargadas de su desarrollo fueron los estudiantes Héctor Cedeño y Wilmer Solano. Hay que tomar en cuenta que estos últimos también son los responsables de realizar las pruebas de funcionamiento y de calidad de la aplicación. Además de que es importante señalar que todos desempeñan responsabilidades de manera efectiva, de acuerdo con lo establecido.

TABLA III

ROLES Y RESPONSABILIDADES

| Rol | Persona responsable | Tarea |
|--|----------------------------|---|
| Propietario | Ing. Luis Ortiz | Determina la funcionalidad de la aplicación, presentación y características. |
| Líder de Grupo | Luis Ortiz | Dirige e indica las actividades y cosas que se deben realizar para el desarrollo de la aplicación. |
| Desarrollador y Gestor | Hector Cedeño | Responsable de gran parte de la programación de la aplicación, además de gestionar y trabajar las tareas asignadas. |
| Tester | Héctor Cedeño | Encargado de realizar las pruebas de funcionamiento y calidad de la aplicación. Esto con el fin de identificar errores o problemas en el programa, e implementar soluciones. |
| Desarrollador y Analista de Pruebas | Wilmer Solano | Responsable de parte de la programación de la aplicación, además de realizar pruebas con ejemplos reales para asegurar que el software cumpla con las expectativas del usuario. |

TABLA III
(CONTINUACIÓN) ROLES Y RESPONSABILIDADES

| | | |
|------------------|------------------|--|
| Diseñador | Wilmer Solano | Se encarga de realizar la interfaz de presentación y funcionamiento de la aplicación, además de sus autores. |
|------------------|------------------|--|

Nota. Esta tabla indica los roles de cada uno de los integrantes del proyecto y que funciones cumplen cada uno de ellos dentro de este.

2) Tarjetas:

La creación de las tarjetas en Kanban permite la visualización y gestión del flujo de trabajo de las tareas de un proyecto. Cada una de estas tarjetas representan una unidad de trabajo individual, la cual contiene información sobre las tareas específicas que se debe de realizar. Lo que permite tener un mayor seguimiento de las tareas, teniendo una representación visual de los trabajos que deben ser completados. A continuación, desde la Figura 12 hasta la Figura 33 se presentará la forma en la que son aplicadas estas tarjetas en las diferentes partes en la que se divide el proyecto:

| | |
|------------------|---|
| ID | Tarea |
| KAN-1 | Crear el proyecto en Visual Studio Code |
| Prioridad | Descripción |
| Alta | Identificar el tipo de librería que se debe de utilizar para mostrar datos espacio temporales, que este caso va a ser la librería de Three.js, usando la tecnología de React. |
| Fechas | 13/10/2023 – 14/10/2023 |

Fig. 12. Tarjeta de creación del proyecto en Visual Studio Code

| | |
|------------------|---|
| ID | Tarea |
| KAN-4 | Verificar el tipo de JSON que se debe de manejar |
| Prioridad | Descripción |
| Alta | Verificar los parámetros y el formato en que debe estar el archivo JSON que va a contener la información de los puntos, para visualizar los datos espacio-temporales. |
| Fechas | 13/10/2023 – 17/10/2023 |

Fig. 13. Verificar el tipo de JSON que se debe de manejar

| | |
|------------------|---|
| ID | Tarea |
| KAN-5 | Crear la escena, cámara, luces y el cubo de la aplicación |
| Prioridad | Descripción |
| Alta | Implementar los elementos principales para representar los datos espacio-temporales |
| Fechas | 13/10/2023 – 12/11/2023 |

Fig. 14. Crear la escena, cámara, luces y el cubo de la aplicación

| | |
|------------------|--|
| ID | Tarea |
| KAN-9 | Colocar una imagen de un mapa dentro del cubo |
| Prioridad | Descripción |
| Alta | Dentro del archivo JSON debe haber un campo con el enlace de la imagen, y al momento de cargar el JSON se debe colocar la imagen en el plano inferior. |
| Fechas | 12/11/2023 – 12/01/2024 |

Fig. 15. Colocar una imagen de un mapa dentro del cubo

| | |
|------------------|---|
| ID | Tarea |
| KAN-14 | Obtener los datos del archivo JSON a través de un botón |
| Prioridad | Descripción |
| Alta | Crear un botón que se encargue de cargar la información del archivo JSON para poder visualizarla en el cubo |
| Fechas | 02/01/2024 – 12/01/2024 |

Fig. 16. Obtener los datos del archivo JSON a través de un botón

| | |
|------------------|---|
| ID | Tarea |
| KAN-15 | Crear botones para ocultar líneas y puntos |
| Prioridad | Descripción |
| Alta | Agregar dos botones, uno para ocultar y volver a mostrar los puntos que se visualizan en el cubo, y otro en cambio para las líneas que se crean |
| Fechas | 02/01/2024 – 12/01/2024 |

Fig. 17. Crear botones para ocultar líneas y puntos

| | |
|------------------|--|
| ID | Tarea |
| KAN-16 | Mostrar la información de los puntos en un mensaje flotante |
| Prioridad | Descripción |
| Muy Alta | Al momento de que el mouse pase por encima del botón se debe visualizar la información de los puntos (x, y) y el tiempo. |
| Fechas | 02/01/2024 – 12/01/2024 |

Fig. 18. Mostrar la información de los puntos en un mensaje flotante

| | |
|------------------|--|
| ID | Tarea |
| KAN-17 | Estandarizar los datos en el tiempo para implementarla dentro del cubo |
| Prioridad | Descripción |
| Muy Alta | Hacer que la hora mínima se coloque en el punto mas bajo del cubo, y que la hora máxima se coloque en el punto más alto del cubo, y los demás puntos se vayan colocando según corresponda el rango |
| Fechas | 02/01/2024 – 26/01/2024 |

Fig. 19. Estandarizar los datos en el tiempo para implementarla dentro del cubo

| | |
|------------------|---|
| ID | Tarea |
| KAN-18 | Colocar el valor de z en horas en la etiqueta que muestra los puntos |
| Prioridad | Descripción |
| Muy Alta | Arreglar la programación de la aplicación para que el valor del tiempo ya no se muestre con datos numéricos sino como el formato hh:mm:ss, con el fin de visualizar la hora |
| Fechas | 02/01/2024 – 15/01/2024 |

Fig. 20. Colocar el valor de z en horas en la etiqueta que muestra los puntos

| | |
|------------------|---|
| ID | Tarea |
| KAN-19 | Tener la opción de pantalla completa |
| Prioridad | Descripción |
| Media | Crear un botón que tenga la función de colocar la escena en pantalla completa, para presenta una mejor visualización. |
| Fechas | 02/01/2024 – 15/01/2024 |

Fig. 21. Tener la opción de pantalla completa

| | |
|------------------|---|
| ID | Tarea |
| KAN-20 | Mejorar la presentación de las líneas |
| Prioridad | Descripción |
| Media | Hacer que las líneas estén compuestas por diferentes tipos de colores cuando se vaya trazando el camino al unir los puntos. |
| Fechas | 02/01/2024 – 15/01/2024 |

Fig. 22. Mejorar la presentación de las líneas

| | |
|------------------|---|
| ID | Tarea |
| KAN-23 | Hacer que la información de los puntos se vea cuando no se está en pantalla completa |
| Prioridad | Descripción |
| Muy Alta | Arreglar el inconveniente de que cuando la escena que contiene al cubo no se encuentre en pantalla completa no muestre los puntos, haciendo que estos se visualizasen en ambos estados. |
| Fechas | 15/01/2024 – 25/01/2024 |

Fig. 23. Hacer que la información de los puntos se vea cuando no se está en pantalla completa

| | |
|------------------|---|
| ID | Tarea |
| KAN-24 | Hacer los puntos redondos |
| Prioridad | Descripción |
| Alta | Cambiar la figura en la que se representa los puntos que se grafican en el cubo, pasándolos de cuadrados a redondos |
| Fechas | 15/01/2024 – 25/01/2024 |

Fig. 24. Hacer los puntos redondos

| ID | Tarea |
|-----------|--|
| KAN-25 | Hacer las opciones para el zoom y para el gráfico 2D |
| Prioridad | Descripción |
| Media | Crear dos botones para implementar el zoom en el cubo, uno será para aumentarlo y el otro para disminuirlo, además de otro botón para restablecer la posición al estado 2D |
| Fechas | 25/01/2024 – 26/01/2024 |

Fig. 25. Hacer las opciones para el zoom y para el gráfico 2D

| ID | Tarea |
|-----------|--|
| KAN-28 | Dar la opción de descargar reporte |
| Prioridad | Descripción |
| Alta | Crear un botón que permita descargar una captura de la escena en el estado que se encuentra en un PDF. |
| Fechas | 25/01/2024 – 27/01/2024 |

Fig. 26. Dar la opción de descargar reporte

| ID | Tarea |
|-----------|---|
| KAN-31 | Letras/Cubo cara de arriba transparente, y Axes Helper |
| Prioridad | Descripción |
| Media | Implementar la función Axes Helper para indicar en que dirección se encuentra cada eje (x, y, t). Colocando letras para identificarlas, además de hacer la cara del cubo transparente para una mejor visualización en el plano 2D |
| Fechas | 29/01/2024 – 06/02/2024 |

Fig. 27. Letras/Cubo cara de arriba transparente, y Axes Helper

| | |
|------------------|--|
| ID | Tarea |
| KAN-30 | Hacer un filtro de hora |
| Prioridad | Descripción |
| Muy Alta | Desarrollar un filtro que permita solo visualizar los puntos desde el rango de horas que se especifica |
| Fechas | 29/01/2024 – 11/02/2024 |

