



**DESARROLLO DE UN COMPONENTE WEB PARAMETRIZABLE PARA LA
VISUALIZACIÓN DE DATOS CIENTÍFICOS**

**UNIVERSIDAD CATÓLICA DE COLOMBIA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ D.C
Año 2017**

**DESARROLLO DE UN COMPONENTE WEB PARAMETRIZABLE PARA LA
VISUALIZACIÓN DE DATOS CIENTÍFICOS**

**Edward Jiménez Hernández
Cesar Yesid Pineda Patarroyo**

**Trabajo de Grado para Optar al Título de
Ingeniero de Sistemas**

**DIRECTOR
DIEGO ALBERTO RINCÓN YÁÑEZ MSc
INGENIERO DE SISTEMAS**

**UNIVERSIDAD CATÓLICA DE COLOMBIA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ D.C
2017**



Atribución 2.5 Colombia (CC BY 2.5 CO)

Este es un resumen legible por humanos (y no un sustituto) de la [licencia](#).

[Advertencia](#)



Usted es libre para:



Compartir — copiar y redistribuir el material en cualquier medio o formato

Adaptar — remezclar, transformar y crear a partir del material

Para cualquier propósito, incluso comercialmente

El licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:



Atribución — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

No hay restricciones adicionales — Usted no puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otros hacer cualquier uso permitido por la licencia.

Aviso:

Usted no tiene que cumplir con la licencia para los materiales en el dominio público o cuando su uso esté permitido por una excepción o limitación aplicable.

No se entregan garantías. La licencia podría no entregarle todos los permisos que necesita para el uso que tenga previsto. Por ejemplo, otros derechos como relativos a publicidad, privacidad, o derechos morales pueden limitar la forma en que utilice el material.

NOTA DE ACEPTACIÓN

Aprobado por el comité de grado en cumplimiento de los requisitos exigidos por la Facultad de Ingeniería y la Universidad Católica de Colombia para optar al título de Ingenieros de Sistemas.

Jurado

Diego Alberto Rincón
Director

Revisor Metodológico.

Fecha de Entrega

AGRADECIMIENTOS

Le agradecemos especialmente a nuestro director de trabajo de grado Diego Alberto Rincón, por su apoyo incondicional en el proceso de elaboración y producción de la presente.

A nuestros familiares por creer en nosotros y ofrecernos su colaboración en este importante suceso de nuestras vidas.

TABLA DE CONTENIDO

1. GENERALIDADES	12
1.1. ANTECEDENTES.....	12
1.2. PLANTEAMIENTO DEL PROBLEMA.....	14
1.3. DESCRIPCIÓN DEL PROBLEMA	15
1.4. JUSTIFICACIÓN.....	16
1.5. DELIMITACIÓN	17
1.5.1. Alcance	17
1.5.2. Limitaciones	17
2. OBJETIVOS DEL PROYECTO	18
2.1. OBJETIVO GENERAL.....	18
2.2. OBJETIVOS ESPECÍFICOS.....	18
3. MARCO DE REFERENCIA.....	18
3.1. ESTADO DEL ARTE.....	18
4. MARCO CONCEPTUAL.....	26
5. METODOLOGÍA.....	26
6. DESARROLLO DEL PROYECTO.....	29
6.1. ANÁLISIS.....	29
6.2. DISEÑO.....	30
6.3. DESARROLLO (CODIFICACIÓN).....	31
7. RESULTADOS.....	35
7.1. ANÁLISIS DE RESULTADOS	37
8. CONCLUSIONES.....	42
9. TRABAJOS FUTUROS	43
10. BIBLIOGRAFÍA	43
11. GLOSARIO	46
12. ANEXOS.....	48

TABLA DE ILUSTRACIONES

Ilustración 1. Tabla de precios	15
Ilustración 2 Proceso de Big Data(Rincón, 2016)	20
Ilustración 3. Localizaciones de menor concentración de material particulado(Abiertos, 2016)	23
Ilustración 4. A Nation of Wineries(Nytimes, 2013)	23
Ilustración 5. Clouds of words(Auraria Library, 2017)	24
Ilustración 6. Libor: The Spider Network(Jovi Juan, 2015)	24
Ilustración 7. Bogotá, Colombia(Maps, 2017)	25
Ilustración 8. Confecciona tu Pep Team(Marca, 2009)	25
Ilustración 9. Configuración del proyecto con ExpressJS	32
Ilustración 10. Diagrama de Componentes	33
Ilustración 11. Grafica de líneas.....	34
Ilustración 12. Grafica de barras	34
Ilustración 13. Grafica de torta	35
Ilustración 14. Caso Pruebas Unitarias	36
Ilustración 15. Resultado Pruebas Unitarias	37
Ilustración 16. Endpoint grafica de torta.....	38
Ilustración 17. Visualización grafica de torta	38
Ilustración 18. Endpoint de categorías, Eje X	39
Ilustración 19. Endpoint de grafico de barras y líneas	39
Ilustración 20. Visualización de la gráfica de barra.	40
Ilustración 21. Visualización de la gráfica de línea.....	40
Ilustración 22. Componente de Visualización de datos científicos.....	41

TABLA DE ANEXOS

ANEXO 1	48
ANEXO 2	48
ANEXO 3	81

ABSTRACT

This project was based on the development of a parametrizable web component for the visualization of scientific data, which establishes a visual analysis, in the first instance for the project of the psychology faculty of emotional reexperimentation and Mindfulness but seeing the potential of the component, this It adapts to any project under certain established standards. To carry out the visualization component, some concepts of Big Data, visual analysis, dashboard and graphic taxonomy were analyzed, emphasizing our process. The methodology proposed for the development of the visualization component is based on the methodology of agile development XP or extreme programming(Lappo, 2005) and through a set of practices, gives an application focused on development in a unique or exclusive way for a single programmer. Taking advantage of their execution characteristics in which they classify it within the group of agile development methodologies, which involve software engineering methods based on iterative and incremental development, these methodologies are necessary in a dynamic and unpredictable world such as ours. to propose an agile manifesto which, according to values or prioritizes the individual and the interactions of the development team about the process and the tools. The interaction of the visualization component with the other components of the project implied the use of technologies for development, visual alternatives and consumption of web service, which allowed to carry out a process of analysis, design and development of the component in an effective way and thus allow the component can continue to evolve and be a tool that offers a variety other than those that are freely used in the market.

Key Words: Big Data, Dashboard, Graphic Taxonomy, Visual Analysis, Visualization.

RESUMEN

Este proyecto se fundamentó en el desarrollo de un componente web parametrizable para la visualización de datos científicos, que establece un análisis visual, en primera instancia para el proyecto de la facultad de psicología de re experimentación emocional y Mindfulness pero viendo el potencial del componente, este se adapta a cualquier proyecto bajo ciertos estándares establecidos. Para llevar a cabo el componente de visualización se analizó algunos conceptos de Big Data, análisis visual, dashboard y taxonomía de graficas haciendo énfasis en nuestro proceso. La metodología propuesta para el desarrollo del componente de visualización toma como base la metodología de desarrollo ágil XP o programación extrema y a través de un conjunto de prácticas, da una aplicación enfocada al desarrollo de manera singular o exclusiva para un solo programador. Aprovechando sus características de ejecución en el que la clasifican dentro del grupo de metodologías de desarrollo ágiles, las cuales implican métodos de ingeniería del software basados en el desarrollo iterativo e incremental, estas metodologías son necesarias en un mundo dinámico e impredecible como el nuestro, además de plantear un manifiesto ágil el cual según valora o da prioridad al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La interacción del componente de visualización con los demás componentes del proyecto implico el uso de tecnologías para el desarrollo, alternativas visuales y consumo de web service, que permitieron realizar un proceso de análisis, diseño y desarrollo del componente de manera eficaz y así permitir que el componente pueda seguir evolucionando y ser una herramienta que ofrece una variedad distinta a las que son de uso libre en el mercado.

Palabras Claves: Análisis Visual, Big Data, Dashboard, Taxonomía de gráficas, Visualización.

INTRODUCCIÓN

Los datos se han convertido en una parte clave en la política, la ciencia, la economía y las estructuras empresariales, y ahora incluso vidas sociales. Esta tendencia es claramente visible en las redes sociales como Facebook, Twitter e Instagram donde los usuarios generan una enorme cantidad de diferentes tipos de información diarias (música, imágenes, textos, entre otros.)(McKinsey & Company, 2011). Con los grandes volúmenes de datos, la complejidad del análisis presenta un desafío innegable a la hora de desarrollar técnicas y métodos de visualización que deben siempre estar en constante mejora.

Independientemente de la cantidad de datos que tenga, una de las mejores maneras de discernir relaciones importantes, es a través visualizaciones fáciles de entender; con el fin de abordar diferentes frentes al momento de analizar fuentes de información que apuntan a una finalidad(Chen & Jin, 2017), es muy importante considerar características tales como el tamaño de los datos, el tipo de datos, la composición de las columnas y el mensaje que se quiere expresar en ellos.

Pero ¿Cómo puedo visualizar un gran volumen de datos y obtener un beneficio del análisis visual?

Este documento tiene como objetivo, proponer formalmente el desarrollo de un componente web parametrizable para la visualización de datos científicos.(Leimer, Gersthofer, Wimmer, & Musialski, 2017) Este desarrollo está acogido por un proyecto padre el cual tiene un enfoque psicológico y el cual busca plantear la efectividad de un protocolo de reexperimentación emocional y mindfulness en adultos expuestos a situaciones traumáticas en un contexto de violencia política, en el cual, su primera fase requiere de la construcción de esta herramienta para posteriormente realizar el análisis visual y se adaptará a los objetivos del proyecto padre, no obstante, este proyecto tiene como finalidad la construcción de un componente que sirva para diferentes propósitos de análisis de información de forma visual y que sea de fácil análisis para el investigador.(Abdelsadek, Chelghoum, Herrmann, Kacem, & Otjacques, 2018)

Para que todo este proceso se lleve a cabo de la forma correcta, se debe hacer una previa investigación y estudio acerca de las metodologías, conceptos, antecedentes y demás información relevante al proyecto. Esta información es de carácter formativo en la cual se expone un punto de vista relacionado en el entorno en que se desarrolla el proyecto. Esto permite dar un contexto de integración de la información y da a conocer el porqué es necesaria la construcción de una herramienta de este tipo.

1. GENERALIDADES

1.1. ANTECEDENTES

Se realiza un balance de investigaciones que han realizado diferentes compañías, referente al tema de visualización de datos científicos; En este sentido se realiza un análisis sobre las herramientas o prototipos que han realizado y más importante sobre las taxonomías aplicadas a la visualización de datos científicos.

- **Big Data With SAS® Visual**

Big data trae nuevos desafíos a la visualización debido a la velocidad, el tamaño y la diversidad de datos que deben tenerse en cuenta. También se debe considerar la cardinalidad de las columnas que está intentando visualizar.

Una de las definiciones más comunes de grandes datos son los datos que son de tal volumen, variedad y la velocidad que una organización debe moverse más allá de su zona de confort tecnológicamente para derivar inteligencia para tomar decisiones efectivas.

- Volumen se refiere al tamaño de los datos.
- Variedad describe si los datos están estructurados, semiestructurados o no estructurados.
- La velocidad es la velocidad a la que fluyen los datos y con qué frecuencia cambia.

Basándose en las técnicas básicas de gráficos y visualización, SAS Visual Analytics ha adoptado un enfoque innovador para abordar los desafíos asociados con la visualización de bigdata. Utilizando capacidades innovadoras de memoria combinadas con SAS Analytics y descubrimiento de datos, SAS ofrece nuevas técnicas basadas en los fundamentos básicos del análisis de datos y la presentación de resultados(photo, n.d.).

Tableplots

El tableplot es un método de visualización que se utiliza para explorar y analizar grandes conjuntos de datos. Tableplots se utilizan para explorar las relaciones entre las variables, para descubrir patrones de datos extraños, y para comprobar la ocurrencia y la selectividad de los valores perdidos.

Tableplots están dirigidos para visualizar conjuntos de datos multivariados con varias variables y un gran número de registros, por ejemplo, al menos un millar. Tableplots también se pueden generar para conjuntos de datos con menos registros, pero pueden ser menos útiles. El número máximo de filas que puede ser visualizado con el `tableplot` package depende de la memoria del software R ya que `tableplot` se integra como `plugin` a la herramienta (Tennekes, 2013).

Tableau

Tableau Software es una empresa estadounidense de software de computadora con sede en Seattle, WA, EE. UU. Produce una familia de productos interactivos de visualización de datos enfocados en la inteligencia de negocios.

La compañía fue fundada en el departamento de la universidad de Stanford de la informática entre 1997 y 2002. Profesor Pat Hanrahan y Ph.D. El estudiante Chris Stolte, que se especializó en técnicas de visualización para explorar y analizar bases de datos relacionales y cubos de datos, condujo la investigación en el uso de pantallas basadas en tablas para explorar bases de datos relacionales multidimensionales.

Juntos, combinaron un lenguaje de consulta estructurado para bases de datos con un lenguaje descriptivo para renderizar gráficos e inventaron un lenguaje de visualización de base de datos llamado VizQL (Visual Query Language). VizQL formó el núcleo del sistema Polaris, una interfaz para explorar grandes bases de datos multidimensionales. En 2003, después de que Stolte reclutara a su antiguo socio comercial y amigo, Christian Chabot, para desempeñarse como CEO Tableau, se separó de Stanford con una aplicación de software equivalente. (Tableau, 2017)

El producto consulta las bases de datos relacionales, los cubos, la base de datos en la nube y las hojas de cálculo y luego genera una serie de tipos de gráficos que se pueden combinar en cuadros de mando (Dashboard) y compartidos a través de una red informática o de Internet (Sood, Sinha, Dewjee, & Zhao, 2013).

Power Bi

Esta herramienta es proporcionada por Microsoft y es un conjunto de aplicaciones de análisis de negocios la cual analiza los datos y permite compartir la información. Los paneles estadísticos muestran lo mejor de sus métricas y también permite actualizar la información en tiempo real. Con un menor esfuerzo los usuarios pueden

explorar los datos del panel mediante herramientas intuitivas que brindan respuestas de manera sencilla. La creación de un panel es una sencilla operación los cuales fueron diseñados de manera cuidadosa por expertos para ayudar al usuario a poner en marcha su informe.

Proporciona informes y análisis a su organización es una herramienta de mashup de datos que incluye numerosas características, además de combinar datos de bases de datos, archivos y servicios web con herramientas visuales que servirán de gran ayuda para poder corregir problemas de formato y calidad de los datos automáticamente. Esta herramienta puede unificar todos los datos de su organización, ya sea en la nube o localmente, puede conectar bases de datos SQL Server y muchos otros orígenes de datos.(Microsoft, 2017)

Domo

Comenzó con ejecutivos altamente respetados, altamente frustrados que se sentían excluidos de sus propios datos de negocio. Ahora, con más de \$ 500 millones en fondos, sirve a ejecutivos de todas las clases, y en una amplia gama de industrias. Conecta a su personal con los datos que necesitan para mejorar los resultados del negocio. Puede elegir entre más de 1.000 aplicaciones para aprovechar rápidamente la experiencia y los flujos de trabajo de su equipo necesita. Además, construir aplicaciones personalizadas rápidamente con herramientas de diseño de gran alcance.

Tanto si es una organización con 50 usuarios o una empresa con 50.000, la nube de negocios escalable puede satisfacer sus necesidades. Y debido a la nube de negocios es una plataforma de sistema abierto, que no está limitado a un solo modelo o despliegue metodología específica. La nube de negocios dispone de soluciones de adquisición de gran alcance que traen los datos pertinentes, sin importar dónde se encuentren almacenados. Y una vez que los datos están en Domo, esta herramienta ETL dinámica ayudan a los usuarios de cualquier nivel que esté listo para la visualización.(Domo, 2017)

1.2. PLANTEAMIENTO DEL PROBLEMA

En la actualidad se almacenan grandes volúmenes de datos los cuales siguen en aumento a través del tiempo y del cual es necesario usarlos para sacar un mayor provecho de ellos. Estos datos están relacionados a diferentes empresas, instituciones y personas que pueden estar en diferentes sectores de la industria y la ciencia.