Fig. 28. Hacer un filtro de hora

| | |
|------------------|---|
| ID | Tarea |
| KAN-26 | Mejorar el estilo de la presentación de la aplicación en la parte gráfica |
| Prioridad | Descripción |
| Alta | Dividir en secciones cada uno de los componentes funcionales de la aplicación web para saber identificar su funcionalidad |
| Fechas | 25/01/2024 – 11/02/2024 |

Fig. 29. Mejorar el estilo de la presentación de la aplicación en la parte gráfica

| | |
|------------------|---|
| ID | Tarea |
| KAN-21 | Hacer una página de presentación de la aplicación y de los autores |
| Prioridad | Descripción |
| Muy Alta | Presentar una página que de a conocer sobre el funcionamiento de la aplicación web, como funciona, sus pasos, el formato JSON que recibe, y como se visualizan los datos. Además de la presentación de los desarrolladores de la aplicación |
| Fechas | 12/01/2024 – 11/02/2024 |

Fig. 30. Hacer una página de presentación de la aplicación y de los autores

| | |
|------------------|---|
| ID | Tarea |
| KAN-22 | Realizar filtro de trayectorias |
| Prioridad | Descripción |
| Muy Alta | Desarrollar un filtro que permita solo visualizar las trayectorias que se indican mediante un <u>checkbox</u> |
| Fechas | 25/01/2024 – 17/02/2024 |

Fig. 31. Hacer filtro trayectorias

| | |
|------------------|--|
| ID | Tarea |
| KAN-27 | Realizar pruebas de funcionalidad, y probar con datas reales |
| Prioridad | Descripción |
| Muy Alta | Probar el funcionamiento de la aplicación con diferentes datas de la vida real, además de probar todas sus excepciones, con el fin de hallar errores y solucionarlos, haciendo que algunos usuarios lo prueben |
| Fechas | 25/01/2024 – 18/02/2024 |

Fig. 32. Realizar pruebas de funcionalidad, y probar con datas reales

| | |
|------------------|---|
| ID | Tarea |
| KAN-36 | Entrega del producto |
| Prioridad | Descripción |
| Alta | Entrega de la aplicación web, la cual va a visualizar los datos espacio-temporales cargados desde un archivo JSON. Donde se observan las trayectorias que se forman a unir los puntos y poder visualizar la información que contienen. Además de las opciones de filtrado, descarga de reporte, zoom, pantalla completa, ocultar líneas y puntos. |
| Fechas | 06/02/2024 – 01/03/2024 |

Fig. 33. Entrega del producto

3) Flujo de trabajo:

Para el desarrollo de la aplicación web para la visualización de datos espacio-temporales, el flujo de trabajo se divide en diferentes etapas para identificar en qué estado se encuentra cada una de las etapas. En la Figura 34 se puede observar la manera en la que se encuentra establecida cada una de las etapas de este flujo.

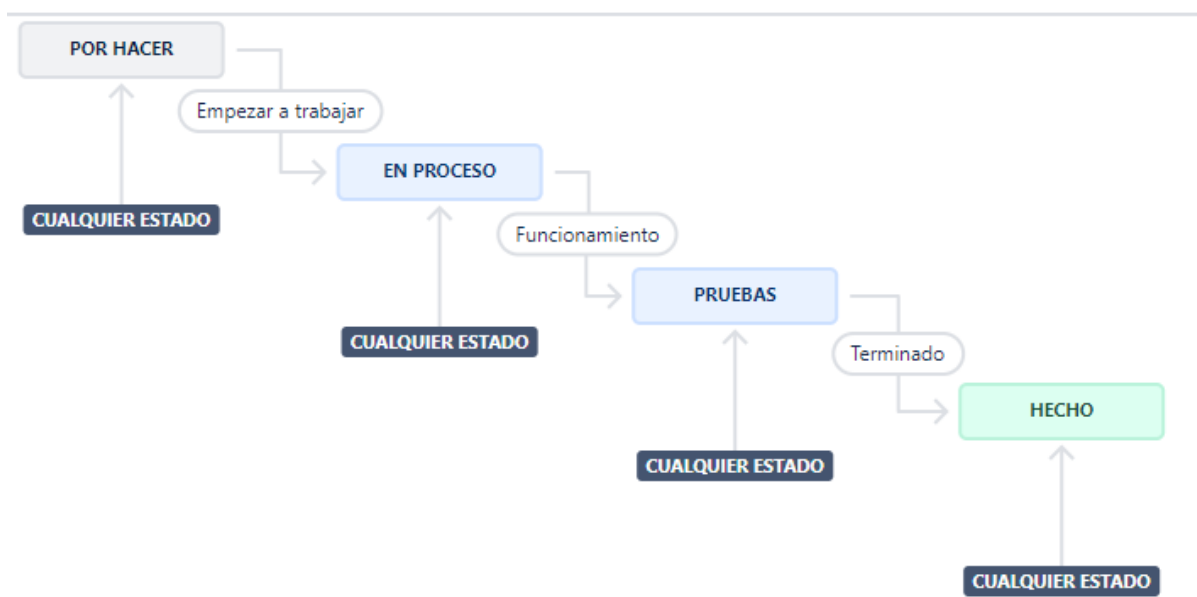


Fig. 34. Flujo de trabajo

4) Backlog:

El backlog se refiere a la lista de tareas que se encuentran para ser definidas en el desarrollo del proyecto. Estos elementos se encuentran ordenados por prioridad, de manera que lo más importante se coloque en la parte superior de la lista de tareas. Esto se puede observar en la Figura 35 donde se utilizó la herramienta de Jira Software para la creación de las actividades.

Backlog

| Task ID | Description | Status | Assignee |
|----------|--|------------------|----------|
| ✓ KAN-17 | Estandarizar los datos en el tiempo para implementarla dentro del cubo | FINALIZADA | HC |
| ✓ KAN-23 | Hacer que la información de los puntos se vea cuando no se esta en pantalla completa | FINALIZADA | HC |
| ✓ KAN-21 | Hacer una página de presentación de la aplicación y de los autores | EN CURSO | WL |
| ✓ KAN-27 | Realizar pruebas de funcionalidad, y probar con datas reales | PRUEBAS | WL |
| ✓ KAN-30 | Hacer un filtro de hora | PRUEBAS | WL |
| ✓ KAN-25 | Hacer las opciones para el zoom y para el grafico 2D | FINALIZADA | HC |
| ✓ KAN-24 | Hacer los puntos redondos | FINALIZADA | HC |
| ✓ KAN-28 | Dar la opción de imprimir reporte | FINALIZADA | WL |
| ✓ KAN-26 | Mejorar el estilo de la presentación de la aplicación en la parte gráfica | EN CURSO | HC |
| ✓ KAN-36 | Entrega del producto | TAREAS POR HACER | HC |
| ✓ KAN-31 | Letras/Cubo cara de arriba transparente, y Axes Helper | FINALIZADA | WL |

Fig. 35. Backlog

5) *Criterios de aceptación:*

TABLA IV

CRITERIOS DE ACEPTACIÓN

| Criterio | Evento | Resultado Esperado |
|--|------------------------|--|
| Verificar el tipo de formato que se ingresa | Cargar el archivo JSON | Advertencia sobre el inconveniente |
| Verificar el tipo de datos que se ingresa | Cargar el archivo JSON | Advertencia sobre el inconveniente |
| Verificar el rango especificado en cada uno de los campos | Cargar el archivo JSON | Advertencia sobre el inconveniente |
| Demostrar los puntos y líneas | Cargar el archivo JSON | Visualización de los puntos y las líneas que trazan las trayectorias |

TABLA IV
(CONTINUACIÓN) CRITERIOS DE ACEPTACIÓN

| | | |
|---|--|--|
| Ocultar y mostrar los puntos | Presionar el botón de Puntos o Líneas | Ocultan los puntos del cubo, y si se lo presiona de nuevo los muestra de nuevo, lo mismo para las líneas |
| Ocultar y mostrar las líneas | Presionar el botón de Líneas | |
| Tener la función de pantalla completa | Presionar botón de Fullscreen | Muestra en pantalla completa la escena con todos sus elementos |
| Hacer zoom al cubo | Presionar el botón de Zoom de “+” o “-” | Aumenta o disminuye el tamaño del cubo |
| Mostrar una imagen en el plano inferior del cubo | Cargar el archivo JSON con un campo que tenga el enlace de la imagen | Visualiza el mapa en la parte inferior del cubo con sus respectivos puntos y trayectorias |
| Visualizar la información de los puntos | Pasar el cursor por encima de los puntos | Visualiza la información del punto que se pasa por encima, como lo son las coordenadas (x,y) y la hora. |
| Restablecer la posición a la perspectiva 2D | Presionar el botón de 2D | Coloca el cubo en la perspectiva 2D en caso de que se encuentre en otra posición. |
| Descargar reporte | Presionar el botón de Descargar Reporte | Descarga un PDF con una captura de la escena, el cubo y los demás elementos que la compone, además de la información de esta |
| Filtro de hora | Se selecciona un rango de horas | Muestra los puntos y la trayectoria que se forman en el rango establecido por el filtro |

TABLA IV

(CONTINUACIÓN) CRITERIOS DE ACEPTACIÓN

| | | |
|--|---|---|
| Visualizar más de una trayectoria | Cargar un JSON con más de una trayectoria | Visualiza los caminos creados por los puntos de diferentes trayectorias |
|--|---|---|

Nota. En esta tabla se especifican los criterios de aceptación del funcionamiento de la aplicación web.