Si se centra en la persona, las redes sociales son una fuente de datos que nos permiten examinar el qué está pensando esa persona cuando lo plasma y que ciertas herramientas extraen estos datos y los almacenan en de una manera adecuada.

Este tipo de visualizaciones las puede realizar softwares populares como, por ejemplo, Tableau, que permite que el cliente pueda administrar de una forma centralizada todos sus metadatos y en temas de procesamiento puede tener acceso a gran velocidad de millones de datos y graficarlos. Pero en términos de precios es costoso ya que tiene un valor de USD 2000 anuales en su versión profesional.

Como lo muestra la siguiente tabla, se relaciona por herramienta el costo que representa la adquisición de licencias y el periodo de validez. Mostrando los altos costos que una herramienta completa y administrable por los usuarios representa.

Ilustración 1. Tabla de precios

HERRAMIENTA	PRECIO	PERIODO
CartoDB Enterprise	USD 825,00	Mensual
Tableau Enterprise	USD 2.000,00	Anual
Quadrigram Professional	USD 63,27	Mensual
iCharts Professional	USD 64,95	Mensual
Q Research Software Standard	USD 1.499,00	Anual
Jolicharts Enterprise	USD 74,06	Mensual
Plotly	USD 33,00	Mensual
chartblocks Elite	USD 65,00	Mensual
infogr.am Professional	USD 67,00	Mensual
datawrapper Professional	EUR 279,00	Mensual
Highcharts	USD 595,00	Anual

Si se pretende tomar cualquier set de datos los cuales han tenido previamente un proceso de almacenamiento y análisis. ¿Cómo podría visualizar esta información de una manera entendible?

1.3. DESCRIPCIÓN DEL PROBLEMA

La presente investigación implementa un componente de visualización parametrizable de datos científicos para una herramienta de extracción y almacenamiento de datos, con el fin de mostrar y comprender cantidades de datos para así tomar decisiones basadas en información eficaz y veraz.

La visualización de datos es una de las técnicas de análisis de datos que más tiene demanda y que se da mucha importancia. Esto porque a través de una gráfica

resulta fácil detectar patrones de datos. Es bastante útil para poder buscar y entender de una forma rápida y simplificada grandes volúmenes de datos.

El planteamiento iniciación del proyecto estaba basado en un trabajo de investigación en conjunto con la facultad de psicología, en el proyecto Efectividad de un protocolo de re-experimentación emocional y mindfulness en adultos expuestos a situaciones traumáticas en un contexto de violencia política, donde se realizan estudios de grupos que realizan acciones violentas donde construyen justificaciones ideológicas lo cual se busca es minimizar el impacto emocional negativo del ser humano pero en un contexto político que normalmente existen partes a favor y en contra de ciertos temas. Se tiene presente que se han generado una cantidad de datos, los cuales están en un proceso de almacenamiento y curación que posteriormente, se les realizará un proceso de análisis para hallar ciertos patrones siguiendo la correspondiente directriz del proyecto con el enfoque de violencia política. Pero luego de un proceso de análisis se realizó un nuevo planteamiento, que consiste en desarrollar una herramienta que permita la visualización no solo del proyecto planteado por la facultad de psicología, sino que se puedan anexar diferentes investigaciones que conlleven la visualización de sets de datos que puedan ser parametrizados. (Naur & Randell, 1968)

De acuerdo con el marco establecido con el director del proyecto, el componente de visualización hace parte de un macroproyecto de Big Data, el cual está dividido en diferentes etapas. Al haber finalizado las etapas de preparación, y teniendo total veracidad en la información la cual debe haber pasado por una etapa de análisis, debe existir un componente de visualización muy significativo. Sin importar si todos los procesos son automatizados o no, la forma de presentar la información ya gobernada y analizada es clave para realizar la implementación de los diferentes componentes genéricos que permita transformar los datos o información en gráficas.

Así mismo, el objetivo del componente es lograr transformar grandes volúmenes de datos independientemente del origen y el contexto de la investigación que permitan a los usuarios, obtener información que admita tomar decisiones en un periodo de tiempo corto y que sea parametrizados.(Arbib & Manes, 1982)

1.4. JUSTIFICACIÓN

La presente investigación implementa un componente de visualización parametrizable de datos científicos para una herramienta de extracción y almacenamiento de datos, con el fin de mostrar y comprender cantidades de datos para así tomar decisiones basadas en información eficaz y veraz.

La visualización de datos es una de las técnicas de análisis de datos que más tiene demanda y que se da mucha importancia. Esto porque a través de una gráfica resulta fácil detectar patrones de datos. Es bastante útil para poder buscar y

entender de una forma rápida y simplificada grandes volúmenes de datos.(Franklin et al., 2017)

Como continuación al trabajo de investigación en conjunto con la facultad de psicología, en el proyecto Efectividad de un protocolo de re-experimentación emocional y mindfulness en adultos expuestos a situaciones traumáticas en un contexto de violencia política, donde se realizan estudios de grupos que realizan acciones violentas donde construyen justificaciones ideológicas lo cual se busca es minimizar el impacto emocional negativo del ser humano pero en un contexto político que normalmente existen partes a favor y en contra de ciertos temas.(Sacha et al., 2017) Se tiene presente que se han generado una cantidad de datos, los cuales están en un proceso de almacenamiento y curación que posteriormente, se les realizará un proceso de análisis para hallar ciertos patrones siguiendo la correspondiente directriz del proyecto con el enfoque de violencia política.

Al haber finalizado las etapas de preparación, y teniendo total veracidad en la información, debe existir un componente de visualización muy significativo. Sin importar si todos los procesos son automatizados o no, la forma de presentar la información ya gobernada y analizada es clave para realizar la implementación de los diferentes componentes genéricos que permita transformar los datos o información en gráficas.(Vilarinho, Lopes, & Sousa, 2017)

Así mismo, el objetivo del componente es lograr transformar grandes volúmenes de datos que permitan a los usuarios, obtener información que admita tomar decisiones en un periodo de tiempo corto.

1.5. DELIMITACIÓN

Como aspectos importantes que puede afectar de manera positiva se señala en los numerales siguientes.

1.5.1. Alcance

- ❖ El periodo de tiempo de implementación es menor a 1 año (6 meses aproximadamente).
- ❖ La elaboración de esta herramienta permitirá una visualización limpia de los datos para un posterior análisis.

1.5.2. Limitaciones

- ❖ La actualización o retroalimentación de la herramienta se ve comprometida por tiempo definido

- ❖ La visión planteada por el proyecto general implica que este se desarrolle por fases, en las cuales el proyecto propuesto tomaría el papel de visualización de datos y estará sujeto a fases previas que se han desarrollado o están en desarrollo.
- ❖ La implementación del prototipo se llevará a cabo en el segundo semestre del año 2017.
- ❖ El tiempo de desarrollo está limitado por el periodo académico.

2. OBJETIVOS DEL PROYECTO

2.1. OBJETIVO GENERAL

Desarrollar un componente web parametrizable para la visualización de datos científicos, en el contexto de la implementación de una plataforma analítica desarrollado por la Universidad Católica de Colombia.

2.2. OBJETIVOS ESPECÍFICOS

- Realizar el análisis de un componente de visualización parametrizable para mostrar datos científicos.
- Diseñar un componente de visualización.
- Desarrollar el componente de visualización, aplicando una metodología de desarrollo de software.
- Realizar una fase de pruebas que permita certificar la veracidad de las conclusiones expuestas en la visualización de los datos.

3. MARCO DE REFERENCIA

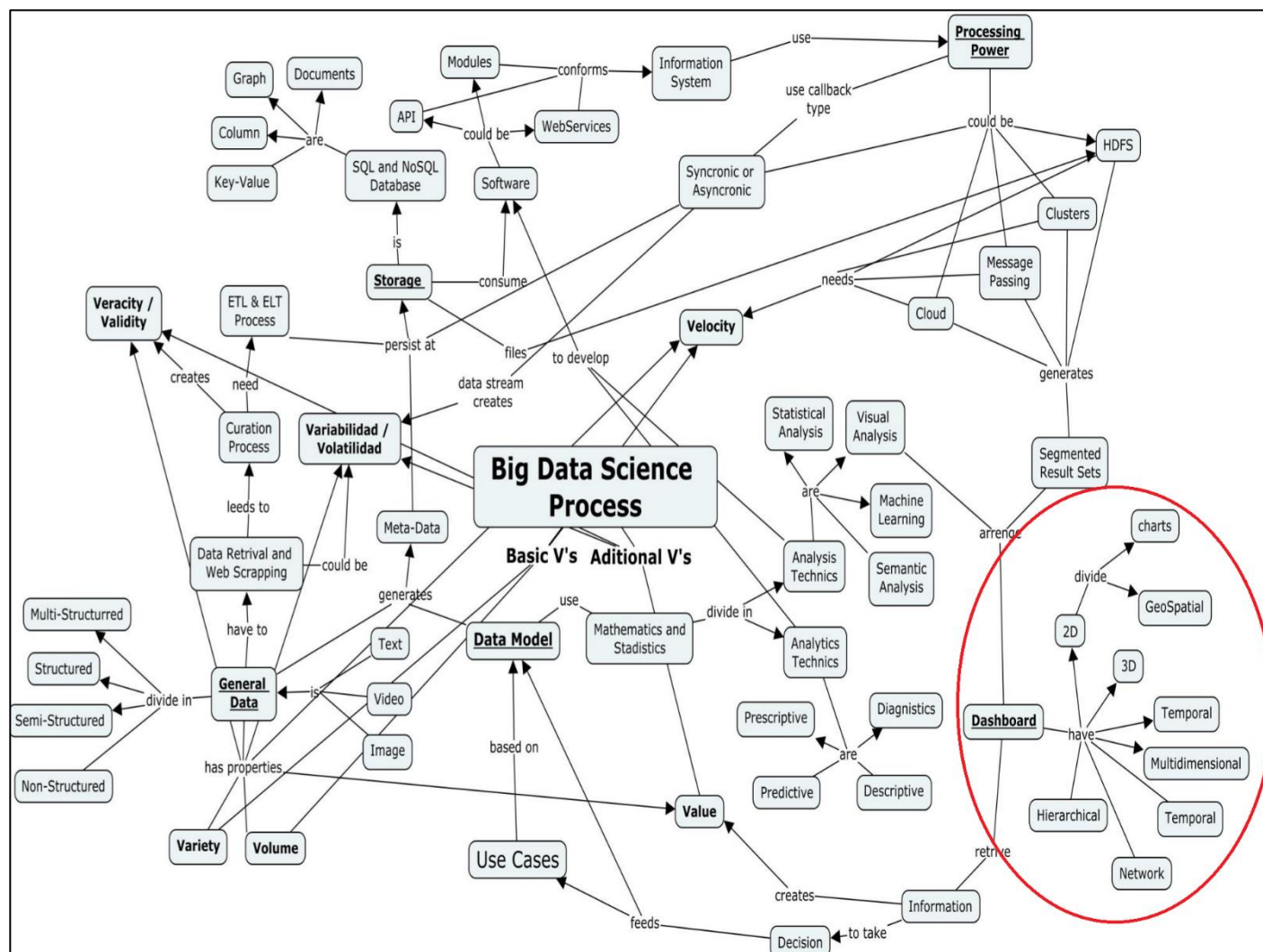
3.1. ESTADO DEL ARTE

La comprensión del Big Data genera un nuevo reto en cuanto a la visualización de datos, debido a la cantidad y diversidad de variables y datos que se manejan. Analizar la información a través de imágenes es más sencillo, de modo que: Los Data Scientists que utilizan la visualización de datos para presentar la data, tienen como aproximado un 28% más de probabilidad de encontrar datos relevantes para el análisis y/o predicción de tendencias en proyectos o negocios a fines, que los que analizan la información en dashboards tradicionales(Centric, n.d.).

La visualización de datos es la forma óptima de exponer la infinidad de datos que están al alcance las organizaciones en el auge de lo conocido como Big Data. Debido a la exposición visual de cientos de variables y cruces que puede incluir una base de datos, el analista de datos tiene una herramienta que le permite extraer y exponer mucho más rápido y con mayor certeza los puntos clave para encontrar cualquier solución(Centric, n.d.).

La visualización de los datos es el término más general que se podría tomar para todos los acontecimientos que en este nivel es construir un conjunto gráfico que represente lo más significativo de los datos clave. El término visualización de la información se aplica generalmente a la representación visual de colecciones a gran escala de información numérica y no numérica, como reportes empresariales, estadísticas de una biblioteca, análisis de comportamientos de los sistemas de movilidad, entre otros más.(Martin, Bergs, Eerdeken, Depaire, & Verelst, 2017)

Ilustración 2 Proceso de Big Data



(Rincón, 2016)

Los datos pueden ser representaciones a variables o atributos cuantificados, los cuales se han originado por una toma de muestras y que luego se realiza un proceso de transformación donde quedan alfanuméricos. Normalmente lo que se busca con estos datos es asociar tendencias estadísticas con dichos datos, que muchas veces es la base informativa para la toma de decisiones de contenidos diversos. Para esto se debe tener algunos aspectos en cuenta:

- Los datos son registros codificados de observaciones de la realidad, son notaciones convencionales sobre lo que se detecta en el estudio de los fenómenos que ocurren en ella.

- En el origen están los datos fijados y cerrados o bien se encuentran activos y sometidos a constante transformación o cambios, los primeros pueden ser históricos y los segundos sujetos a variaciones de la naturaleza o realidad desafiante de cada momento que se vive.
- La forma de obtener los datos no es uniforme, no siempre aparecen con aparatos medidores de los fenómenos, aunque en otras ocasiones son medibles o estimables directamente.
- Los aparatos con que se miden variables no están exentos de defectos y pueden contener errores en sus lecturas incluso con las personas que llevan a cabo la medición.
- A menudo los datos son tomados de forma atribuida y no por variables medibles, por lo que no se pueden considerar con la misma precisión.
- Los datos pueden ser el resultado del análisis de muestras seleccionadas y limitadas no necesariamente de poblaciones enteras, fenómenos u objetos, como sucede en estadísticas y sondeos.
- Los datos pueden tener su origen en la medición o son resultado de transformaciones que cambian su magnitud, tales como los valores logarítmicos que transforman magnitudes geométricas en aritméticas, o todo tipo de traducciones o adaptaciones para que puedan explicarse en el contexto al que son dirigidos. (Valero Sancho, Català Domínguez, & Marín Ochoa, 2014)

La visualización científica se ocupa principalmente de la visualización de 3D donde se hace representaciones realistas de volúmenes, superficies, fuentes de iluminación, entre otros., con un componente dinámico (tiempo).

A través de la historia se han ido desarrollando un conjunto de técnicas la cual su función es transmitir datos, conocimiento e información logrando juntar las partes y así dar la relación y comprensión de estas. El conjunto de técnicas lo integran esquemas, planos, diagramas entre otras (Segel & Heer, 2010). Los primeros ejemplos que se conocen de visualización de datos son los diagramas geométricos, los mapas de navegación y las posiciones de las estrellas y de otros cuerpos celestes.

Crear un gráfico a partir de una tabla de Excel o incluso un SQL como los que se han utilizado hasta hace unos años es bastante sencillo. Las variables apenas pasaban las 100 columnas y no todas estaban relacionadas entre sí. Sin embargo, a medida que se agregan e interrelacionan variables esto se vuelve más complicado. Mucho más complicado (Centric, n.d.).

Para datos estructurados, como ventas, ingresos, entre otros, los gráficos sencillos son más que suficiente. Sin embargo, para los datos desestructurados, como textos o imágenes, captados con sistemas de Big Data, hace falta otro tipo de técnica para visualizar la información adecuadamente(Centric, n.d.).