6) Tablero:

El tablero de Kanban es una herramienta visual que otorga una representación gráfica del flujo de trabajo del proyecto. Por ende, este representa las diferentes fases en la que se divide el desarrollo de la aplicación web para la visualización de datos espacio-temporales. Estas fases se organizan a través de columnas que representan cada una de las etapas del flujo de trabajo en la que se conforma un conjunto de tareas específicas. Las cuales se deben llevar a cabo para cumplimiento del proyecto.

- **Por Hacer (To Do):** Esta etapa trata sobre los trabajos que se han creado, pero no se han comenzado a realizar. Es decir, es la fase inicial del proyecto donde se colocan todas las tareas recién creadas o que todavía están pendientes. Son actividades que se encuentran en la lista de espera para ser tomadas y emplearlas en el proyecto.
- **En Curso (In Progress):** Es la fase donde las tareas son tomadas para ser realizar e implementarlas su funcionalidad dentro del proyecto. Con el fin de mostrar al equipo de trabajo en que tareas se encuentran enfocados.
- **Pruebas (Testing):** Cuando las tareas han sido completadas se realiza la respectiva evaluación de su funcionalidad o cualquier tipo de verificación necesaria. Esto con la finalidad de que su implementación funcione sin problemas antes de ser considerar la tarea por terminada.
- **Listo (Done):** Indica que una vez que se haya comprobado su funcionalidad con éxito y sin inconvenientes cada una de las pruebas se considera la tarea como terminada. Es decir, contiene las tareas que ya han sido completadas y han cumplido con los criterios de aceptación y prueba. Con esto se indica que esa tarea ya puede ser entregada dentro de la aplicación.

Para el desarrollo de esta aplicación web se dio uso de la herramienta de Jira Software para la creación de del tablero que se puede observar en la Figura 36. Donde se indica cada una de las fases en las que se encuentran las distintas tareas que se desarrollan dentro del proyecto.

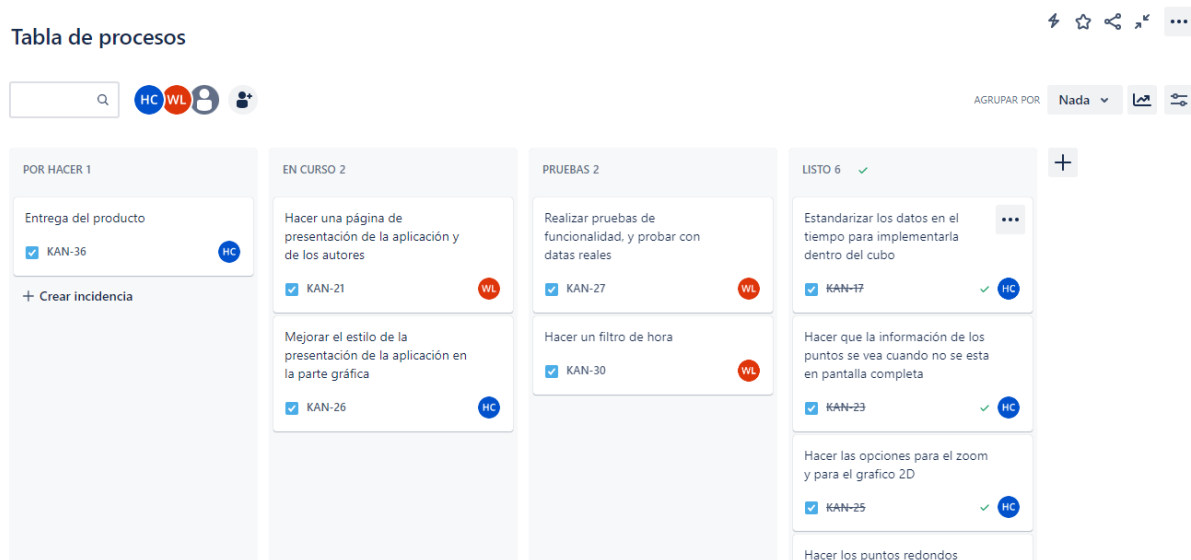


Fig. 36. Tablero Kanban

IV. RESULTADOS

A. Análisis de resultados

A continuación, se presentan los resultados de las pruebas que se han realizado para la aplicación web desarrollada. A la vez también se analiza el producto final que se ha obtenido.

En la Figura 37 se muestran los resultados obtenidos de la primera prueba que trata sobre la carga de trabajo. En este caso se simula 100 peticiones a la aplicación web, obteniendo un resultado positivo, asegurando que responde correctamente al número de peticiones asignadas. De igual manera, se analiza que la latencia y el tiempo de conexión se encuentra en un rango aceptable, el cual no va a dificultar cuando el usuario acceda a la aplicación.

Ver Resultados en Árbol

Nombre: Ver Resultados en Árbol

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Navegar... Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

| Muestra # | Tiempo de co... | Nombre del hilo | Etiqueta | Tiempo de Mu... | Estado | Bytes | Sent Bytes | Latency | Connect Time... |
|-----------|-----------------|-----------------|---------------|-----------------|--------|-------|------------|---------|-----------------|
| 1 | 00:55:37.241 | User 1-1 | Petición HTTP | 35 | ✓ | 1819 | 116 | 35 | 2 |
| 2 | 00:55:37.262 | User 1-3 | Petición HTTP | 16 | ✓ | 1819 | 116 | 16 | 1 |
| 3 | 00:55:37.271 | User 1-4 | Petición HTTP | 13 | ✓ | 1819 | 116 | 13 | 1 |
| 4 | 00:55:37.260 | User 1-2 | Petición HTTP | 26 | ✓ | 1819 | 116 | 26 | 0 |
| 5 | 00:55:37.282 | User 1-5 | Petición HTTP | 8 | ✓ | 1819 | 116 | 8 | 1 |
| 6 | 00:55:37.292 | User 1-6 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 7 | 00:55:37.303 | User 1-7 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 8 | 00:55:37.313 | User 1-8 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 9 | 00:55:37.322 | User 1-9 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 10 | 00:55:37.332 | User 1-10 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 11 | 00:55:37.344 | User 1-11 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 12 | 00:55:37.352 | User 1-12 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 0 |
| 13 | 00:55:37.361 | User 1-13 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |
| 14 | 00:55:37.374 | User 1-14 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |
| 15 | 00:55:37.382 | User 1-15 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |
| 16 | 00:55:37.392 | User 1-16 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |
| 17 | 00:55:37.403 | User 1-17 | Petición HTTP | 6 | ✓ | 1819 | 116 | 6 | 1 |
| 18 | 00:55:37.412 | User 1-18 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |
| 19 | 00:55:37.421 | User 1-19 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |
| 20 | 00:55:37.431 | User 1-20 | Petición HTTP | 5 | ✓ | 1819 | 116 | 5 | 1 |

Fig. 37. Resultados de prueba de carga

En la segunda prueba donde se insertan varios archivos JSON primero se ha probado cuando un archivo contiene datos fuera del rango especificado. En la Figura 38 se muestra la información del primer archivo que se usa, el cual tiene como un valor de X 25. Por ello, la

aplicación web muestra el mensaje de la Figura 39, indicando que algún valor se encuentra fuera de rango.

```
{
  "paths": [
    {
      "points": [
        {
          "x": 25,
          "y": 7,
          "z": "00:30:00"
        },
        {
          "x": 10,
          "y": -13,
          "z": "00:30:00"
        },
        {
          "x": 3.5,
          "y": 14.5,
          "z": "01:30:00"
        }
      ],
      "x": 1,
      "y": 14
    }
  ]
}
```

Fig. 38. Archivo 1



Fig. 39. Mensaje fuera de rango

Otra de las validaciones que se encuentran es cuando al ingresar la hora esta se encuentra fuera del rango de las 24 horas, tal y como se muestra en el ejemplo de la Figura 40, botando la advertencia de la Figura 41.

```
{
  "paths": [
    {
      "points": [
        {
          "x": -5,
          "y": 2,
          "z": "24:30:00"
        },
        {
          "x": -5,
          "y": 0,
          "z": "04:30:00"
        },
        {
          "x": 4,
          "y": -9,
          "z": "12:30:00"
        }
      ]
    }
  ]
}
```

Fig. 40. Archivo 2

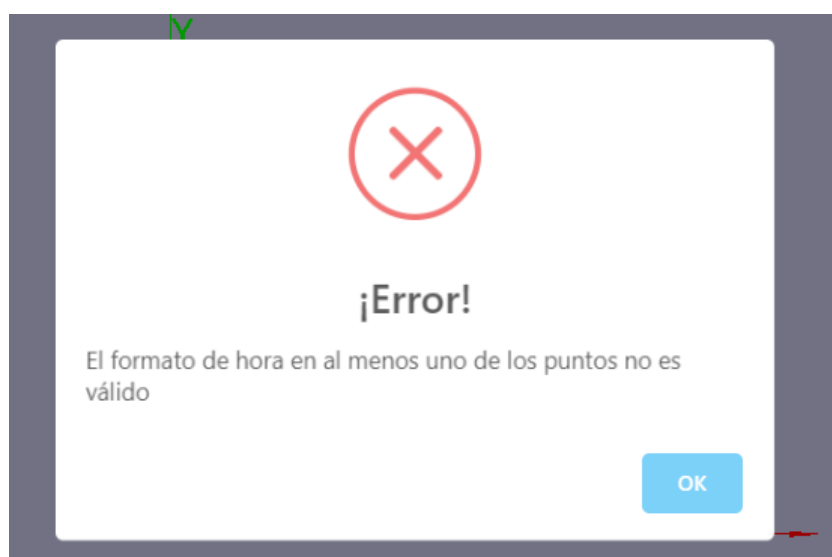


Fig. 41. Mensaje de hora inválida

En el caso de que la hora sea otro tipo de datos, por ejemplo, numérico como se muestra en la Figura 42 muestra el mensaje de la Figura 43.


```
{
  "paths": [
    {
      "points": [
        {
          "x": -5,
          "y": -3,
          "z": 10
        },
        {
          "x": -5,
          "y": 0,
          "z": "04:30:00"
        },
        {
          "x": 4,
          "y": -9,
          "z": "12:30:00"
        },
        {
          "x": 0,
          "y": 5,
          "z": "04:30:00"
        }
      ]
    }
  ]
}
```