Con la llegada de la era digital, con el uso de equipos de cómputo y tecnologías de alto rendimiento para visualización, sabemos que la información está creciendo de manera exponencial donde la comunidad científica está realizando proyectos bastante complejos y es requerido una explicación para un entendimiento común de la población. Además, fueron inventados todo tipo de gráficas visuales, como el de barras y circulares, histogramas, gráficos de línea, líneas de tiempo, mapas de contornos, gráficos isotérmicos, cartogramas, mapas de puntos, entre otros.(Kim, Carlis, & Keefe, 2017)

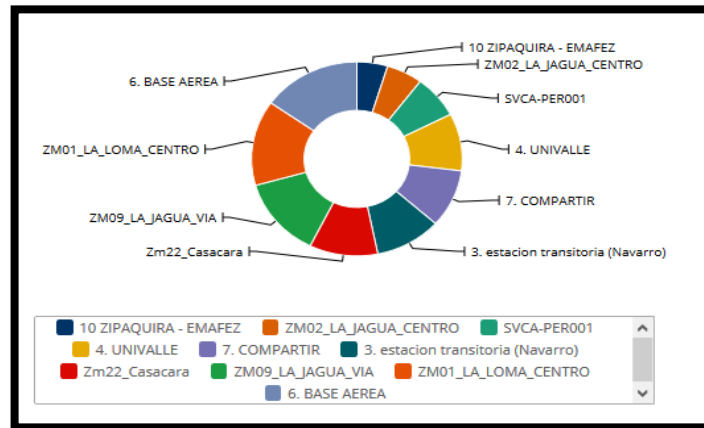
La gestación de nuevas ideas y el desarrollo de las tecnologías permitieron explorar las posibilidades de representar en planos 2D y 3D la información, favoreciendo el desarrollo de gráficos multivariantes(Zoss, 1996).

Muchos países han desarrollado sus estrategias, con dashboard para indicadores nacionales con participaciones de varias instituciones gubernamentales, entidades privadas e instituciones educativas. Como resultado se vuelve una herramienta útil para ver tendencias anteriores, problemas actuales y futuras trayectorias dentro del ámbito económico.(Sardain, Tang, & Potvin, 2016)

La visualización de datos se puede clasificar de la siguiente manera:

- Visualización espacial: Se basa en comparar espacios con interacción de puntos, líneas y zonas con colores que se genera a partir de los datos. Esta se representa por medio de gráficas circulares, barras, radiales, dispersión y burbuja. Es una de las técnicas que más se usa para datos variables como tasas o porcentajes.

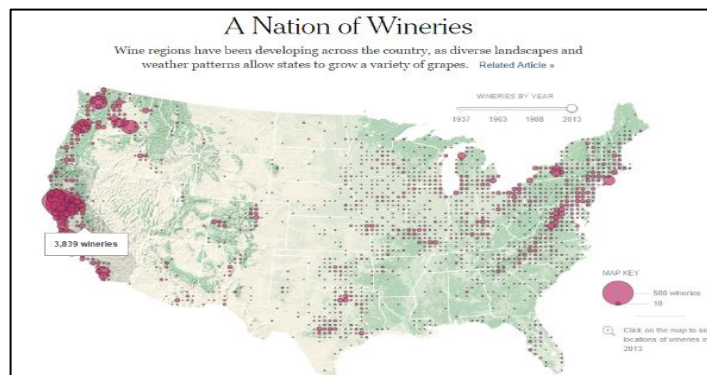
Ilustración 3. Localizaciones de menor concentración de material articulado



(Abiertos, 2016)

- Otra manera de representar la visualización espacial es por medio de nubes de puntos, superficies hasta efectos en 4 dimensiones que incluyen vuelos de cámara sobre objetos.

Ilustración 4. A Nation of Wineries



(Nytimes, 2013)

- Visualización tabular: Se fundamenta en la relación entre los registros de una misma categoría, son usadas en los ámbitos deportivos y políticos como por ejemplo para comprar rivales electorales o rivales en un encuentro de futbol. Por lo general se pueden encontrar acompañando información tipográfica.

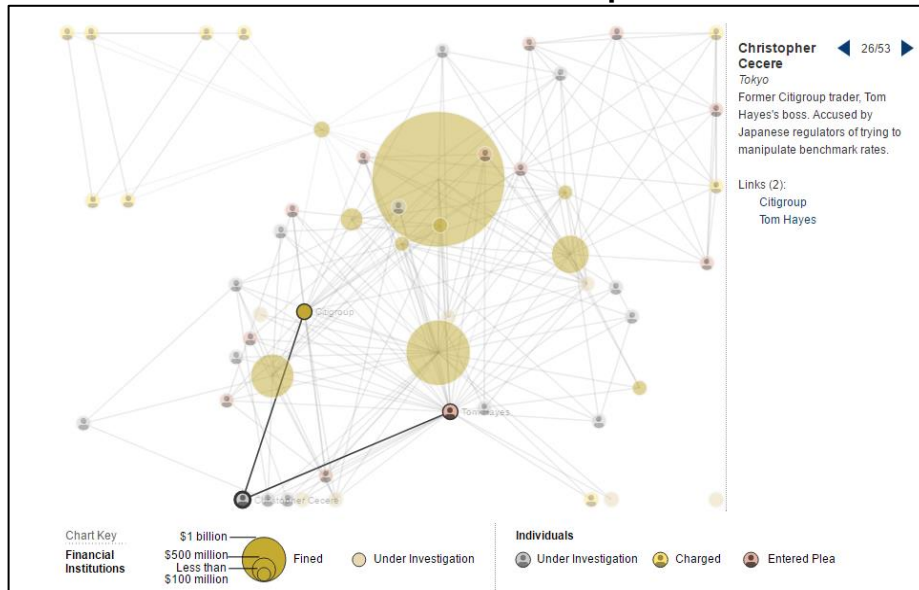
Ilustración 5. Clouds of words



(Auraria Library, 2017)

- Visualización posicional: Es una clasificación que nos permite saber qué relaciones hay entre diferentes objetos u elementos, como por ejemplo los árboles genealógicos, esquemas organizacionales de empresas, diagramas de flujo y líneas de tiempo.

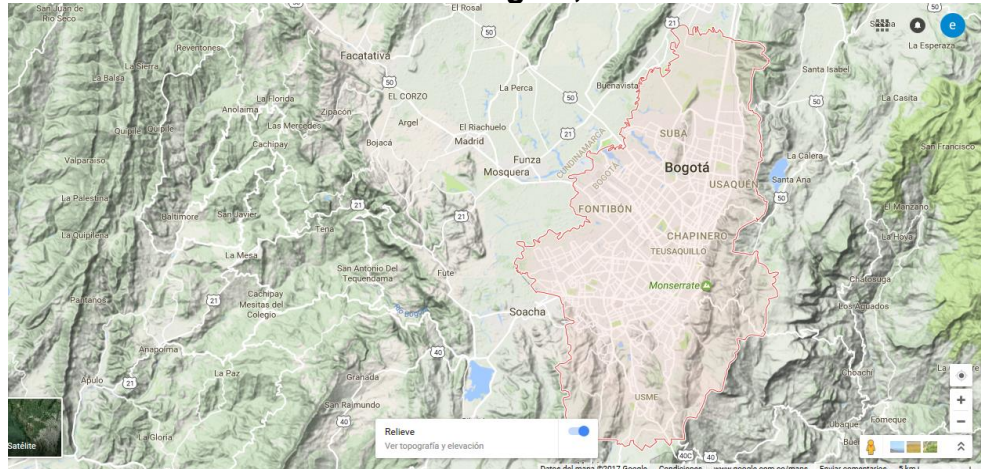
Ilustración 6. Libor: The Spider Network



(Jovi Juan, 2015)

- Visualización topográfica: Es la representación de terrenos como planos con diversas propiedades multiescalares y con multicolores el cual, es una de muchos planos en el estudio de los terrenos.

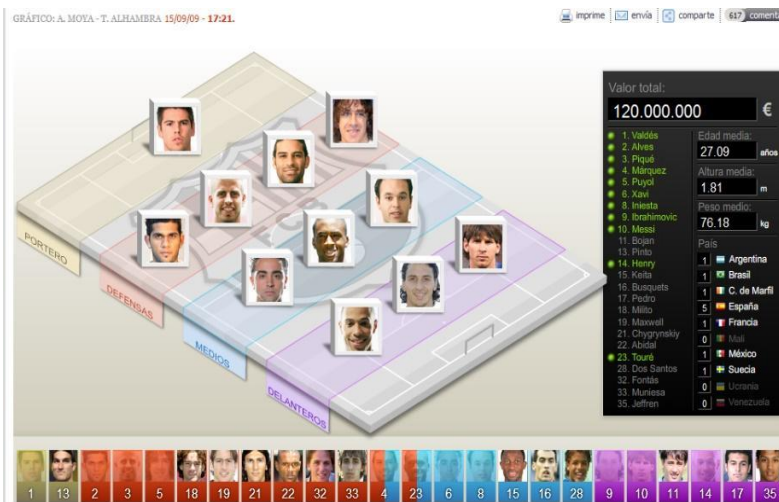
Ilustración 7. Bogotá, Colombia



(Maps, 2017)

- Visualización interactiva: Esta permite al usuario arrastrar diferentes tipos de elementos que le permiten generar una conclusión. Estos pueden ser de diferentes tipos de gráficos (barras, circular, entre otros.) pero también puede estar compuesta de imágenes y acompañadas de cálculos matemáticos o tabulaciones. (Jarabo, Masia, Marco, & Gutierrez, 2016)

Ilustración 8. Confecciona tu Pep Team



(Marca, 2009)

4. MARCO CONCEPTUAL

- Datos Científicos: Es el valor de una variable en una unidad de análisis.
- Big Data: Es el proceso en el que se recolecta grandes cantidades de datos que son almacenados, con los cuales se empieza un análisis para encontrar patrones y correlaciones que deben al final ser visualizados. Se debe tener en cuenta la velocidad de generar información como de analizarla porque debe existir una necesidad de estudiarla además de la veracidad donde el dato debe ser de calidad(MiBloguel, 2017).
- Visualización científica: La visualización científica pretende a través de los datos generar un conjunto gráfico que muestre puntos clave y agrupados que permita entender relaciones o tendencias que llevan a conclusiones en su interpretación. Estos pueden ser modelados en 2D (barras, circular, entre otros.) o en 3D (superficies, objetos)(Valero Sancho et al., 2014)
- Análisis Visual: Este análisis visual está muy relacionado con el análisis de datos ya que de allí es donde parte la correcta visualización de datos. Es la manera más sencilla de mostrar una visualización donde sin necesidad de tener un soporte estadístico formal, los usuarios pueden interactuar con diferentes gráficas y de esta manera sacar sus propias conclusiones(Zoss, 1996).
- Dashboard: Es una representación gráfica de los principales indicadores que intervienen para conseguir objetivos específicos que favorecen la toma de decisiones. Los gráficos deben ser limpios y ordenados que permiten con solo mirarlo entender de qué trata. Debe permitir comparar las diferentes gráficas para que así permita extraer conclusiones útiles. Es una herramienta que es de principal ayuda para las organizaciones.

5. METODOLOGÍA

La metodología propuesta para el desarrollo del componente de visualización toma como base la metodología de desarrollo ágil XP o programación extrema y a través de un conjunto de prácticas, da una aplicación enfocada al desarrollo de manera singular o exclusiva para un solo programador.

Aprovechando sus características de ejecución en el que la clasifican dentro del grupo de metodologías de desarrollo ágiles, las cuales implican métodos de ingeniería del software basados en el desarrollo iterativo e incremental, estas metodologías son necesarias en un mundo dinámico e impredecible como el nuestro, además de plantear un manifiesto ágil el cual según valora o da prioridad

al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas (Es más importante construir un buen equipo que construir el entorno). Además, plantea la importancia de desarrollar software que funcione más que conseguir una buena documentación, La colaboración con el cliente más que la negociación de un contrato y responder a los cambios más que seguir estrictamente un plan(Canós, Letelier, Penadés, & Valencia, 2003).

La programación extrema (XP) tomando como base los conceptos planteados en una metodología de desarrollo de software planteada por Kent Beck en el año (1999). Este tipo de programación se diferencia de las metodologías comunes principalmente en que propone como factor primario la adaptabilidad antes que la previsibilidad. Posee un conjunto de valores que la identifican además de un conjunto de etapas definidas(Fallis, 2013).

- Valores:
 - Simplicidad: La simplicidad es la base de la programación extrema. Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento(Canós et al., 2003).
 - Comunicación: La comunicación se realiza de diferentes formas. Para los programadores el código comunica mejor cuanto más simple sea(Yolanda, 2013).
 - Retroalimentación (feedback): Al estar el cliente integrado en el proyecto, su opinión sobre el estado del proyecto se conoce en tiempo real(Yolanda, 2013).
 - Coraje o valentía: Muchas de las prácticas implican valentía. Una de ellas es siempre diseñar y programar para hoy y no para mañana.

- Fases de la metodología XP
 - Análisis: Esta etapa comprende la definición de historias de usuario con el cliente, las historias tienen una gran similitud a los casos de uso relacionándose con el levantamiento de requerimientos, pero con algunas diferencias, estas historias son líneas escritas por el cliente en lenguaje no técnico y son usadas para la estimación de tiempos. Continuo a esto se elabora un "Release plan" (planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario), en seguida se definen las iteraciones y velocidad del proyecto.(Yolanda, 2013)

- Diseño: Consta del planteamiento de diseños simples y sencillos, pero de calidad, también promueve la utilización de glosarios de términos para facilitar el entendimiento del diseño y abrir paso a la reutilización de código (Yolanda, 2013) por ejemplo en posteriores actualizaciones, adicionalmente, plantea el tratamiento de riesgo para su reducción de forma rápida, el descarte de la funcionalidad extra y la refactorización de código sin alterar su funcionalidad con el fin de optimizar el funcionamiento.
- Codificación: Esta fase comprende la programación o codificación de las historias donde la presencia del cliente se hace aún más esencial, esta programación debe hacerse controlada por un estándar definido esto mantiene el código consistente y facilita su comprensión y escalabilidad.
- Pruebas: Como fase final, implica el testeado del código para verificar su funcionalidad, sometiendo a estas pruebas distintas clases del sistema omitiendo los métodos más triviales, estos test no deben tener ninguna dependencia del código que en un futuro evaluará.

El control de versiones hace parte de esta metodología, existen diferentes herramientas que permiten llevar este control almacenando los diferentes archivos o carpetas de forma histórica, de manera que cualquier persona involucrada en el desarrollo pueda recuperar versiones anteriores de estos documentos. Adicionalmente estas herramientas permiten llevar un control de los diferentes cambios que se realicen en cada sincronización a través de comentarios generados y la identificación de la persona que realizó el cambio. En el proyecto actual se lleva un control de versiones a través de la plataforma GitHub, la cual es una plataforma de desarrollo colaborativo de software que nos permite subir nuestros proyectos o codificaciones utilizando el sistema de control de versiones Git este último creado por el ingeniero Linus Torvalds en el 2005 (GIT, n.d.).

Como apoyo a la fase de desarrollo del proyecto se ha creado un repositorio en GitHub. En el repositorio se han realizado los diferentes commits que permite el seguimiento al desarrollo realizado para el componente de visualización de datos científicos.

En la presentación del concepto de PXP, donde se puede identificar también un conjunto comparativo de doce prácticas propias de la metodología XP y como varían para la aplicación de PXP (Joskowicz & Mingus, 2008). Básicamente hacen referencia al juego de roles que debe realizar el desarrollador, al igual que el orden de las diferentes tareas diarias de desarrollo. Una de las ventajas frente a XP es la independencia a la hora de tomar decisiones estructurales en el desarrollo, la simpleza en el diseño incrementa sin afectar la funcionalidad del software y también

permite una concreta planificación y control de tiempos durante el proceso general(Barriga Mariño, 2017).