Fig. 42. Archivo 3

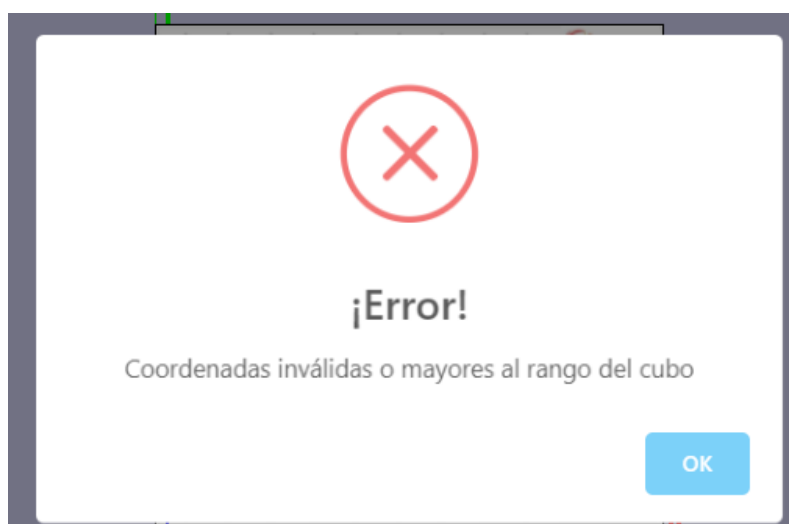


Fig. 43. Mensaje de formato de hora inválida

Y en el caso de que se quiera subir otro tipo de formato como se muestra en la Figura 44 mostrará el mensaje de la Figura 45. Toca recalcar que en cada vez que se muestre esta advertencia no mostrará ni puntos ni líneas en el cubo.

```
{  
  "type": "scatter3d",  
  "mode": "lines+markers",  
  "x": [40.7128, 41.8781, 34.0522, 37.7749, 33.6846, 39.7392],  
  "y": [-74.0060, -87.6298, -118.2437, -122.4194, -117.8265, -104.9903],  
  "z": [  
    "2023-10-01T10:00:00",  
    "2023-10-01T12:00:00",  
    "2023-10-02T09:00:00",  
    "2023-10-02T12:00:00",  
    "2023-10-03T08:00:00",  
    "2023-10-03T11:00:00"  
  ],  
}
```

Fig. 44. Archivo 4

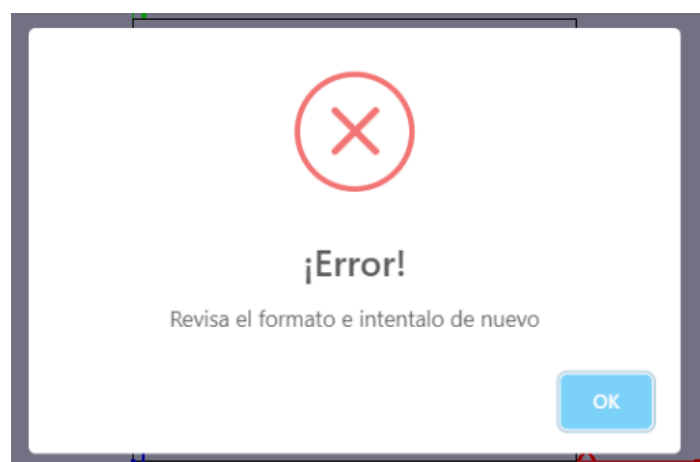


Fig. 45. Mensaje de formato incorrecto

Por parte del formulario que se ha realizado, a continuación, se presentan las respuestas de las preguntas más relevantes. Desde la Figura 46-48 se indica que tan conforme se encontraba el usuario con la información que se encontraba en la aplicación web. En el grafico se puede analizar que la única sección de información que recibió una valoración de regular fue la guía de uso.

Indique el nivel de acuerdo a sus afirmaciones sobre la página web informativa en los siguientes puntos:

[Copiar](#)

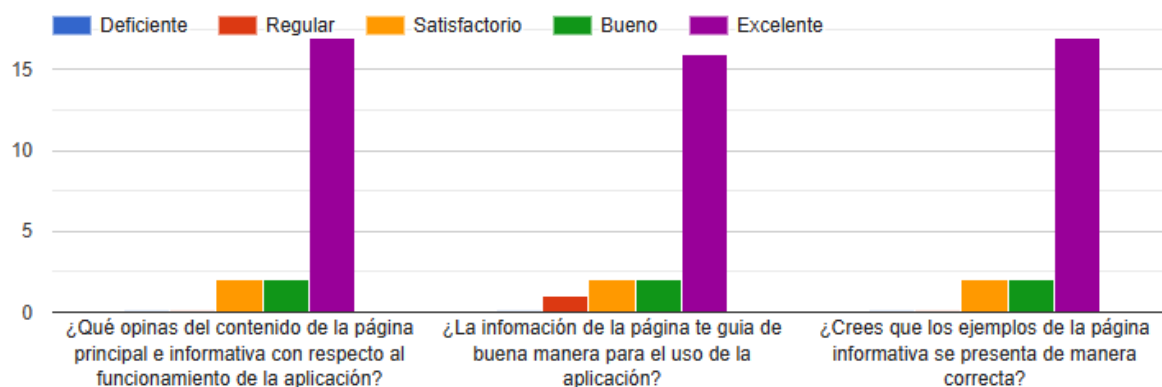


Fig. 46. Formulario resultados información

Indique el nivel de acuerdo a sus afirmaciones sobre la página web informativa en los siguientes puntos:

[Copiar](#)

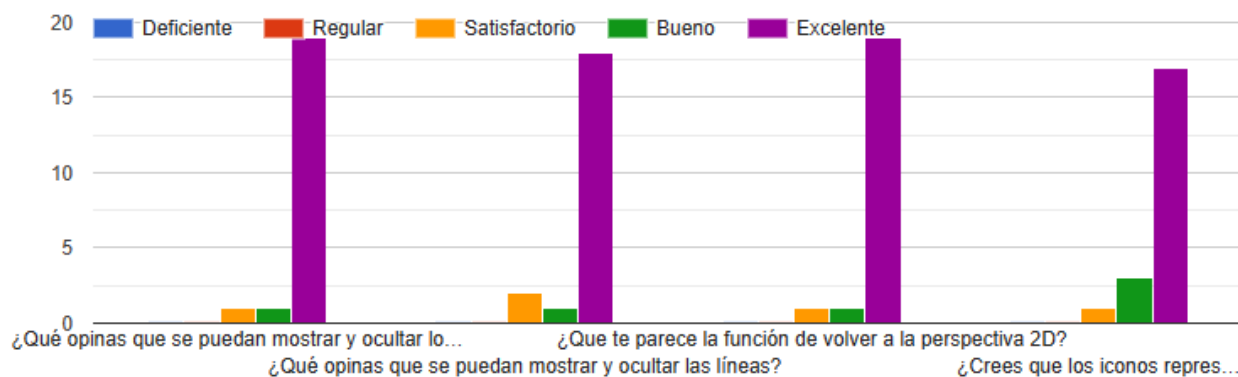


Fig. 47. Preguntas de funcionalidad en opciones

Indique el nivel de acuerdo a sus afirmaciones sobre la aplicación web en los siguientes puntos:

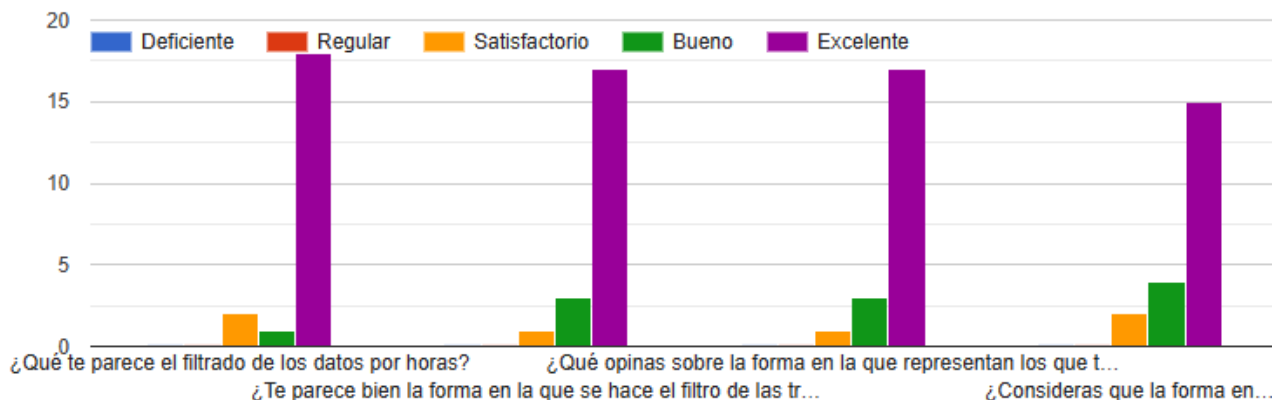


Fig. 48. Preguntas filtrado

Por otro lado, también se tiene en la Figura 49 las respuestas sobre las opiniones de los usuarios relacionados con la facilidad de uso, velocidad de carga de datos y calidad de visualización. En la Figura 50 se presenta que el 71% de las respuestas les pareció excelente la visualización desde las perspectivas 2D y 3D. Y en la Figura 51 se obtiene que un 95.2% está conforme con la descarga de visualización.