Por su funcionalidad, condiciones, ventajas y técnicas de implementación el concepto de PXP como metodología ágil fue implementado durante el desarrollo del proyecto y se promueve su aplicación en posteriores desarrollos con las características apropiadas para su aplicación

Entre las fases de la metodología, se comprende en la planeación el proceso de obtención y especificación de requerimientos, fue implementada para este proyecto de una forma en la cual se define la metáfora del sistema, se crean las historias de usuario y se plantean los requisitos funcionales y no funcionales de la herramienta, definiciones que se pueden encontrar en el documento de especificación de requerimientos de software (Véase el [Anexo 1](#)).

6. DESARROLLO DEL PROYECTO

En este ítem se realiza el desarrollo del trabajo de grado, utilizando la metodología mencionada en el ítem anterior y herramientas que apoyan la elaboración del mismo, comenzando por la fase de análisis hasta la terminación del mismo.

6.1. ANÁLISIS

El desarrollo del proyecto inicia a través de una fase de análisis, en la cual el objetivo es indagar, comparar y aprender sobre diferentes conceptos que pudieran ser necesarios en la implementación del componente web parametrizable para la visualización de datos científicos, conceptos referentes al diseño de aplicaciones web, herramientas, métodos y técnicas de visualización de datos postularon como los temas principales de investigación.

Más allá de ser un actor secundario en aplicaciones de análisis, la visualización de datos llena varios papeles cruciales en todo el proceso análisis de información. Desde la exploración de datos inicial hasta el desarrollo de modelos de predicción para informar sobre los resultados analíticos producidos por los modelos, las técnicas de visualización de datos y el software son componentes clave para tener una mejor comprensión de los datos(Noblejas & Rodríguez, 2014).

En el común, existen analistas de información que se apoyan en herramientas de software para realizar una exploración y visualización de sets de datos, como, por ejemplo, Tableau que es una herramienta muy completa y de fácil interacción para el usuario, pero tiene como limitante el elevado costo de la licencia. Otras herramientas de software que enlazan por medio de componentes de visualización integrados en software de analítica más sofisticado, como SAS, R. También, algunos analistas que no utiliza ningún software especializado optan por realizar la visualización de datos directamente en hojas de cálculo de Excel.

Para la fase de investigación se realiza búsquedas en bases de datos científicas, bibliotecas virtuales y diferentes recursos web, luego de haber identificado los conceptos importantes de la investigación y sus palabras clave como por ejemplo taxonomías de visualización(Chi, 2015).

Consecutivamente en la fase de análisis, en el cual se detallan las bases del proyecto y se delimita el camino correcto que se debe seguir para la respectiva implementación, en el análisis, se tuvieron los posibles casos y procesos requeridos para desarrollar el proyecto, encolando la investigación realizada hasta el momento.

Como parte primordial de la fase de análisis, se lleva a cabo un levantamiento de requerimientos para el desarrollo del componente de visualización. Por lo tanto, antes de definir los requerimientos de la herramienta, era necesario definir la metodología de desarrollo a seguir, esto debido a la necesidad de establecer la mejor manera de realizar el proyecto desde la planificación a la fase de pruebas, como lo vimos en el marco teórico y en la definición metodológica.

Detallando, en la fase de análisis, se realiza la planeación y especificaciones de los requerimientos que son necesarios para el desarrollo del componente web parametrizable para la visualización de datos científicos (Véase el [Anexo 1](#)).

Vinculado a la definición de los requerimientos del componente de visualización, se debe definir cuáles son herramientas o componentes que permitirán realizar el desarrollo del proyecto, de igual forma, realizar cuestionamientos como: ¿Qué arquitectura de software es la ideal? ¿Qué lenguaje de programación utilizar en la codificación? ¿Qué tipo de bases de datos deberían implementarse? ¿Cómo estarán relacionados los datos dentro del sistema? Y varias más surgen en esta fase de planificación y se abordan en la siguiente la cual corresponde a la fase de diseño(Barriga Mariño, 2017).

6.2. DISEÑO

Iniciando la fase de diseño, primero se define la arquitectura que mejor se acomoda al componente como se muestra en el documento de arquitectura de software (Véase el [Anexo 3](#)), esta arquitectura se realiza con el apoyo de la herramienta del lenguaje unificado de modelado UML, esta herramienta permite diagramar la arquitectura del sistema mediante diferentes diagramas, los cuales tienen un enfoque específico con respecto a la funcionalidad y estructura.(Object Management Group, 2017)(Valero Sancho et al., 2017)

El diseño de la herramienta se describe el diseño de software del componente web parametrizable para la visualización de datos científicos. Sin embargo, es de ayuda para el desarrollo en los procesos del ciclo de vida y de entrega de software en el cual se nombra las suposiciones y restricciones que se deben tener en cuenta al diseñarlo (Véase el [Anexo 2](#)).

6.3. DESARROLLO (CODIFICACIÓN)

De acuerdo con el análisis de requerimientos (Véase el [Anexo 1](#)) y la arquitectura diseñada (Véase el [Anexo 3](#)) se decide, en forma conjunta con el coordinador, el uso de lenguaje NodeJS, debido a que este lenguaje, bajo el enfoque de ser útil en la creación de programas web, es altamente escalable y es muy conocido por ser una tecnología que permite trabajar con Javascript del lado del servidor (Véase el [Anexo 2](#)). El motor de base de datos con el cual se implementa el proyecto es MySQL, debido a que MySQL es un producto OpenSource, el cual, además de ofrecer velocidad también es un motor que presenta buen rendimiento y facilidad de administración (Véase el [Anexo 2](#)). De acuerdo con la metodología especificada en el numeral 5, se realizó la creación del repositorio en la plataforma GitHub, para tener el control de versiones en el desarrollo del componente.(Enríquez Toledo Alma; Maldonado Ayala Jesús; Nakamura Ortega Yunko; Nogueron Toledo Goretty, 1981)(ALIDA VERGARA, n.d.)

Como primera instancia, en ambiente local, se instala la versión 6.11.2 de NodeJS, el cual se puede descargar desde la página oficial de NodeJS para el respectivo sistema operativo, ya que NodeJS es multiplataforma -Para realizar el proceso de instalación, en la página oficial ofrecen la respectiva documentación-. Seguido de la instalación, es esencial contar con el framework ExpressJS(Herman, 2016), el cual permite desplegar la configuración de un proyecto en NodeJS de forma fácil y rápida. Para la configuración del framework se deben realizar los siguientes pasos:

- Ubicándose en el directorio o folder del aplicativo, se debe ejecutar el comando '**npm install express-generator -g**' que permite la instalación del framework ExpressJS.
- Con el comando '**express nombre_proyecto**' se realiza la creación y configuración del proyecto con sus respectivas dependencias y directorios.

Ilustración 9. Configuración del proyecto con ExpressJS

```
C:\Users\cpineda\purocodigo>express proyecto

warning: the default view engine will not be jade in future releases
warning: use '--view=jade' or '--help' for additional options

create : proyecto
create : proyecto/package.json
create : proyecto/app.js
create : proyecto/views
create : proyecto/views/index.jade
create : proyecto/views/layout.jade
create : proyecto/views/error.jade
create : proyecto/routes
create : proyecto/routes/index.js
create : proyecto/routes/users.js
create : proyecto/public
create : proyecto/bin
create : proyecto/bin/www
create : proyecto/public/images
create : proyecto/public/stylesheets
create : proyecto/public/stylesheets/style.css

install dependencies:
  > cd proyecto && npm install

run the app:
  > SET DEBUG=proyecto:* & npm start

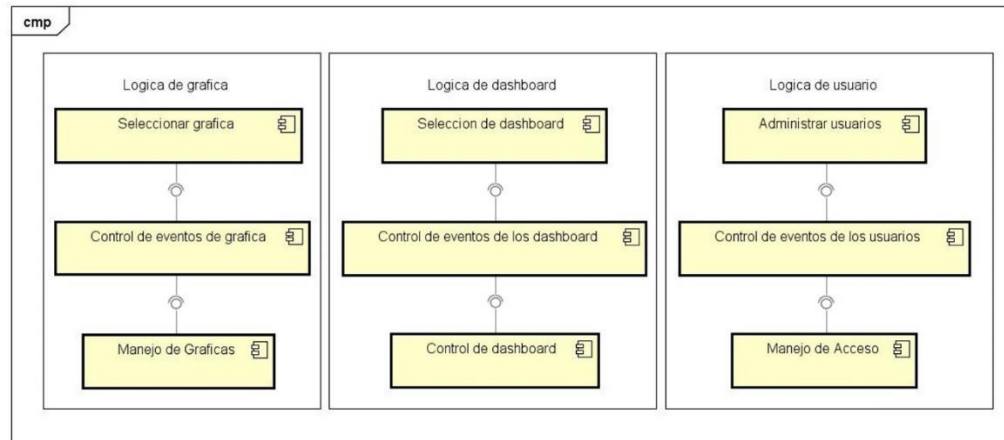
create : proyecto/public/javascripts

C:\Users\cpineda\purocodigo>_
```

Express por defecto instala y configura el manejador de plantillas JADE, el cual permite la interacción entre NodeJS y HTML, enviando parámetros a la plantilla web.

Como lineamiento principal del proyecto, es poder proveer, una herramienta capaz de facilitar la visualización de cualquier set de datos que tenga una estructura definida, acorde a lo determinado por la coordinación del proyecto. Y, conforme se especifica en la arquitectura de software (Véase el [Anexo 3](#)), se evidencia los elementos principales del componente de visualización de datos científicos, los cuales son lógica del usuario, lógica de dashboard y lógica de gráfica.

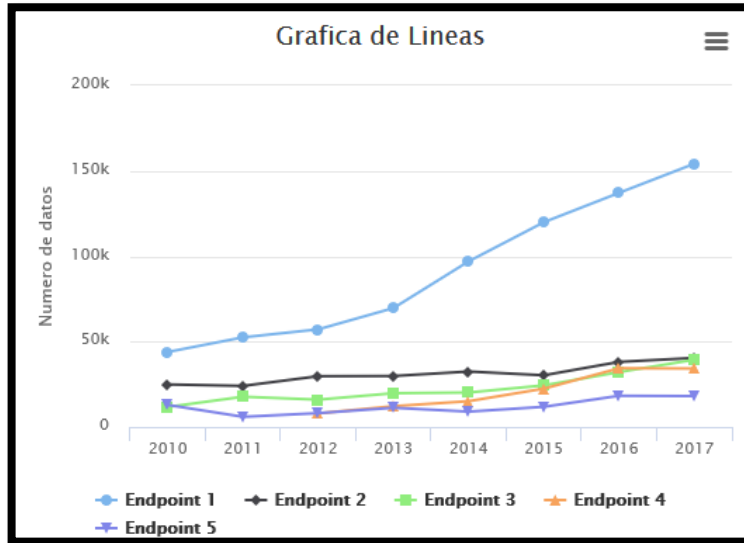
Ilustración 10. Diagrama de Componentes



Teniendo en cuenta lo anterior, se elaboró un aplicativo web responsive, el cual permite a un usuario crear diferentes dashboard; cada dashboard puede ser visualizado en 3 tipos de layout y ubicar una gráfica determina en una posición específica del layout seleccionado. Las gráficas que se pueden visualizar en el componente con una cantidad de set de datos son:

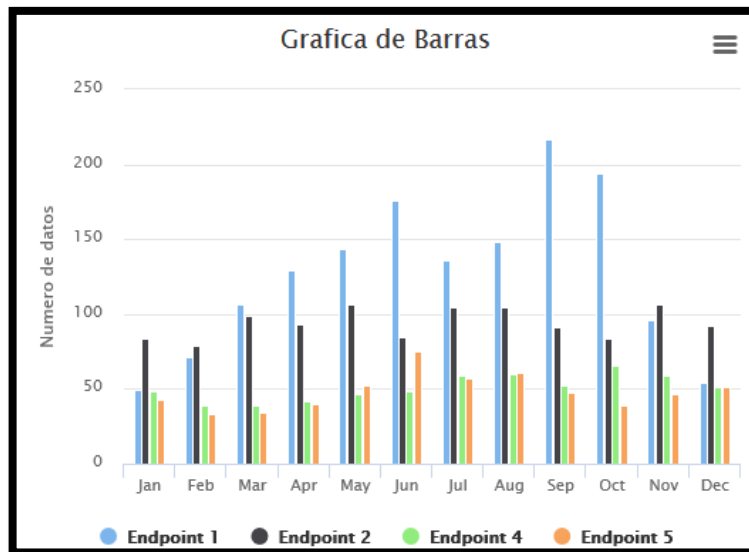
- Grafica de Línea, puede tener un máximo de 6 endpoints; cada endpoint está conformado por un set de datos con una estructura especifica la cual es compartida por los endpoint que se pretende visualizar.

Ilustración 11. Grafica de líneas



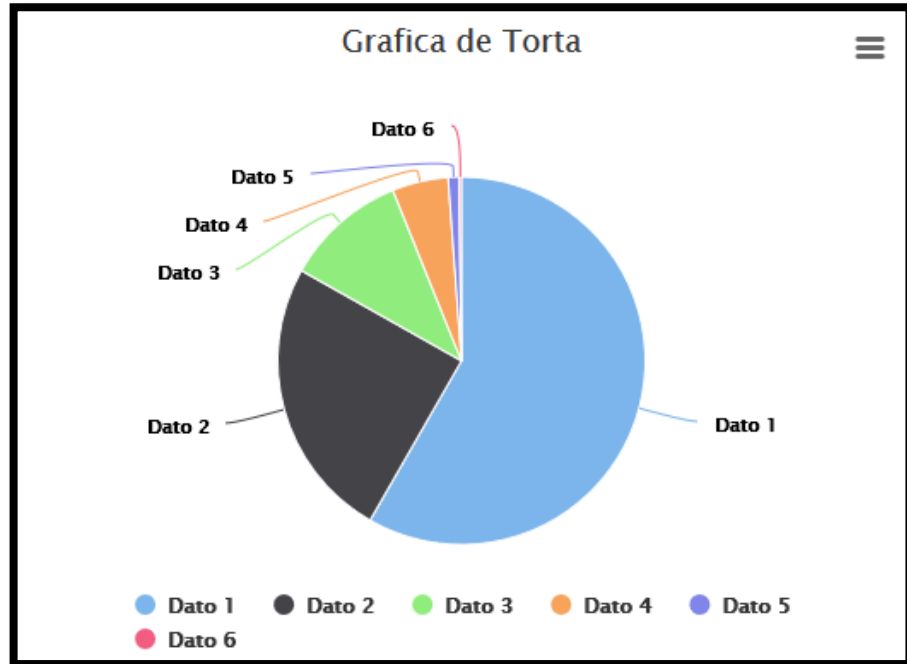
- Grafica de Barras, puede tener un máximo de 4 endpoints; cada endpoint está conformado por un set de datos con una estructura especifica la cual es compartida por los endpoint que se pretende visualizar.

Ilustración 12. Grafica de barras



- Grafica de Torta, puede tener un solo endpoint asociado, debido que este tipo de grafica cumple con la finalidad de visualizar el porcentaje con respecto a un total.

Ilustración 13. Grafica de torta



Los gráficos obtendrán los sets de datos por medio de la comunicación de webservices de tipo Restful, los cuales serán adicionados por el usuario que parametriza la gráfica. Los usuarios pueden customizar las gráficas adicionando información como título, subtítulo, título del eje Y para los tipos de graficas barras y líneas. Los colores de los gráficos serán estándar, es decir esta propiedad no será modificable por el usuario.

7. RESULTADOS

Parte vital del proceso de desarrollo de un proyecto de software esta la fase de pruebas, en esta fase se determina el comportamiento de las funciones en relación con el correcto funcionamiento y el rendimiento óptimo del componente de visualización de datos científicos.

En esta fase se determina el cumplimiento de los objetivos propuestos que enmarcan al proyecto, y su respectiva ejecución sea satisfactoria. Las pruebas tienen un valor importante en la representación del proyecto ya que estas definen el flujo correcto en la codificación y/o desarrollo del mismo; luego de realizar las pruebas necesarias, se generarán las evidencias correspondientes y se informa sobre los resultados obtenidos.

Para la fase de pruebas, se aplica el concepto de pruebas unitarias o Unit Testing; las pruebas unitarias tienen como característica probar componentes de software de forma aislada, consintiendo que la calidad sea satisfactoria para cada componente de software (Departamento de Lenguajes y Sistemas Informáticos. Escuela Superior de Ingeniería Informática, n.d.; Probar, Unitarias, & Test, 2006; Victor Tejada Yau, 2013) y este se comporte de la manera que corresponde frente a diversas situaciones (Victor Tejada Yau, 2013).

La implementación de las pruebas unitarias en NodeJS, se realiza mediante el framework Mocha, el cual se le puede definir como un framework de pruebas rico en características que puede ser implementado en NodeJS o por medio de un navegador; Mocha (Rafael Márquez, 2017; Zaza, 2016) ejecuta las pruebas en serie permitiendo reportes flexibles y exactos, haciendo que el código del componente de software sea óptimo. Como principales características Mocha posee soporte para diferentes navegadores, soporte de Debugger para nodejs (Victor Tejada Yau, 2013).

Para realizar la instalación de Mocha, se debe utilizar el gestor de paquetes de NodeJS (NPM), por línea de comando ejecutar la instrucción 'npm install -g mocha' (Herman, 2016); una vez instalado, se puede realizar las respectivas pruebas unitarias en el proyecto.

Una vez instalado Mocha, se debe realizar la creación del directorio llamado 'test', el cual contendrá el archivo test.js. En el archivo test.js se debe realizar la configuración de las pruebas unitarias, en este caso, se realizan las pruebas al controlador principal que genera los siguientes casos:

Ilustración 14. Caso Pruebas Unitarias

```
1 var assert = require('assert');
2 var controllers = require('../controllers');
3
4 describe('Dashboard', function(){
5   describe('Module Dashboard', function(){
6     it('Run dashboardController', function(){
7       assert.equal(typeof controllers.dashboardController, 'object');
8     });
9
10    it('Run method getLoadDashboardUser', function(){
11      assert.equal(typeof controllers.dashboardController.getLoadDashboardUser, 'function');
12    });
13
14    it('Run method WebService', function(){
15      assert.equal(typeof controllers.webserviceController.getWebService, 'function');
16    });
17
18    it('load render index', function(){
19      assert.equal(typeof controllers.homecontroller.index, 'function');
20    });
21  });
22 });
```

Para realizar la ejecución de las pruebas unitarias, se debe ubicar en la raíz del proyecto, luego, se debe ejecutar el comando mocha, el comando anterior ejecuta

el ambiente de pruebas de los casos planteados como pruebas unitarias. El resultado de las pruebas son las siguientes.

Ilustración 15. Resultado Pruebas Unitarias

```
PS C:\repositorioNODE\ComponenteVisualizacion\Visualizacion> mocha

Dashboard
  Module Dashboard
    ✓ Run dashboardController
    ✓ Run method getLoadDashboardUser
    ✓ Run method WebService
    ✓ load render index

4 passing (16ms)

PS C:\repositorioNODE\ComponenteVisualizacion\Visualizacion>
```

Como se muestra en el resultado del test, se evidencia que las pruebas unitarias se ejecutaron de manera óptima, garantizando el correcto funcionamiento del componente de visualización de datos científicos.

7.1. ANÁLISIS DE RESULTADOS

De acuerdo con el resultado de las pruebas unitarias realizadas al componente de visualización de datos científicos, se establecieron pruebas mecánicas ejecutando componente por componente siguiendo el respectivo flujo del aplicativo, dando prioridad a la visualización de los sets de datos parametrizados en las gráficas de líneas, barras y tortas. Para la elaboración de esta prueba, se realizó la construcción de un webservice de tipo Restful, el cual se realizaba diferentes parámetros para emular el respectivo dinamismo de la aplicación.

El proceso de realización de las pruebas mecánicas se realiza con la herramienta postman, la cual permite visualizar los Endpoints de pruebas que se desarrollaron para las pruebas de ejecución del aplicativo. Se realiza la ejecución de los Endpoint en relación con las gráficas de torta, líneas y barras.

Se relaciona las pruebas de ejecución para la visualización de la gráfica de torta, mostrando los valores del Endpoint seguido de la visualización de la gráfica.

Ilustración 16. Endpoint grafica de torta

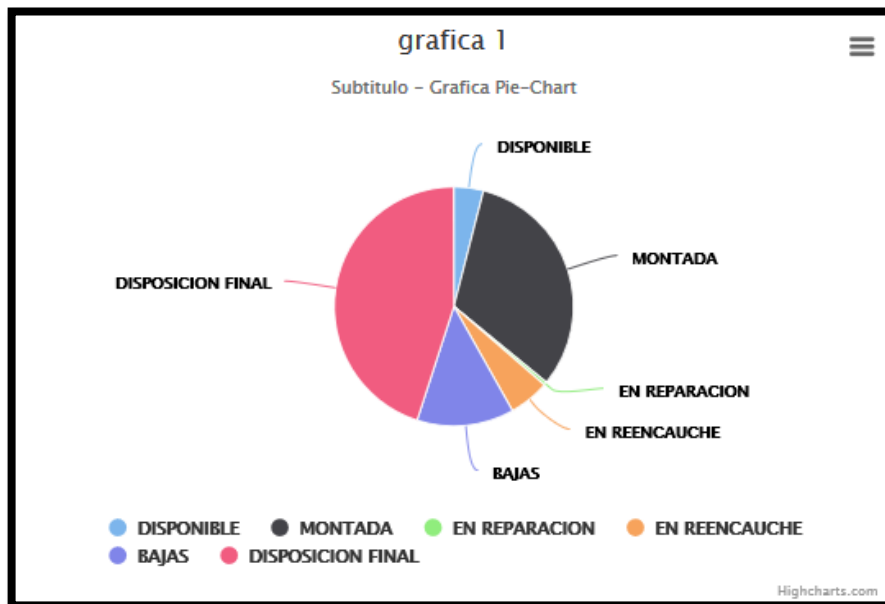
```
GET http://ctw.automundial.com.co/ctw/reporteador2.php?ACC=STATUS

Body Cookies Headers (10) Test Results

Pretty Raw Preview JSON

1 {
2   "name": "LLANTAS",
3   "output": [
4     {
5       "name": "DISPONIBLE",
6       "value": "501"
7     },
8     {
9       "name": "MONTADA",
10      "value": "4012"
11     },
12     {
13      "name": "EN REPARACION",
14      "value": "55"
15     },
16     {
17      "name": "EN REENCAUCHE",
18      "value": "688"
19     },
20     {
21      "name": "BAJAS",
22      "value": "1648"
23     },
24     {
25      "name": "DISPOSICION FINAL",
26      "value": "5654"
27     }
28   ]
29 }
```

Ilustración 17. Visualización grafica de torta



Para la ejecución del gráfico de barras y líneas es importante relacionar los endpoint que se quiere visualizar, de igual forma se debe consumir un endpoint adicional que representa la categoría del eje X. Cabe resaltar que los endpoint deben tener el mismo contexto y orden de generación de los datos.

Ilustración 18. Endpoint de categorías, Eje X

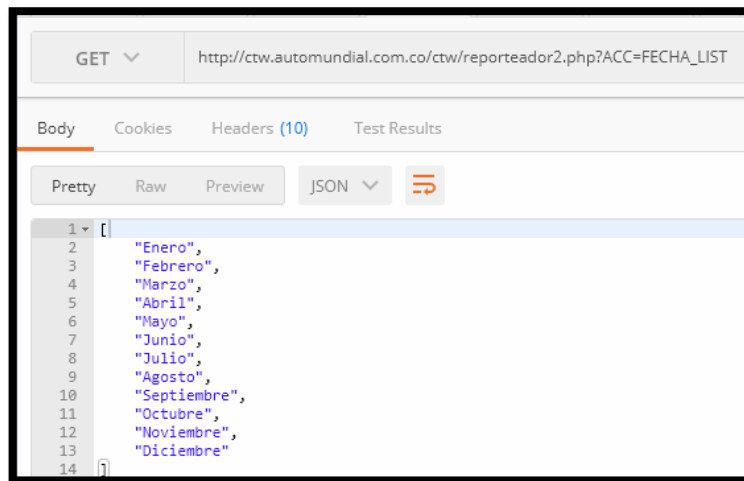


Ilustración 19. Endpoint de grafico de barras y líneas

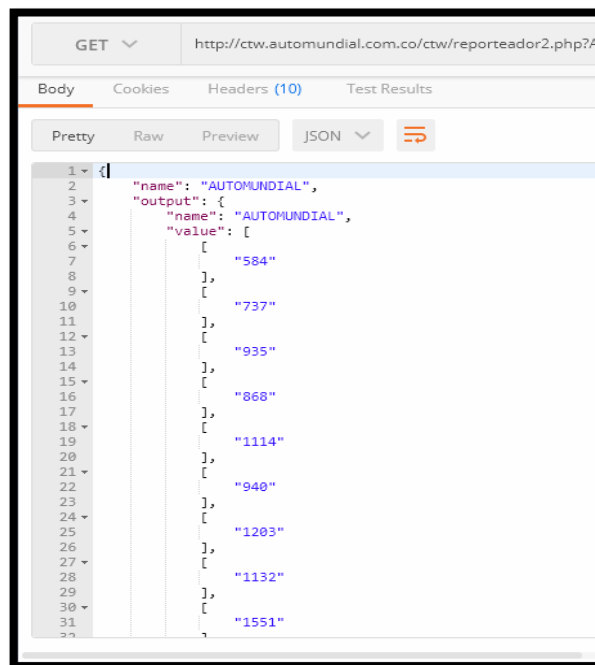


Ilustración 20. Visualización de la gráfica de barra.

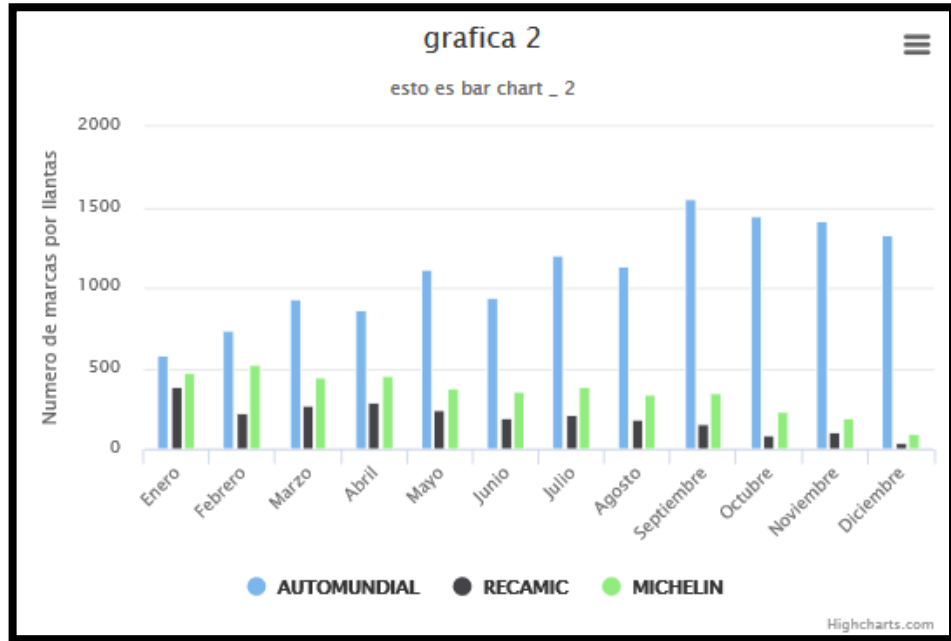


Ilustración 21. Visualización de la gráfica de línea.

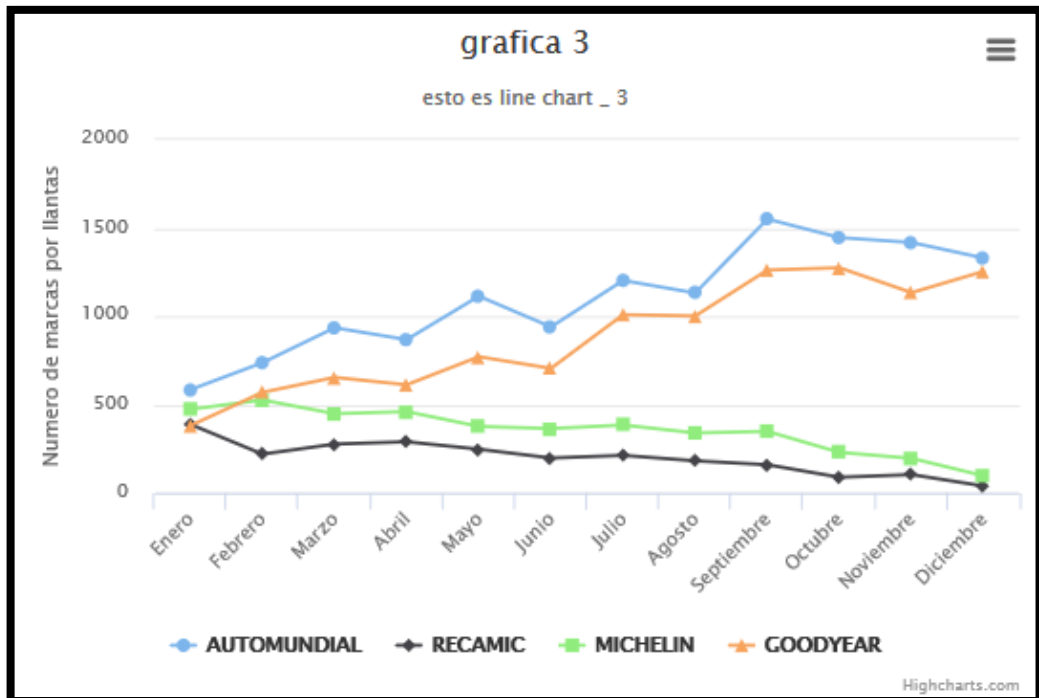
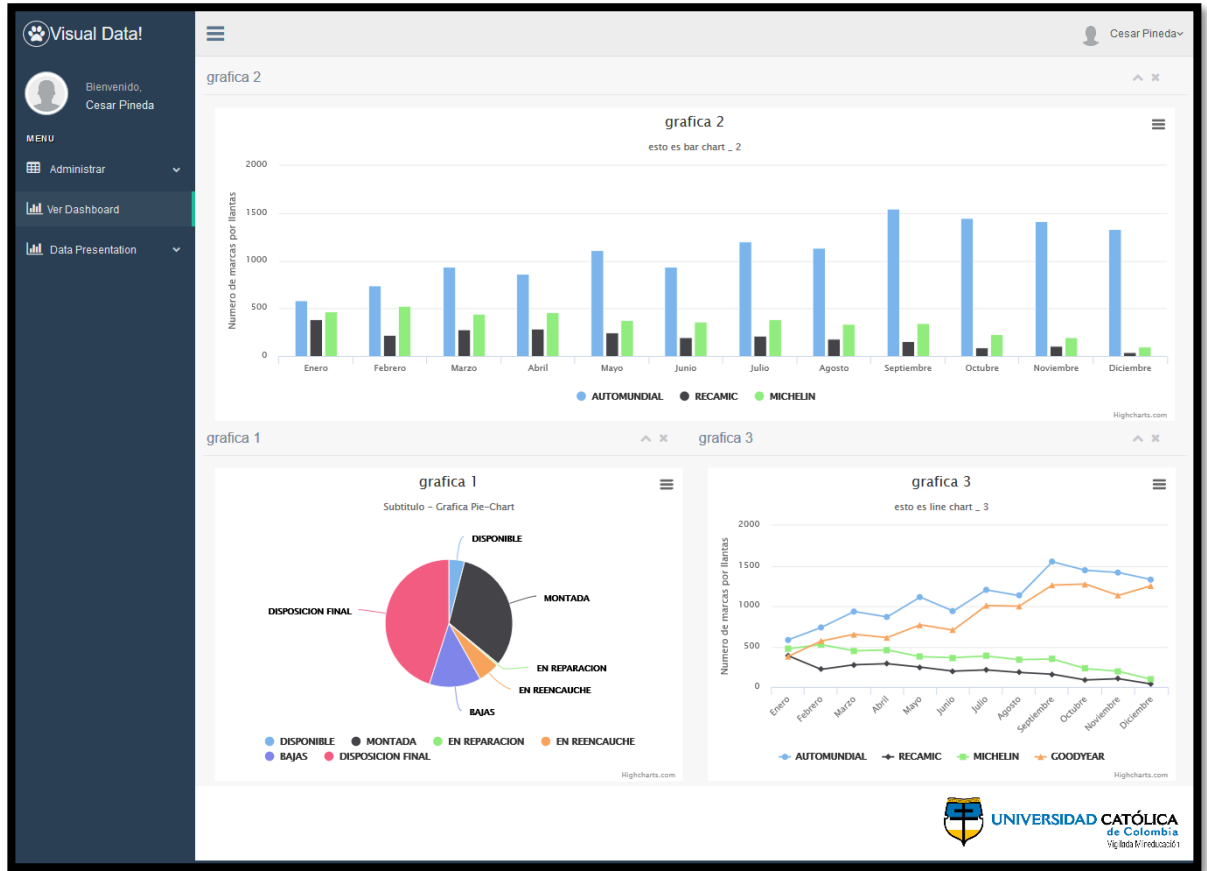


Ilustración 22. Componente de Visualización de datos científicos



El resultado del componente de visualización de datos científicos presenta el propósito de los objetivos específicos planteados, los cuales han sido cumplidos a cabalidad en el proceso de desarrollo y pruebas realizadas.