Por favor, califica los siguientes aspectos de nuestra aplicación del 1 al 5, donde 1 es "Muy insatisfecho" y 5 es "Muy satisfecho".

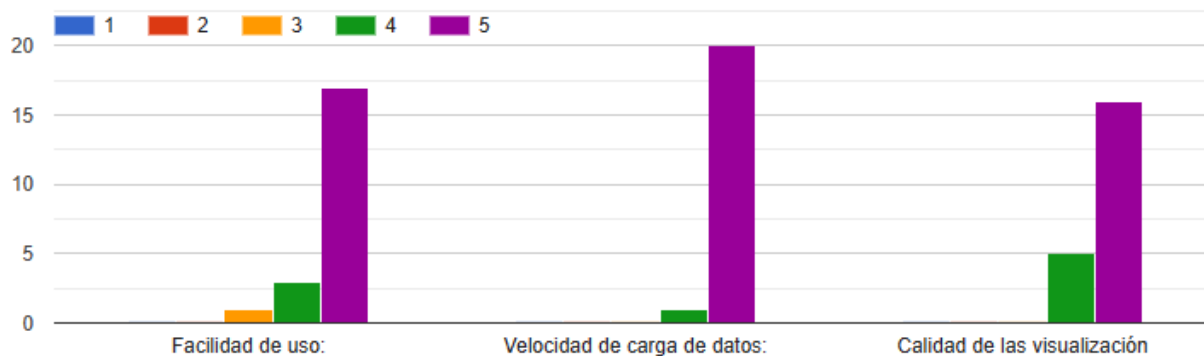


Fig. 49. Preguntas de facilidad de uso, velocidad de cargar y visualización

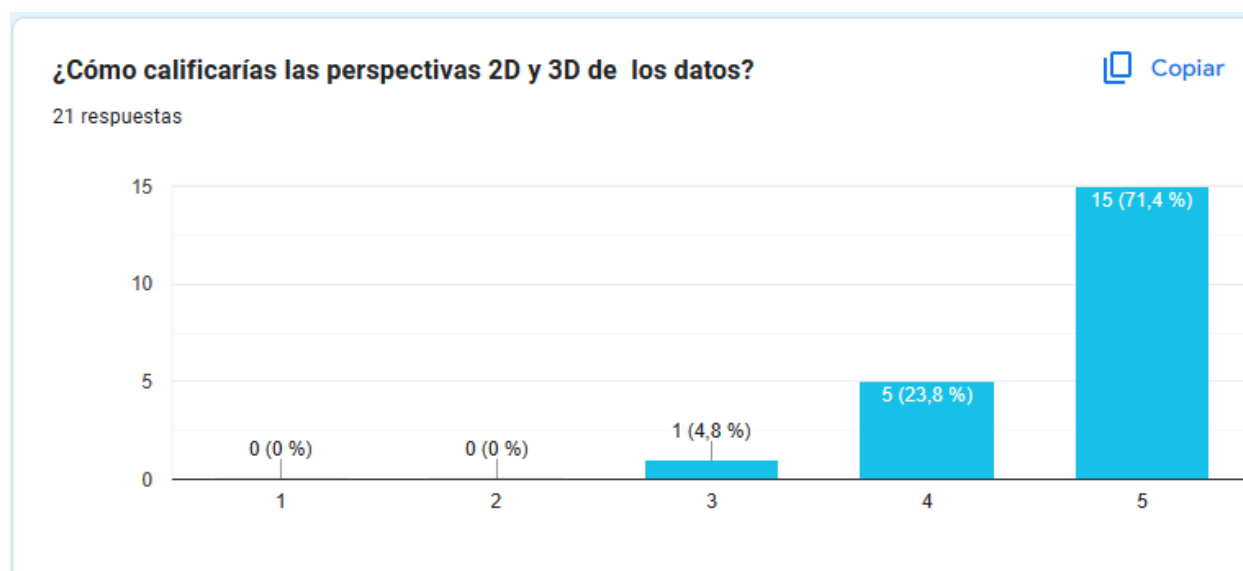



Fig. 50. Calificación de visualización en 2D y 3D



Fig. 51. Respuesta de descarga de visualización

Como últimas pruebas se tiene el funcionamiento de las opciones que contiene el menú al momento de visualizar los datos espacio-temporales. Para ello se utilizará datos del Huracán Rai que sucedió el 12 de diciembre del 201 en Filipinas. En la Figura 52 se muestra los datos con los que se van a realizar las pruebas de funcionamiento.



```
1 {
2   "paths": [
3     {
4       "points": [
5         {
6           "x": 11,
7           "y": -4,
8           "z": "01:00:00"
9         },
10        {
11          "x": 7,
12          "y": -2,
13          "z": "04:00:00"
14        },
15        {
16          "x": 2,
17          "y": -1.5,
18          "z": "07:00:00"
19        },
20        {
21          "x": -2,
22          "y": -1,
23          "z": "09:00:00"
24        },
25        {
26          "x": -5,
27          "y": -0.5,
28          "z": "11:00:00"
29        },
30        {
31          "x": -8,
32          "y": 0.5,
33          "z": "13:00:00"
34        },
35        {
```

Fig. 52. Datos ejemplo

En la Figura 53 se observa el funcionamiento correcto por parte de la opción que se encuentra en visualización, que son los puntos. Al hacer clic se ocultan como se tiene previsto, sin interferir los demás componentes que son las líneas y el mapa.

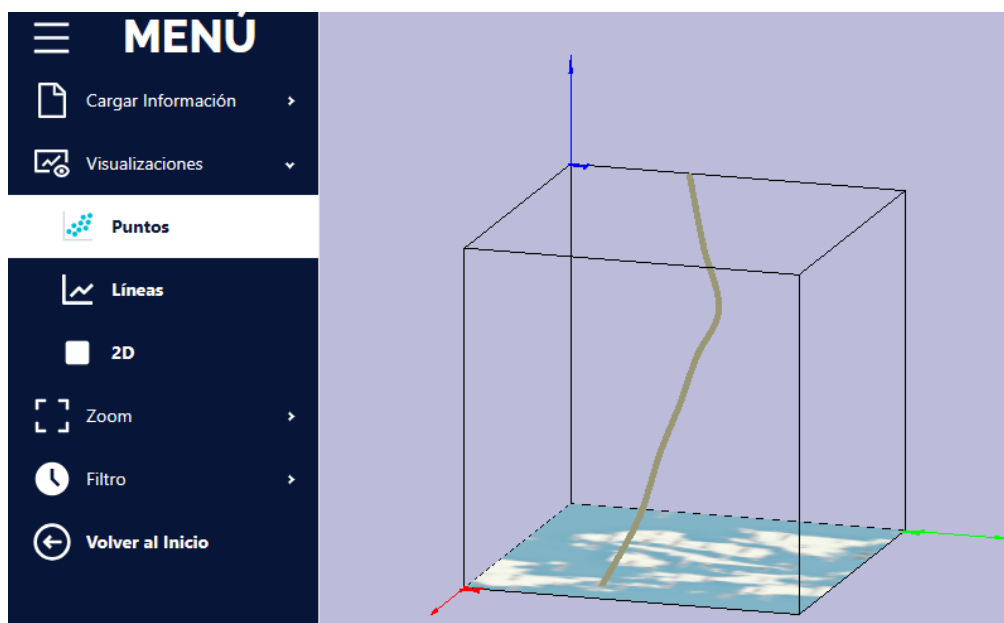


Fig. 53. Ocultar puntos

En la misma sección de visualización se encuentra el apartado de las líneas. En la Figura 54 se muestra cómo se ocultaron todas las líneas, dejando solo los puntos como quiso el usuario.

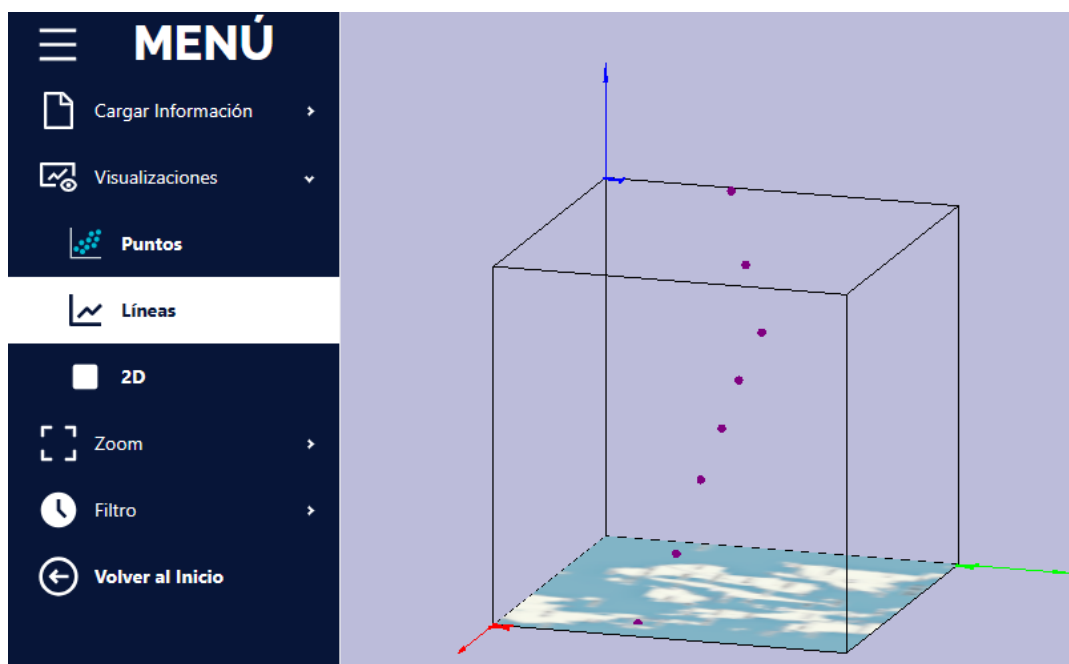


Fig. 54. Ocultar líneas

Y en el tercer apartado de la sección de visualización se encuentra la funcionalidad de 2D. Se va a encargar de colocar el cubo que muestra los datos con sus trayectorias en el desde la perspectiva en segunda dimensión tal y como se muestra en la Figura 55.