8. CONCLUSIONES

Al comienzo de la estructuración del proyecto se tenía pensado ofrecer un componente de visualización para el proyecto de investigación de la facultad de psicología, pero analizando los requerimientos del proyecto, se logró entender que la herramienta puede ofrecer la utilidad de visualizar cualquier conjunto de set de datos que se puedan parametrizar para cada una de las gráficas establecidas, siguiendo un respectivo estándar, logrando de esta forma un modelo para la visualización de un proceso de Big Data.

Este componente de visualización de datos científicos ofrece la posibilidad de realizar análisis de forma visual, con una herramienta desarrollada por estudiantes de pregrado, que a pesar de solo ofrecer 3 tipos de graficas las cuales fueron previamente analizadas en sus taxonomías donde se comprendió la complejidad que tienen cada una para tomar los datos y visualizarlos, puede competir a largo plazo con herramientas comerciales, que no son de fácil acceso por sus costos elevados y lograr de esta forma la posibilidad de crecimiento del componente de visualización.

Aplicando una metodología de desarrollo como lo fue XP (Programación Extrema), con la cual se siguió el lineamiento de un proceso de desarrollo de software como lo es el ciclo de investigación, planificación, diseño, desarrollo y pruebas, esto ayudo a generar una gran cantidad de información y poder avanzar de una manera más rápida para poder finalizar el desarrollo del proyecto.

La formación desarrollada durante todas las fases del proyecto muestra las diferentes capacidades que se adquieren analizando las dimensiones del componente, donde se afronta nuevos lenguajes de programación, variedad de métodos de autoaprendizaje y herramientas para lograr el objetivo estipulado, con lo cual muestra el verdadero entorno en el que nos desempeñaremos como ingenieros.

9. TRABAJOS FUTUROS

De acuerdo a los resultados que se obtuvieron y el alcance que se define para este proyecto, se espera las siguientes actividades a futuro:

- Aplicar complementos que permitan la visualización en tiempo real de la información presentada.
- Aumentar el uso de más tipos de gráficas para los dashboard del componente de visualización.
- Realizar un módulo de administración de usuarios con más opciones que permitan interactuar con otros componentes.

10. BIBLIOGRAFÍA

- Abdelsadek, Y., Chelghoum, K., Herrmann, F., Kacem, I., & Otjacques, B. (2018). Community extraction and visualization in social networks applied to Twitter. *Information Sciences*, 424, 204–223. <https://doi.org/10.1016/j.ins.2017.09.022>
- Abiertos, D. (2016). Ambiente y desarrollo sostenible.
- ALIDA VERGARA. (n.d.). 5 ventajas de usar Node.js. Retrieved from <https://www.facilcloud.com/noticias/5-ventajas-de-usar-node-js/>
- Arbib, M. A., & Manes, E. G. (1982). Parametrized data types do not need highly constrained parameters. *Information and Control*, 52(2), 139–158. [https://doi.org/10.1016/S0019-9958\(82\)80026-0](https://doi.org/10.1016/S0019-9958(82)80026-0)
- Auraria Library. (2017). Data Visualization: Quantitative vs. Qualitative.
- Barrientos, P. A. (2014). Enfoque para pruebas de unidad basado en la generación aleatoria de objetos.
- Barriga Mariño, J. C. (2017). *Desarrollo y aplicación de una herramienta de extracción y almacenamiento de datos de twitter a un contexto social de violencia política*. Universidad Católica de Colombia.
- Canós, J. H., Letelier, P., Penadés, C., & Valencia, D. P. De. (2003). Metodologías Ágiles en el Desarrollo de Software. *Development*, 1–8.
- Centric, D. (n.d.). ¿Qué es el Data Visualization o cómo interpretar grandes cantidades de datos? Retrieved from <http://www.datacentric.es/blog/geomarketing/data-visualization-analisis-datos/>
- Chen, X., & Jin, R. (2017). Statistical modeling for visualization evaluation through data fusion. *Applied Ergonomics*, 65, 551–561. <https://doi.org/10.1016/j.apergo.2016.12.016>
- Chi, E. H. (2015). A taxonomy of visualization techniques using the data state reference model. *IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings*, (January 2000), 69–75. <https://doi.org/10.1109/INFVIS.2000.885092>

- Culturación. (n.d.). ¿Qué es y para qué sirve un web service? - Culturación. Retrieved November 18, 2017, from <http://culturacion.com/que-es-y-para-que-sirve-un-web-service/>
- Damian Wajser. (2015). Rest - Restful: Ventajas y diferencias. Retrieved from <http://latamdigital.softtek.co/rest-restful-ventajas-y-diferencias>
- Departamento de Lenguajes y Sistemas Informáticos. Escuela Superior de Ingeniería Informática. (n.d.). Pruebas Unitarias.
- Domo, I. (2017). Domo.
- Enríquez Toledo Alma; Maldonado Ayala Jesús; Nakamura Ortega Yunko; Nogueron Toledo Goretty. (1981). MySQL, 1–2.
- Fallis, A. . (2013). Metodología Actual Metodología XP. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699. <https://doi.org/10.1017/CBO9781107415324.004>
- Franklin, A., Gantela, S., Shifarrow, S., Johnson, T. R., Robinson, D. J., King, B. R., ... Okafor, N. G. (2017). Dashboard visualizations: Supporting real-time throughput decision-making. *Journal of Biomedical Informatics*, 71, 211–221. <https://doi.org/10.1016/j.jbi.2017.05.024>
- GIT. (n.d.). Empezando - Una breve historia de Git. Retrieved from <https://git-scm.com/book/es/v1/Empezando-Una-breve-historia-de-Git>
- Git - Trabajando con repositorios remotos. (n.d.). Retrieved November 18, 2017, from <https://git-scm.com/book/es/v1/Fundamentos-de-Git-Trabajando-con-repositorios-remotos>
- Herman, M. (2016). Testing Node and Express. Retrieved from <http://mherman.org/blog/2016/09/12/testing-node-and-express/#.Wg7zPkribDd>
- Introducción a JSON. (n.d.). Retrieved November 18, 2017, from <https://www.json.org/json-es.html>
- Introducción a los Servicios Web. Invocación de servicios web SOAP. (2014). Retrieved November 18, 2017, from <http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/sesion01-apuntes.html>
- Jarabo, A., Masia, B., Marco, J., & Gutierrez, D. (2016). Recent Advances in Transient Imaging: A Computer Graphics and Vision Perspective. *Visual Informatics*, 1(1), 65–79. <https://doi.org/10.1016/j.visinf.2017.01.008>
- Joskowicz, I. J., & Mingus, C. (2008). Reglas y Prácticas en eXtreme Programming, 1–22.
- Jovi Juan, D. E. W. S. J. (2015). Libor: The Spider Network.
- Kim, K., Carlis, J. V., & Keefe, D. F. (2017). Comparison techniques utilized in spatial 3D and 4D data visualizations: A survey and future directions. *Computers and Graphics (Pergamon)*, 67, 138–147. <https://doi.org/10.1016/j.cag.2017.05.005>
- Lappo, P. (2005). Extreme Programming for Solo Projects. *BCS Sussex*.
- Leimer, K., Gersthofer, L., Wimmer, M., & Musialski, P. (2017). Relation-based parametrization and exploration of shape collections. *Computers and Graphics (Pergamon)*, 67, 127–137. <https://doi.org/10.1016/j.cag.2017.07.001>
- Maps, G. (2017). Bogota, Colombia.
- Marca. (2009). Confecciona tu Pep Team.
- Martin, N., Bergs, J., Eerdeken, D., Depaire, B., & Verelst, S. (2017). Developing

- an emergency department crowding dashboard: A design science approach. *International Emergency Nursing*, 1–9. <https://doi.org/10.1016/j.ienj.2017.08.001>
- McKinsey & Company. (2011). Big data: The next frontier for innovation, competition, and productivity. *McKinsey Global Institute*, (June), 156. <https://doi.org/10.1080/01443610903114527>
- MiBloguel. (2017). Big Data, significado y su utilidad en la sociedad. Retrieved from <https://mibloguel.com/big-data-significado-y-su-utilidad-en-la-sociedad/>
- Microsoft. (2017). Power BI.
- Naur, P., & Randell, B. (1968). Software Engineering: Report of a Conference Sponsored by the NATO Science Committee. *NATO Software Engineering Conference*, (October 1968), 231. <https://doi.org/10.1093/bib/bbp050>
- Noblejas, C. J., & Rodríguez, A. P. (2014). Recuperación y visualización de información en web of science y scopus: Una aproximación práctica. *Investigacion Bibliotecologica*, 28(64), 15–31. [https://doi.org/10.1016/S0187-358X\(14\)70907-4](https://doi.org/10.1016/S0187-358X(14)70907-4)
- Nytimes, N. Y. T. (2013). A Nation of Wineries.
- Object Management Group, I. (2017). What is UML. Retrieved from <http://www.uml.org/what-is-uml.htm>
- photo, P. (n.d.). Title Data Visualization Techniques From Basics to Big Data With SAS® Visual Analytics.
- Probar, J. R., Unitarias, P., & Test, U. (2006). Pruebas unitarias.
- Pruebas REST con Postman Jetpacks. (n.d.). Retrieved November 18, 2017, from <https://profesores.virtual.uniandes.edu.co/~isis2603/dokuwiki/doku.php?id=tutoriales:postman>
- Rafael Márquez. (2017). Testeando JavaScript con Mocha y Chai. Retrieved from <https://www.paradigmadigital.com/dev/testeando-javascript-mocha-chai/>
- Rincón, D. (2016). [UCC] 1. Introducción a BD, (1), 1–10.
- Sacha, D., Sedlmair, M., Zhang, L., Lee, J. A., Peltonen, J., Weiskopf, D., ... Keim, D. A. (2017). What you see is what you can change: Human-centered machine learning by interactive visualization. *Neurocomputing*, 268, 164–175. <https://doi.org/10.1016/j.neucom.2017.01.105>
- Sardain, A., Tang, C., & Potvin, C. (2016). Towards a dashboard of sustainability indicators for Panama: A participatory approach. *Ecological Indicators*, 70, 545–556. <https://doi.org/10.1016/j.ecolind.2016.06.038>
- Segel, E., & Heer, J. (2010). Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6), 1139–1148. <https://doi.org/10.1109/TVCG.2010.179>
- Sood, A., Sinha, N., Dewjee, S., & Zhao, W. (2013). Tableau Tutorial.
- Tableau. (2017). Tableau.
- Tennekes, M. (2013). Visualizing and Inspecting Large Datasets with Tableplots. *Journal of Data Science*, 11, 43–58.
- Valero Sancho, J. L., Català Domínguez, J., & Marín Ochoa, B. E. (2014). Aproximación a una taxonomía de la visualización de datos. *Revista Latina de Comunicación Social*. <https://doi.org/10.4185/RLCS-2014-1021>

- Valero Sancho, J. L., Català Domínguez, J., Marín Ochoa, B. E., Canós, J. H., Letelier, P., Penadés, C., ... IEEE Computer Society. (2017). Big Data, significado y su utilidad en la sociedad. <https://doi.org/10.4185/RLCS-2014-1021>
- Victor Tejada Yau. (2013). Mocha, un nuevo sabor en pruebas unitarias de Javascript. Retrieved from <https://desarrolloweb.com/articulos/mocha-pruebas-unitarias-javascript.html>
- Vilarinho, S., Lopes, I., & Sousa, S. (2017). Design Procedure to Develop Dashboards Aimed at Improving the Performance of Productive Equipment and Processes. *Procedia Manufacturing*, 11(June), 1634–1641. <https://doi.org/10.1016/j.promfg.2017.07.314>
- Yolanda, B. L. (2013). Metodología Ágil de Desarrollo de Software – XP, 10.
- Zaza, S. (2016). Test a Node RESTful API with Mocha and Chai. Retrieved from <https://scotch.io/tutorials/test-a-node-restful-api-with-mocha-and-chai>
- Zoss, A. (1996). Data Visualization: Visualization Types. Retrieved from https://guides.library.duke.edu/datavis/vis_types

11. GLOSARIO

- **COMPONENTE DE SOFTWARE:** Es un módulo, paquete o servicio que encapsula un conjunto de funciones relacionadas para realizar una tarea.
- **ENDPOINT:** Los servicios pueden interconectarse a través de la red. En una arquitectura orientada a servicios, cualquier interacción punto a punto implica dos endpoints: uno que proporciona un servicio, y otro de lo consume. Es decir, que un endpoint es cada uno de los "elementos", en nuestro caso nos referimos a servicios, que se sitúan en ambos "extremos" de la red que sirve de canal de comunicación entre ellos. Cuando hablamos de servicios Web, un endpoint se especifica mediante una URI. ("Introducción a los Servicios Web. Invocación de servicios web SOAP.," 2014)
- **JSON:** Es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - diciembre 1999. JSON es un formato de texto que es completamente independiente del lenguaje, pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. ("Introducción a JSON," n.d.)
- **MINDFULNESS:** es una manera de prestar atención de manera intencional a las experiencias que ocurren en un momento la cual es originaria de las prácticas de meditación orientales.
- **POSTMAN:** es una extensión del navegador Google Chrome, que permite el

envío de peticiones HTTP REST sin necesidad de desarrollar un cliente.("Pruebas REST con Postman Jetpacks," n.d.)

- **PRUEBAS UNITARIAS:** Es una forma de comprobar el correcto funcionamiento de una unidad de código.(Barrientos, 2014)
- **REPOSITORIO:** son versiones del proyecto que se encuentran alojados en Internet o en algún punto de la red.("Git - Trabajando con repositorios remotos," n.d.)
- **RESTFUL:** Es un modelo de arquitectura web basado en el protocolo HTTP, que se nutre con un conjunto de buenas prácticas para mejorar las comunicaciones cliente-servidor(Damian Wajser, 2015)
- **UML:** Lenguaje de Modelado Unificado (Unified Modeling Language).
- **VISUALIZACIÓN CIENTÍFICA:** Pretende a través de los datos generar un conjunto gráfico que muestre puntos clave y agrupados que permita entender relaciones o tendencias que llevan a conclusiones en su interpretación. Estos pueden ser modelados en 2D (barras, circular, etc.) o en 3D (superficies, objetos).
- **WEBSERVICE:** Es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como internet.(Culturación, n.d.)