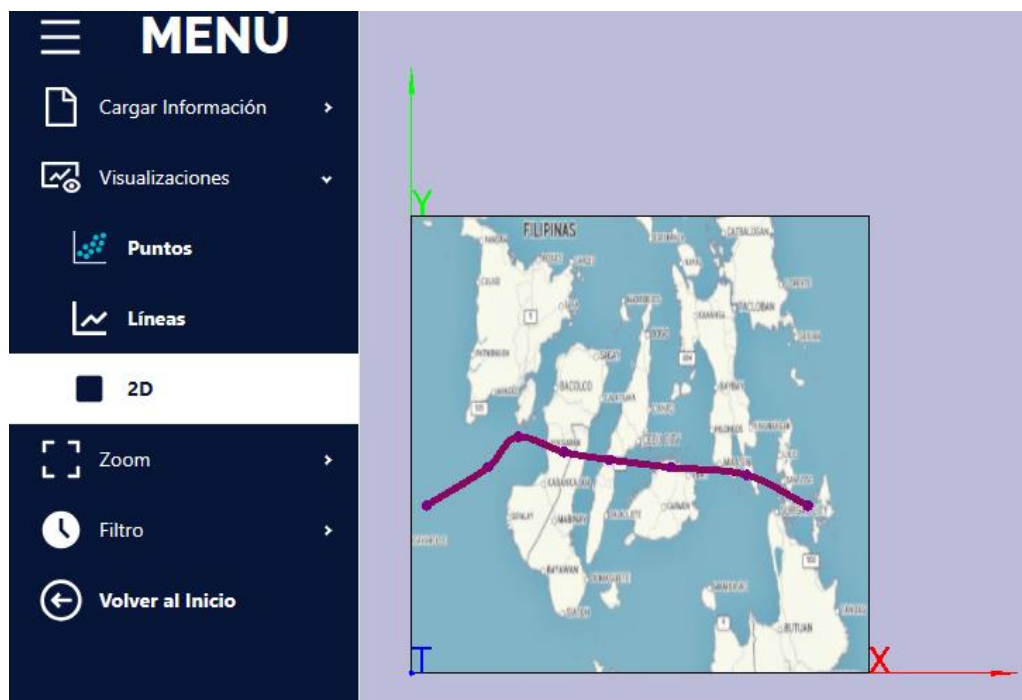


Fig. 55. Perspectiva 2D

En la otra sección llamada Zoom se encuentran 3 apartados los cuales son la función de pantalla completa que se observa en la Figura 56, y las opciones de disminuir y aumentar el tamaño del cubo en las Figura 57 y 58 respectivamente.