12.ANEXOS

ANEXO 1

COMPONENTE WEB PARAMETRIZABLE PARA LA VISUALIZACIÓN DE DATOS CIENTÍFICOS

Especificación de Requerimientos de Software
Versión 1.0

Revisión histórica

Fecha	Versión	Descripción	Autor
08/27/2017	1.0	Se inició el desarrollo del documento	Cesar Yesid Pineda Edward Jiménez

Tabla de Contenido

1. Introducción	53
1.1 Propósito	53
1.2 Alcance	53
1.3 Definiciones, Acrónimos y Abreviaciones	53
1.4 Referencias	54
1.5 Visión general del documento	54
2. Descripción General	54
2.1 Funciones del Producto	54
2.2 Características del usuario	55
2.3 Restricciones	56
2.4 Suposiciones y Dependencias	56
2.5 Requisitos futuros	56
3. Requerimientos Específicos	56
3.1 Características del componente (Requerimientos Funcionales)	56
3.1.1 Autenticación	56
3.1.2 Set de datos	57
3.1.3 Visualización	59

Lista de Tablas

Tabla 1 Características de usuario	56
Tabla 2 Ingresar al sistema.....	57
Tabla 3 Registro de usuario	57
Tabla 4 Escoger set de datos	58
Tabla 5 Guardar set de datos	58
Tabla 6 Modificar origen set de datos	58
Tabla 7 Selección de columna	59
Tabla 8 Consultar set de datos	59
Tabla 9 Seleccionar graficas.....	60
Tabla 10 Visualización de graficas.....	60
Tabla 11 Parametrizar dashboard.....	61
Tabla 12 Crear múltiples dashboard	61
Tabla 13 Selección de grafica en dashboard	61

Lista de Imágenes

Imagen 1 Diagrama de casos de uso	55
---	----

Especificación de Requerimientos de Software

1. Introducción

Este documento SRS describe los requerimientos del componente web parametrizable para la visualización de datos científicos desarrollado para un proyecto de re experimentación emocional y mindfulness para la Universidad Católica de Colombia. Sin embargo, es de ayuda para el desarrollo en los procesos del ciclo de vida y de entrega de software.[1]

1.1 Propósito

El propósito de este documento es brindar una descripción detallada del componente web parametrizable para la visualización de datos científicos en sus características, sus parámetros, objetivos y analizando todos los requerimientos que han surgido para definirlo, de cómo será el componente y su funcionalidad. Este documento va dirigido a desarrolladores que busquen mejorar el componente o interactuar con este, mediante su software.

1.2 Alcance

El componente de visualización se basará en graficas de barras, líneas y circular (tortas). Razón por la cual, los requerimientos funcionales y no funcionales que se mencionaran a continuación en este documento son estrictamente relacionados a estas gráficas.

1.3 Definiciones, Acrónimos y Abreviaciones

- Componente de software: Es un módulo, paquete o servicio que encapsula un conjunto de funciones relacionadas para realizar una tarea.
- Mindfulness: es una manera de prestar atención de manera intencional a las experiencias que ocurren en un momento la cual es originaria de las prácticas de meditación orientales.
- Stakeholders: Persona u organización interesada en un proyecto que lo beneficia.[2]
- Visualización científica: Pretende a través de los datos generar un conjunto gráfico que muestre puntos clave y agrupados que permita entender relaciones o tendencias que llevan a conclusiones en su interpretación. Estos pueden ser modelados en 2D (barras, circular, etc.) o en 3D (superficies, objetos).

1.4 Referencias

- [1] IEEE-SA, “830-1998 - IEEE Recommended Practice for Software Requirements Specifications”, *IEEE STANDARD*, 2009. [En línea]. Disponible en: <https://standards.ieee.org/findstds/standard/830-1998.html>.
- [2] H. A. Rivera Rodríguez y M. N. Malaver Rojas, “La organización: los stakeholders y la responsabilidad social”, 2011.
- [3] Volere, “Volere Requirements Specification Template”, 2016. [En línea]. Disponible en: <http://www.volere.co.uk/template.htm>.

1.5 Visión general del documento

El presente documento de especificación de requerimientos se organiza en 3 secciones que se describen a continuación:

1. Introducción: Sección que busca dar el contexto a la especificación.
2. Descripción general: Sección dedicada a mostrar las consideraciones tenidas en cuenta del componente de visualización para los requerimientos, así como para su realización. Se describen los siguientes elementos:
 - Funciones del producto
 - Características de los usuarios
 - Restricciones
3. Requerimientos específicos: Sección que detalla cada uno de los requerimientos del componente de acuerdo con su clasificación funcional o no funcional, por lo cual permite llevar un correcto desarrollo y de las pruebas.

2. Descripción General

Los requerimientos que se presentan en el presente documento son el resultado de una serie de reuniones e investigaciones acerca de cómo se parametrizará el componente para recibir los datos, el tipo de acciones que podrá realizar el usuario, así como las restricciones que ya se han determinado en el documento de trabajo de grado.

2.1 Funciones del Producto

Con el fin de dar un mejor entendimiento de la herramienta de visualización a los stakeholders, se elaboró un diagrama de caso de uso con el cual se muestran que actores intervienen y como es la interacción de los diferentes casos.

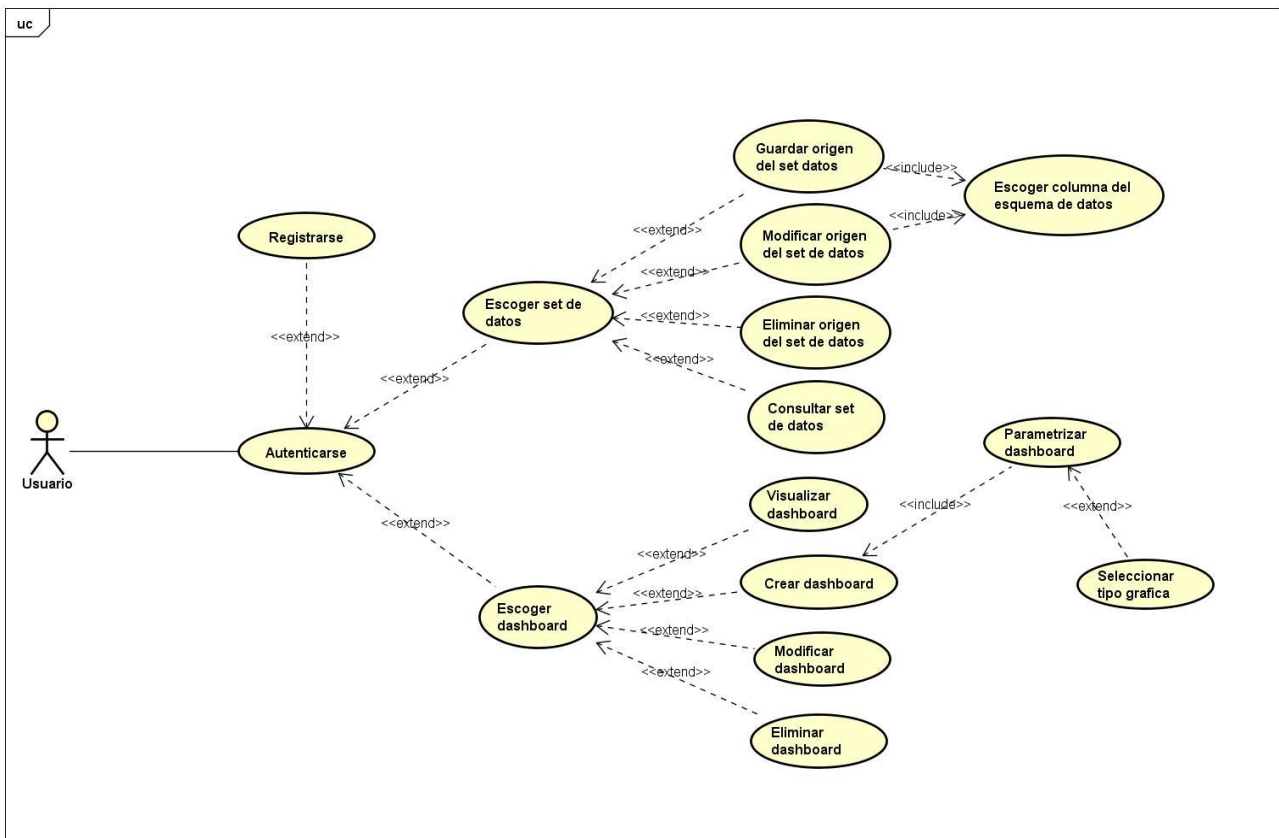


Imagen 1 Diagrama de casos de uso

2.2 Características del usuario

A continuación, se describe las características del usuario que usarán la aplicación teniendo en cuenta los privilegios, el rol, la experiencia técnica y la frecuencia de uso.

Características del usuario	El usuario debe ser una persona interesada en realizar un análisis visual de datos.
Privilegios	El usuario podrá tener acceso a las opciones: <ul style="list-style-type: none"> • Consultar por independiente cada gráfica. • Parametrizar los dashboard. • Consultar los sets de datos.
Rol	El único rol es el de usuario

Experiencia técnica	Debe conocer los diferentes componentes que componen el proyecto
Frecuencia de uso	La frecuencia de uso lo determinará el usuario.

Tabla 1 Características de usuario

2.3 Restricciones

- La interfaz gráfica está desarrollada totalmente en español.
- El lenguaje utilizado será Node js y html.
- El navegador recomendado será Mozilla Firefox.

2.4 Suposiciones y Dependencias

Suposiciones

A continuación, se definen las suposiciones que afectan el componente de visualización y sus requerimientos en caso de que no se cumplan:

- El componente interactuará con otros componentes del proyecto los cuales consumirán sus servicios.
- El componente estará alojado en un servidor estando disponible el 80% de tiempo anual.

Dependencias

- Orquestador: Componente encargado de coordinar los procesos de acuerdo con las peticiones que reciba y verificar si los demás componentes del sistema están activos.

2.5 Requisitos futuros

- Estructurar el componente para el manejo de perfiles de usuario en el proyecto macro.
- Mostrar graficas con actualizaciones en tiempo real.

3. Requerimientos Específicos

En esta sección se explica la especificación de requerimientos de manera detallada, usando como guía la plantilla Volere [3] y se clasifican de forma funcional para facilitar la trazabilidad en su desarrollo.

3.1 Características del componente (Requerimientos Funcionales)

3.1.1 Autenticación

# Requerimiento	01	Tipo de Requerimiento	Funcional	Caso de Uso	
Descripción	El componente debe permitir al usuario ingresar al sistema.				

Razón	El usuario debe autenticarse para poder usar el sistema.		
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández		
Criterio de medición	Mostrar un mensaje de alerta en caso de que el usuario y contraseña sea incorrecto o exista algún error del componente.		
Prioridad	Media		
Versión	1.0	Fecha	2017-09-15

Tabla 2 Ingresar al sistema

# Requerimiento	02	Tipo de Requerimiento	Funcional	Caso de Uso	
Descripción	El componente debe permitir al usuario registrarse				
Razón	El usuario debe autenticarse para poder usar el sistema.				
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández				
Criterio de medición	Si los datos registrados son correctos, debe inmediatamente ser almacenados en la base de datos.				
Prioridad	Media				
Versión	1.0	Fecha	2017-09-15		

Tabla 3 Registro de usuario

3.1.2 Set de datos

# Requerimiento	03	Tipo de Requerimiento	Funcional	Caso de Uso	
Descripción	El componente debe permitir al usuario escoger los sets de datos provenientes de una base de datos relacional				
Razón	Es la fuente de la información para ser visualizada.				
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández				
Criterio de medición	Los sets de datos deben listarse en segundos para poder seleccionarlos e indicar si existe algún error al cargarlos.				

Prioridad	Alta		
Versión	1.0	Fecha	2017-09-15

Tabla 4 Escoger set de datos

# Requerimiento	04	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario guardar el origen del set de datos			
Razón	Poder usarlos de nuevo sin tener que buscarlos.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición	El origen del set de datos debe estar disponible y de acuerdo con esto poderlo cargar en segundos.			
Prioridad	Media			
Versión	1.0	Fecha	2017-09-15	

Tabla 5 Guardar set de datos

# Requerimiento	05	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario modificar el origen del set de datos guardados			
Razón	No requerir de alguno del set de datos que tiene guardados, puede eliminarlo y tomar otro.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición	El set de datos debe estar disponible, tanto en su origen como en donde se encuentra guardado en el componente.			
Prioridad	Media			
Versión	1.0	Fecha	2017-09-15	

Tabla 6 Modificar origen set de datos

# Requerimiento	06	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario seleccionar una columna del esquema de datos relacional			
Razón	Debe parametrizar las gráficas con los set de datos seleccionados.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición				
Prioridad	Alta			
Versión	1.0	Fecha	2017-09-15	

Tabla 7 Selección de columna

# Requerimiento	07	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario consultar los set de datos			
Razón	Debe verificar los set que tiene disponibles para poder seleccionarlos y usarlos en la visualización			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición				
Prioridad	Media			
Versión	1.0	Fecha	2017-09-15	

Tabla 8 Consultar set de datos

3.1.3 Visualización

# Requerimiento	08	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario escoger las graficas			
Razón	Debe poder general a partir de las gráficas sus dashboard			

Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández		
Criterio de medición			
Prioridad	Alta		
Versión	1.0	Fecha	2017-09-15

Tabla 9 Seleccionar graficas

# Requerimiento	09	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario visualizar de manera independiente cada grafica del dashboard			
Razón	Para generar un análisis visual de los sets de datos seleccionados.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición				
Prioridad	Alta			
Versión	1.0	Fecha	2017-09-15	

Tabla 10 Visualización de graficas

# Requerimiento	10	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario parametrizar los dashboard			
Razón	Para generar un análisis visual de los sets de datos seleccionados.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición				
Prioridad	Alta			

Versión	1.0	Fecha	2017-09-15
---------	-----	-------	------------

Tabla 11 Parametrizar dashboard

# Requerimiento	11	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario crear múltiples dashboard con fuente de datos independientes.			
Razón	Para generar un análisis visual de los sets de datos seleccionados.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición				
Prioridad	Alta			
Versión	1.0	Fecha	2017-09-15	

Tabla 12 Crear múltiples dashboard

# Requerimiento	12	Tipo de Requerimiento	Funcional	Caso de Uso
Descripción	El componente debe permitir al usuario seleccionar el tipo de grafica dentro de cada dashboard.			
Razón	Para generar un análisis visual del set de datos seleccionados.			
Autor	Cesar Yesid Pineda Patarroyo, Edward Jiménez Hernández			
Criterio de medición				
Prioridad	Alta			
Versión	1.0	Fecha	2017-09-15	

Tabla 13 Selección de grafica en dashboard

ANEXO 2



UNIVERSIDAD CATÓLICA
de Colombia

COMPONENTE WEB PARAMETRIZABLE PARA LA VISUALIZACIÓN DE DATOS CIENTÍFICOS

Cesar Yesid Pineda Patarroyo

Edward Jiménez Hernández

DOCUMENTO DE DISEÑO DE SOFTWARE

DIEGO ALBERTO RINCÓN YÁÑEZ MSc

Contenido

Control de Cambios	65
Introducción	66
Alcance	66
Supuestos.....	66
Dependencias.....	66
Restricciones	66
Riesgos.....	67
Metodología de Pruebas.....	67
Arquitectura del Sistema	67
Nivel General.....	68
Diagrama de Componentes.....	69
Diagrama de Despliegue	69
Diagrama de Infraestructura	70
Modelo Entidad Relación.....	70
Módulo Administrador de Usuarios.....	71
Restricciones.....	71
Interacciones de Usuario.....	71
Módulo Administrador de Dashboard	75
Restricciones.....	75
Interacciones de Usuario.....	75
Políticas	79
Glosario	79
Bibliografía.....	80

Tabla de imágenes

Imagen 1 Diagrama de componentes	69
Imagen 2 Diagrama de despliegue	69
Imagen 3 Diagrama de infraestructura.....	70
Imagen 4 Diagrama modelo entidad relación	71
Imagen 5 Diagrama de actividades - Crear usuario.....	72
Imagen 6 Diagrama de actividades - Modificar usuario	73
Imagen 7 Diagrama de actividades - Eliminar usuario.....	74
Imagen 8 Diagrama de actividades - Consultar usuario	75
Imagen 9 Diagrama de actividades - Crear dashboard.....	76
Imagen 10 Diagrama de actividades - Modificar dashboard	77
Imagen 11 Diagrama de actividades - Consultar dashboard	78
Imagen 12 Diagrama de actividades - Eliminar dashboard.....	79

Control de Cambios

Fecha	Autor	Versión	Comentarios
10/16/2017	Cesar Yesid Pineda Edward Jiménez	1.0	Se inició el desarrollo del documento

Introducción

Este documento SDD describe el diseño de software del componente web parametrizable para la visualización de datos científicos desarrollado inicialmente para un proyecto de re experimentación emocional y mindfulness para la Universidad Católica de Colombia, pero este componente podrá adaptarse a cualquier proyecto mientras se cumpla con los parámetros que solicita. Sin embargo, es de ayuda para el desarrollo en los procesos del ciclo de vida y de entrega de software en el cual se nombra las suposiciones y restricciones que se deben tener en cuenta al diseñarlo.[1]

Alcance

- El periodo de tiempo de implementación es menor a 1 año (6 meses aproximadamente).
- La elaboración de esta herramienta permitirá una visualización limpia de los datos científicos para un posterior análisis.
- Se diagramará los elementos más importantes del componente de visualización de datos científicos.