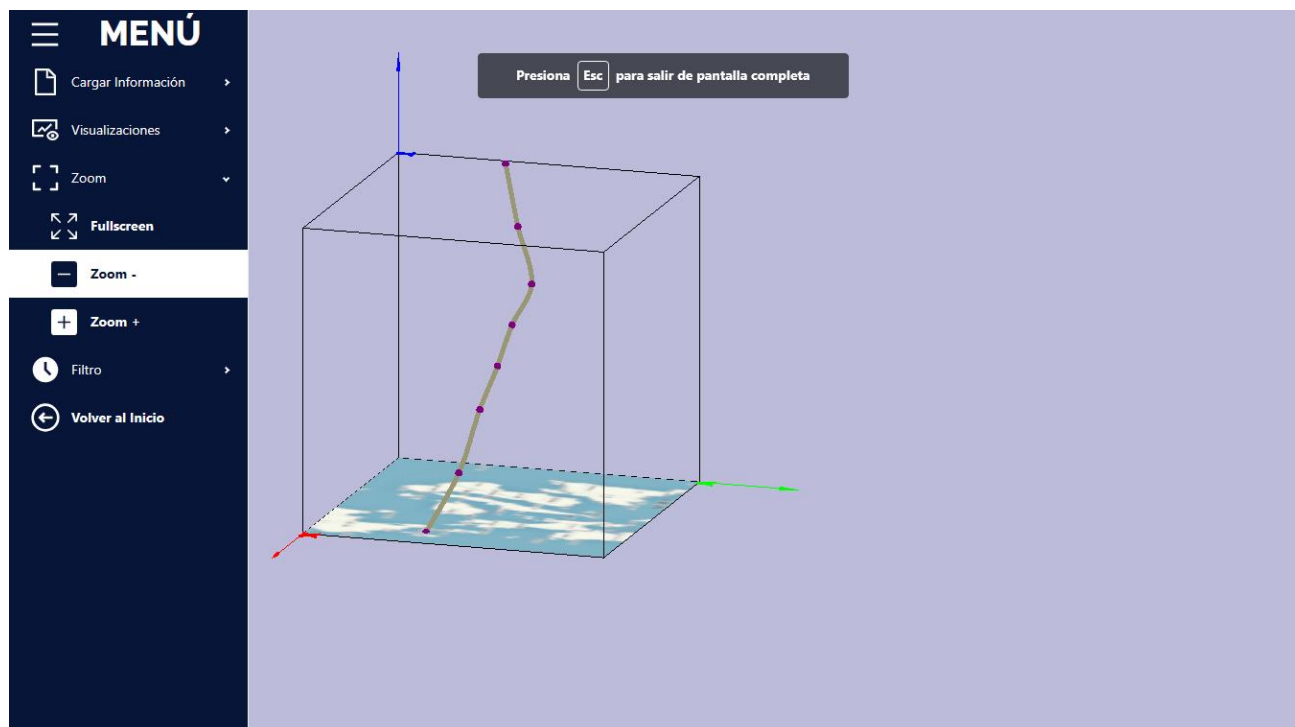


Fig. 56. Pantalla completa

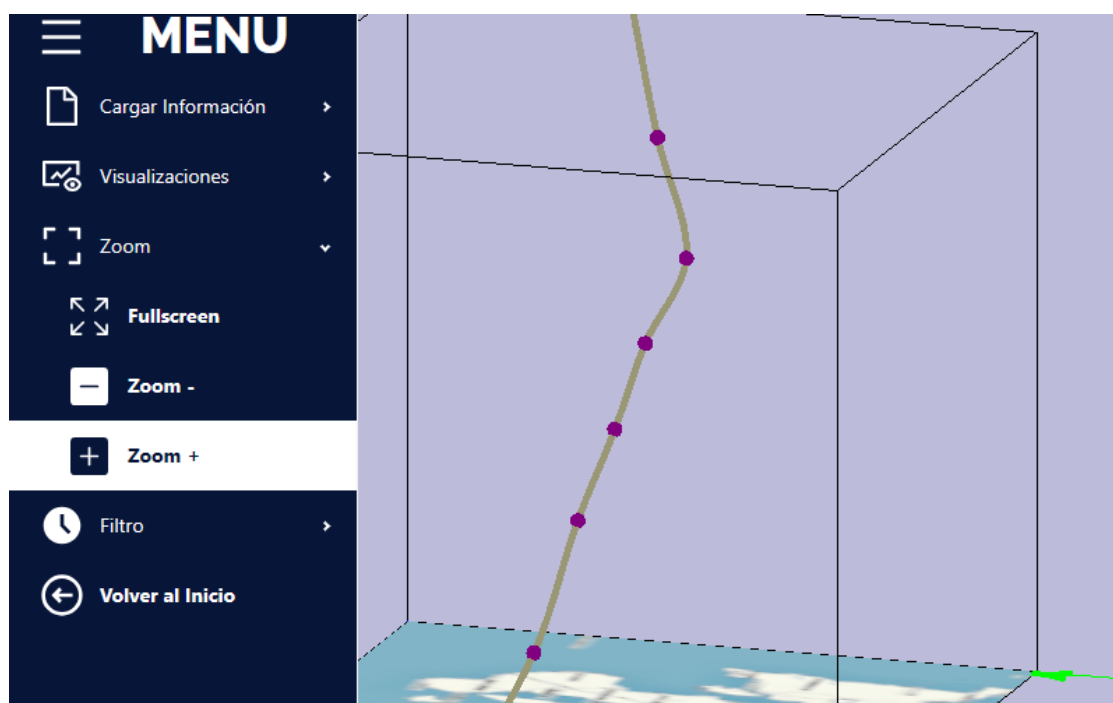


Fig. 57. Zoom +

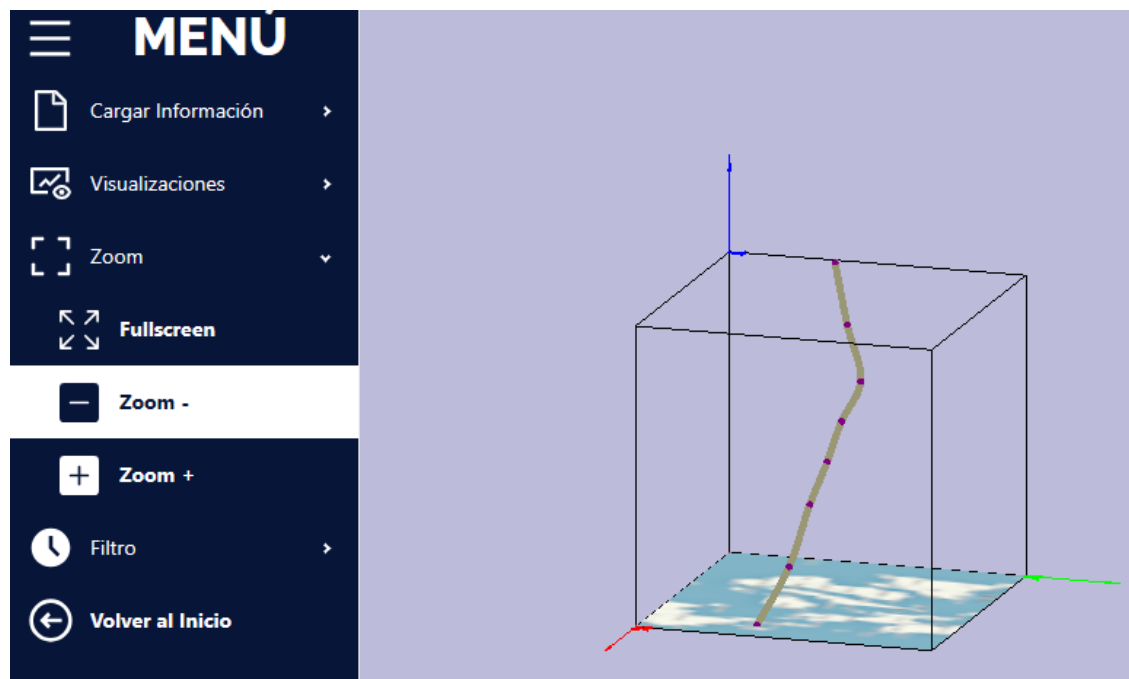


Fig. 58. Zoom -

También se encuentra la sección de filtrado de trayectorias, esta se encarga de ocultar las trayectorias que se seleccionan a través de un checkbox, los cuales van a estar seleccionados y a medida que se van desmarcando las trayectorias se van ocultando. En la Figura 59 se encuentra un ejemplo de varias trayectorias, y en el Figura 60 se encuentra la manera como queda al aplicar el filtro de trayectorias.

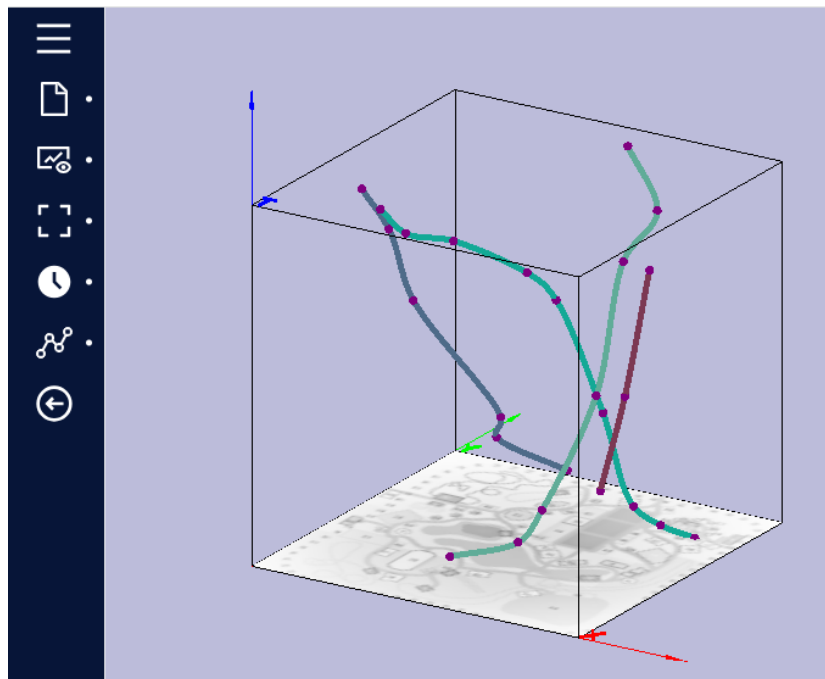


Fig. 59. Varias trayectorias

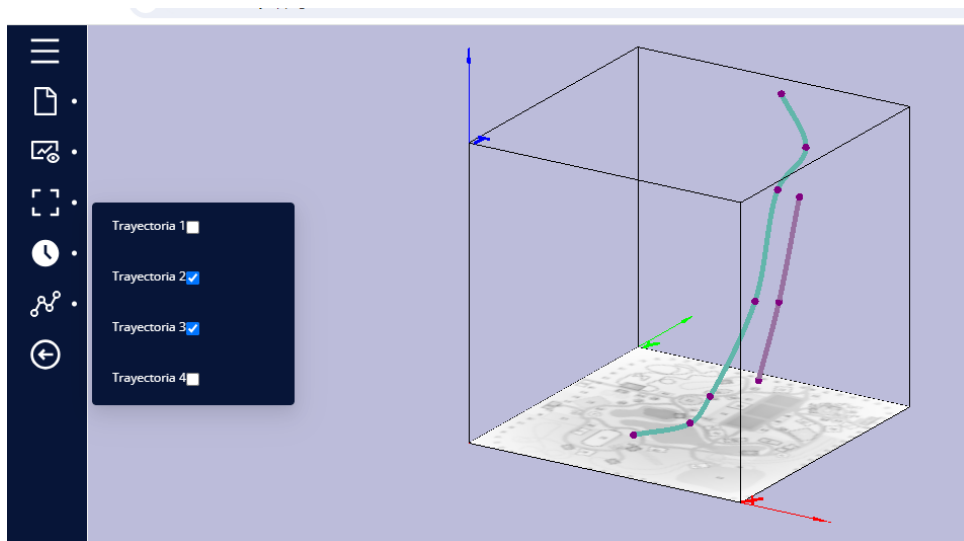


Fig. 60. Filtro de trayectorias

Por último, está la sección de Filtro que se encarga de filtrar los puntos del json mostrado en el cubo. Como se muestra en la Figura 61 se coloca un rango de horas en la solo se muestran los puntos y líneas que se encuentran dentro de ese rango.

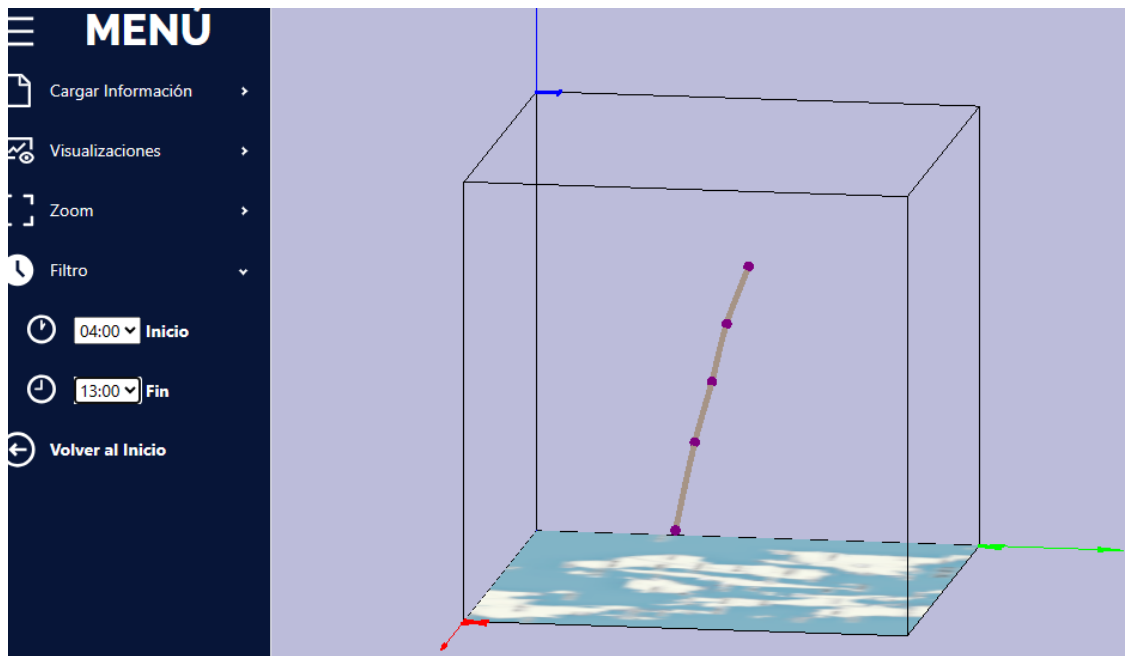


Fig. 61. Filtro

V. CONCLUSIONES Y RECOMENDACIONES

A. Conclusiones

Tras el desarrollo de este proyecto para la visualización de datos espacio-temporales, se determina que esta aplicación web ofrece una forma óptima para presentar las trayectorias que los softwares que existen en el mercado. De esta manera, los usuarios pueden visualizar las trayectorias con sus datos adaptados al archivo JSON específico, con el simple requerimiento que tenga acceso a internet. Es decir, sin la necesidad de instalar herramientas adicionales.

El uso de las metodologías de Design Thinking y Kanban permitió una mayor organización en la implementación del trabajo. Debido a que se adapta a las necesidades que se presentan para el desarrollo de la aplicación, y que ayuda en el trabajo colaborativo debido a que se presenta de manera organizada las actividades para desarrollar el proyecto. Design Thinking permitió presentar la forma en la que se fue adaptando el proyecto. Por otro lado, Kanban permite gestionar el flujo de trabajo del proyecto a través de las asignaciones de tareas, para presentar una visión clara de su implementación.

La elección de la librería de React para el desarrollo de la aplicación web ha facilitado la creación de una interfaz interactiva que permite una experiencia más fluida para los usuarios, debido a su flexibilidad y eficiencia. Esto facilita la gestión de los componentes, siendo crucial en un entorno donde la representación visual de los datos. Además, integra sin problemas otras tecnologías como Three.js para la generación y renderización de elementos tridimensionales, y Node.js para la gestión de dependencias y ejecución de su entorno. Por ende, React ha impulsado el desarrollo eficiente de la aplicación, brindando una base sólida para la visualización eficaz de las trayectorias que se forman a partir de los datos espacio-temporales.

Gracias a la aplicación web TrackView se ha podido analizar la visualización de los datos del huracán RAI que sucedió en 2021. Con la aplicación web se puede observar la trayectoria del tornado con su respectiva hora. Lo que ayuda a tener en cuenta la ubicación donde se encuentran los daños del huracán. Mientras que, en la visualización de los datos de personas del parque, TrackView tiene la capacidad de sacar mejores conclusiones sobre el análisis visual que se realiza al representar estos datos. Ya que, gracias al análisis de la visualización se pueden observar las

zonas en las que las personas circulan más. Y al mismo tiempo en que hora se encuentran las personas de los datos.

B. Recomendaciones

Es importante aplicar de manera correcta el formato del JSON que se va a cargar en la aplicación web. Es decir, realizar la conversión adecuada de los datos espacio-temporales de cualquier ejemplo hacia las características que tiene la aplicación. Con el fin de garantizar su representación y visualización de los datos espacio-temporales otorgados

Realizar otras pruebas con diferentes ejemplos de datos espacio-temporales que tracen una trayectoria, para visualizar la forma en la que se representa su información. Utilizando más ejemplos para evidenciar el funcionamiento de la aplicación web en casos especiales.

Implementar los cambios adecuados cuando al momento de cargar los datos a la aplicación web, indique que hay un error mediante una notificación dentro del formato del archivo JSON. Es decir, corregir los errores dentro formato del JSON que se está cargando dentro de la aplicación para visualizar los datos espacio-temporales.

En el caso de que se quiera volver a poner todos los puntos y líneas de las trayectorias después de aplicar el filtro, solo se debe volver tanto en la hora de inicio como la final el valor que estaba al inicio por defecto (“--:--”).

C. Trabajos a futuro

Uno de los detalles que se puede implementar dentro de la aplicación web a futuro es la importación de diferentes formatos de archivos como XML, HDF, entre otros, para la extracción y visualización de los datos espacio-temporales.

Tomar en consideración la opción de poder aplicar el filtro del tiempo de cada uno de los puntos que se registran junto con la fecha de esta. Esto con la finalidad de que se amplíe el rango en el que transcurre la trayectoria en los casos de que se haga un recorrido de más de un día. Y se pueda implementar un filtro que funcione con las fechas. Además, si es posible aplicar otros tipos de filtrado.

Mostrar barras o líneas que indiquen los valores de cada uno de los ejes direccionales que se encuentran en el cubo. Es decir, que debe haber indicadores de valores en cada línea de los ejes, donde se pueda distinguir como si fuera un eje de coordenadas normal los valores en los que se ubica cada punto, junto con los valores en el que inicia la trayectoria, hasta donde finaliza. Esto para distinguir de mejor manera la normalización de los valores en el tiempo.

Realizar un apartado de transformación de datos, donde se ingresen los datos y se transformen en el formato manejado por la aplicación web. Es decir, que ya no sea necesario que el usuario sea el que realice aparte o de forma individual la conversión de los datos al formato especificado, sino que sea haga desde la misma aplicación web se pueda hacer esta conversión.

VI. REFERENCIAS BIBLIOGRÁFICAS

- [1] S. Scheider, B. Gräler, E. Pebesma, and C. Stasch, “Modeling spatiotemporal information generation,” *International Journal of Geographical Information Science*, vol. 30, no. 10, pp. 1980–2008, Oct. 2016, doi: 10.1080/13658816.2016.1151520.
- [2] P. Moraga, “SpatialEpiApp: A Shiny web application for the analysis of spatial and spatio-temporal disease data,” *Spat Spatiotemporal Epidemiol*, vol. 23, pp. 47–57, Nov. 2017, doi: 10.1016/J.SSTE.2017.08.001.
- [3] C. Zhong, T. Wang, W. Zeng, and S. Müller Arisona, “Spatiotemporal visualisation: A survey and outlook,” *Communications in Computer and Information Science*, vol. 242 CCIS, pp. 299–317, 2012, doi: 10.1007/978-3-642-29758-8_16/COVER.
- [4] M. F. Pang *et al.*, “Spatiotemporal visualization for the global COVID-19 surveillance by balloon chart,” *Infect Dis Poverty*, vol. 10, no. 1, pp. 1–8, Dec. 2021, doi: 10.1186/S40249-021-00800-Z/FIGURES/3.
- [5] C. M. Mendonça de Sá Araújo, I. Miranda Santos, E. Dias Canedo, and A. P. Favacho de Araújo, “Design Thinking Versus Design Sprint: A Comparative Study,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11583 LNCS, pp. 291–306, 2019, doi: 10.1007/978-3-030-23570-3_22/FIGURES/3.

- [6] G. H. Steinke, M. S. Al-Deen, and R. C. LaBrie, "Innovating Information System Development Methodologies with Design Thinking," pp. 1–5, 2017, doi: 10.25673/5884.
- [7] D. Herron, "Node.js Web Development: Server-side web development made easy with Node 14 using practical examples," *Packt Publishing Ltd*, 2020.
- [8] A. Pakkanen, "Rendering a 3D scene in a web application," *Theseus*, pp. 23-24, 2021. [Online]. Available: <http://www.theseus.fi/handle/10024/498942>
- [9] S. bin Uzayr, "Mastering Visual Studio Code: A Beginner's Guide," *CRC Press*, pp. 26-27, 2022.
- [10] Y. L. Erika Dayana and A. G. Kleber Fabián, "Aplicación de la metodología kanban en el desarrollo del software para generación, validación y actualización de reactivos, integrado al sistema informático de control académico UNACH [Universidad Nacional de Chimborazo,2020]," 2020. [Online]. Available: <http://dspace.unach.edu.ec/handle/51000/6457>
- [11] T. B. Fang and Y. Lu, "Constructing a Near Real-time Space-time Cube to Depict Urban Ambient Air Pollution Scenario," *Transactions in GIS*, vol. 15, no. 5, pp. 635–649, 2011.
- [12] M. Huedo Gayo, "Generador de fenómenos atmosféricos para three.js [Universidad Autónoma de Madrid]," 2016. [Online]. Available: <http://hdl.handle.net/10486/676946>
- [13] D. Vanegas and C. Elena, "Caracterización de datos espacio temporales en sistemas de información geográfica," 2013. [Online]. Available: <https://repositorio.unal.edu.co/handle/unal/11829>
- [14] A. R. León Yacelga and M. A. Checa Cabrera, "Uso de tableros Kanban como apoyo para el desarrollo de las metodologías ágiles," *Revista Universidad y Sociedad*, vol. 14, no. S2, pp. 208-214, 2022.
- [15] J. Jimenez, "Descubre React," 2015. [Online]. Available: Octubre, 2, 2018.

- [16] H. F. Villamar Barros, "Visualización de datos Espacio-Temporales de predicción de Eventos de Emergencia en el Ecuador," *Repositorio de La Universidad Estatal de Milagro*, 2023. [Online]. Available: <http://repositorio.unemi.edu.ec/xmlui/handle/123456789/7004>
- [17] J. G. Rodriguez-Peña, "Propuesta para un nuevo enfoque de aplicación ágil con Scrum, LSD y Kanban," *Pontificia Universidad Católica de Valparaíso*, Valparaíso, 2016.
- [18] S. Cao et al., "Research on the Visualization of Spatio-Temporal Data," *IOP Conference Series: Earth and Environmental Science*, vol. 234, no. 1, p. 012013, 2019. [Online]. Available: <https://doi.org/10.1088/1755-1315/234/1/012013>
- [19] A. Tritsarolis, C. Doulkeridis, N. Pelekis, and Y. Theodoridis, "STVISIONS: A Python Library for Interactive Visualization of Spatio-Temporal Data," in *Proceedings - IEEE International Conference on Mobile Data Management*, 2021, pp. 244–247. [Online]. Available: <https://doi.org/10.1109/MDM52706.2021.00048>
- [20] J. E. N. Perugachi, "Recorrido Virtual como impulso turístico utilizando frameworks de WebGL," Quito, [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/64217769/PAPER_EstefaniaNicolaldewith-cover-pagev2.pdf.
- [21] L. A. Ortiz Delgado, "Comparativa de trayectorias espaciotemporales 2D vs 3D," 2018.
- [22] C. Nemocón, "MEDET Modelo para la exploración de datos espacio – temporales," pp. 1–45, 2015.