Supuestos

- Equipo de desarrollo: El grupo de desarrollo cuenta con los elementos necesarios para diseñar y desarrollar el componente de visualización.
- Equipos: Debe contar con acceso a internet.
- Servidor: Debe tener el servicio de node js activo durante la ejecución del componente.

Dependencias

- El equipo donde este almacenado el componente de visualización debe contar con conexión a internet, para comunicarse con los demás componentes.
- Deben existir datos previamente extraídos y procesados para que el componente los pueda usar.

Restricciones

- La actualización o retroalimentación de la herramienta se ve comprometida por tiempo definido.
- La visión planteada por el proyecto general implica que este se desarrolle por fases, en las cuales el proyecto propuesto tomaría el papel de visualización de datos y estará sujeto a fases previas que se han desarrollado o están en desarrollo.
- El tiempo de desarrollo está limitado por el periodo académico.
- La interfaz gráfica está desarrollada totalmente en español.
- El lenguaje utilizado será Node js y html.
- El navegador recomendado será Mozilla Firefox.
- Se implementarán 3 graficas que son: pie, barras y líneas.

Riesgos

- Cambio en algún requerimiento del cual dependen algunos diagramas.
- Cambio en algún diagrama que afecte los demás.
- No se es capaz de comprender toda la lógica del lenguaje de programación.

Metodología de Pruebas

Las pruebas se realizarán de una forma donde se creará el usuario y se crearán los dashboard con este, allí se verificará que esté funcionando también los webservice. De esta forma se verificará que los planteado esté funcionando correctamente.

Arquitectura del Sistema

Debido a las características del componente de visualización se verificaron varios lenguajes que permitieran cumplir con el objetivo de proyecto y pueda seguir creciendo a futuro se usara NodeJS el cual se acopla más con lo que se busca con este componente. Es por ello que se describirá para su comprensión a continuación.

NodeJS, debido a que este lenguaje, bajo el enfoque de ser útil en la creación de programas de red es altamente escalable y es muy conocido por ser una tecnología que permite trabajar con Javascript del lado del servidor.

Node.js está basado en el motor V8 de Javascript de Google, el cual está diseñado para correr en un navegador y ejecutar el código de Javascript de una forma extremadamente rápida. La tecnología detrás de Node.js permite ejecutar este motor en el lado del servidor, abriendo un nuevo abanico de posibilidades en cuanto al mundo de desarrollo se refiere. Para ello el entorno de Node.js ha desarrollado un conjunto amplio de librerías comparables a las de otras plataformas.[2]

NodeJS, posee una gran variedad de ventajas, pero para destacar, a continuación, se destaca, en forma no exhaustiva, 4 características resaltables:

- Con Node se puede utilizar javascript como lenguaje de scripting en consola (como bash, perl, etc.).[2]
- Permite utilizar el mismo lenguaje (javascript), tanto en el cliente como en el servidor.[2]
- Ofrece muy buena gestión de paquetes gracias a NPM. Detrás de NodeJS hay una gran comunidad documentando, haciendo tutoriales y creando nuevos módulos.[2]

Está basado en eventos, así que toda la filosofía asíncrona que ya utiliza AJAX en el cliente (navegador), se puedes pasar al servidor.[2]

MySQL, es un software de código abierto, licenciado bajo la GPL de la GNU. El lenguaje de programación que utiliza MySQL es Structured Query Language (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales.[3]

En las últimas versiones se pueden destacar las siguientes características principales:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.

VENTAJAS

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Soporta gran variedad de Sistemas Operativos
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.

Conectividad y seguridad

Nivel General

El componente de visualización se basará de acuerdo a los tipos de diagramas UML ya que estos nos ayudan a diseñar, estructurar, visualizar, especificar y documentar los elementos del software de los cuales se definió el diseño del componente. [4]

- Arquitectura
 - Diagrama de componentes
 - Diagrama de despliegue
 - Diagramas de comportamiento
 - Diagrama de clases

Diagrama de Componentes

El presente diagrama muestra cómo están divididos los componentes del módulo de visualización del proyecto, donde se llevó a definirlo por el tipo de arquitectura y poder mostrarlo con este diagrama. Representa un alto nivel de abstracción al que presenta un diagrama de clases.

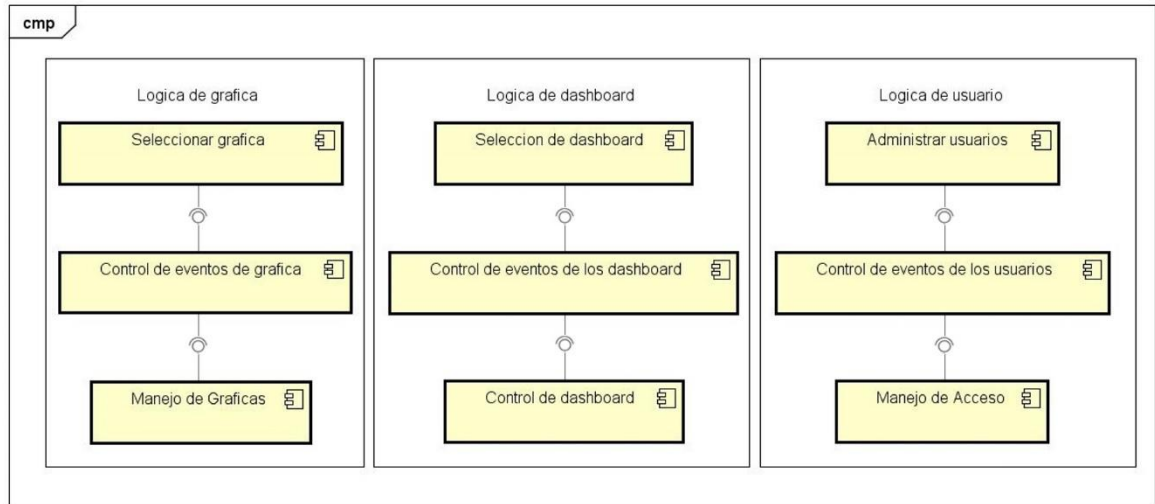


Imagen 2 Diagrama de componentes

Diagrama de Despliegue

El diagrama de despliegue muestra por medio de los nodos, como es la ejecución de los componentes y como se distribuirán de manera correcta, pero en un principio tanto el aplicativo como la base de datos estarán en un mismo servidor.

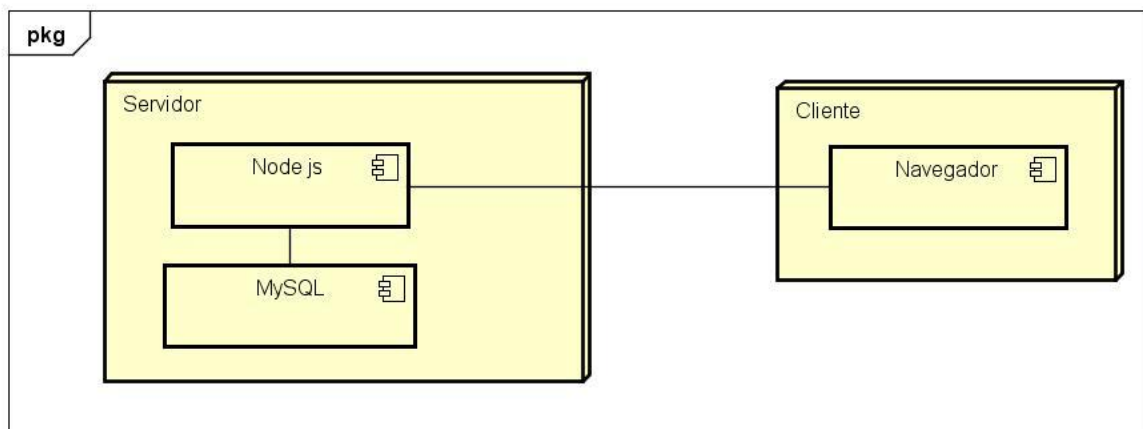


Imagen 3 Diagrama de despliegue

Diagrama de Infraestructura

En un diagrama a un alto nivel que nombramos de infraestructura, se muestra como estará conformado el componente y los utilitarios necesarios para su funcionamiento.

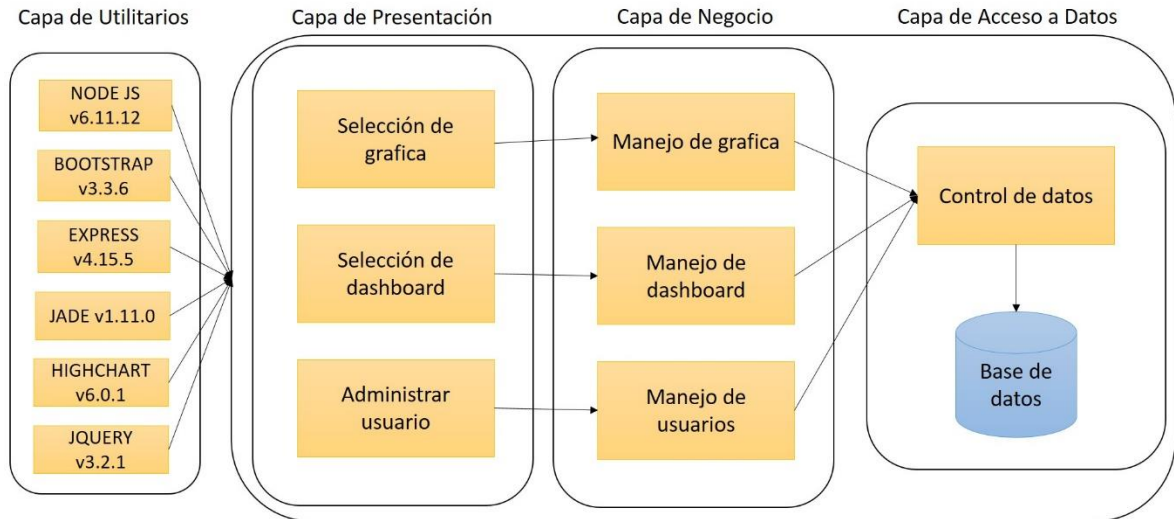


Imagen 4 Diagrama de infraestructura

Modelo Entidad Relación

En el modelo entidad relación se muestra como estará conformada la base de datos para el almacenamiento de los diferentes datos que necesita el componente y los que generará el usuario con su interacción.

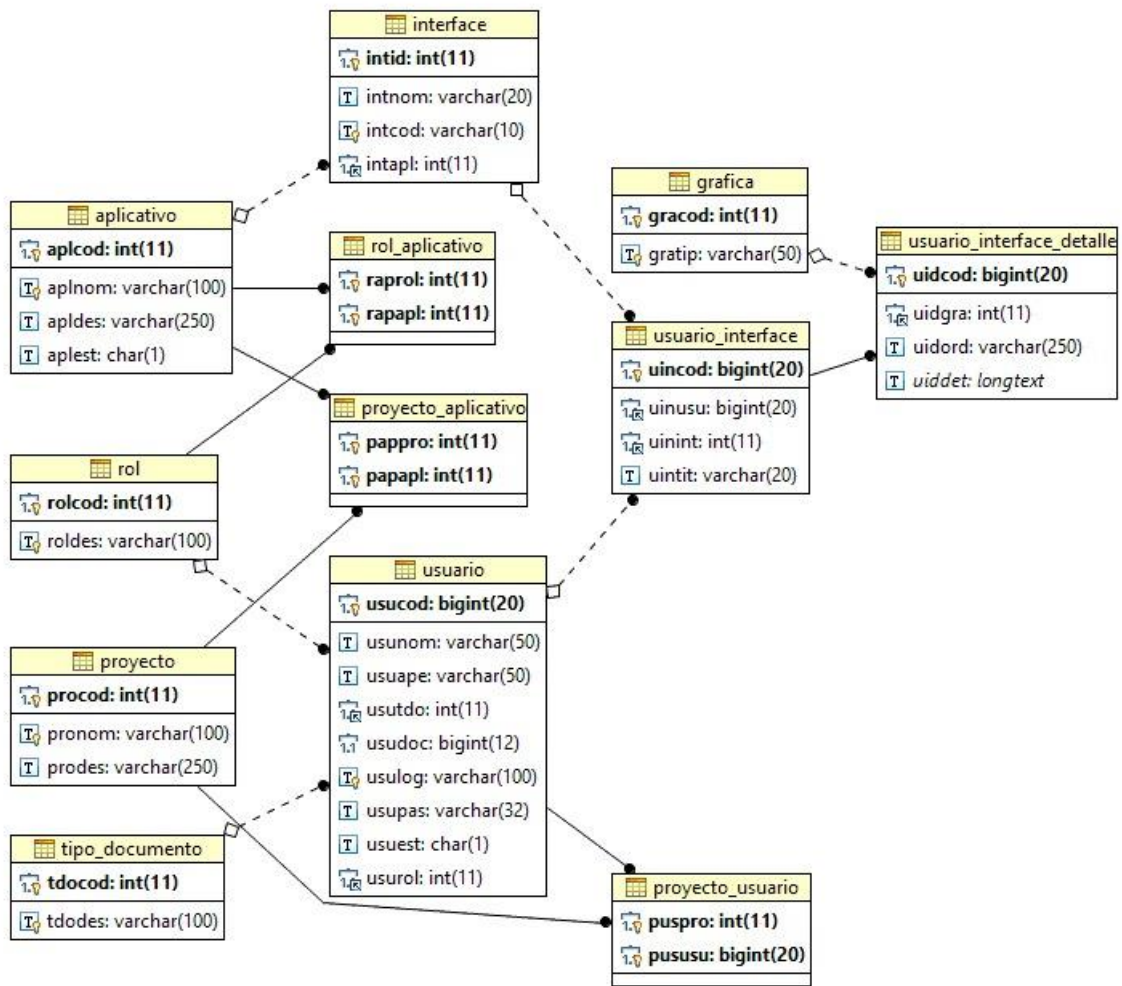


Imagen 5 Diagrama modelo entidad relación

Módulo Administrador de Usuarios

El módulo *Administración de Usuarios* permitirá la creación, actualización y eliminación de usuarios del sistema.

Restricciones

La restricción existirá para todos los usuarios a excepción del usuario Investigador Principal el cual tendrá la opción de crear los usuarios.

Interacciones de Usuario

Se muestra a continuación mediante diagramas UML como interactúa el usuario con el medio del componente de visualización.

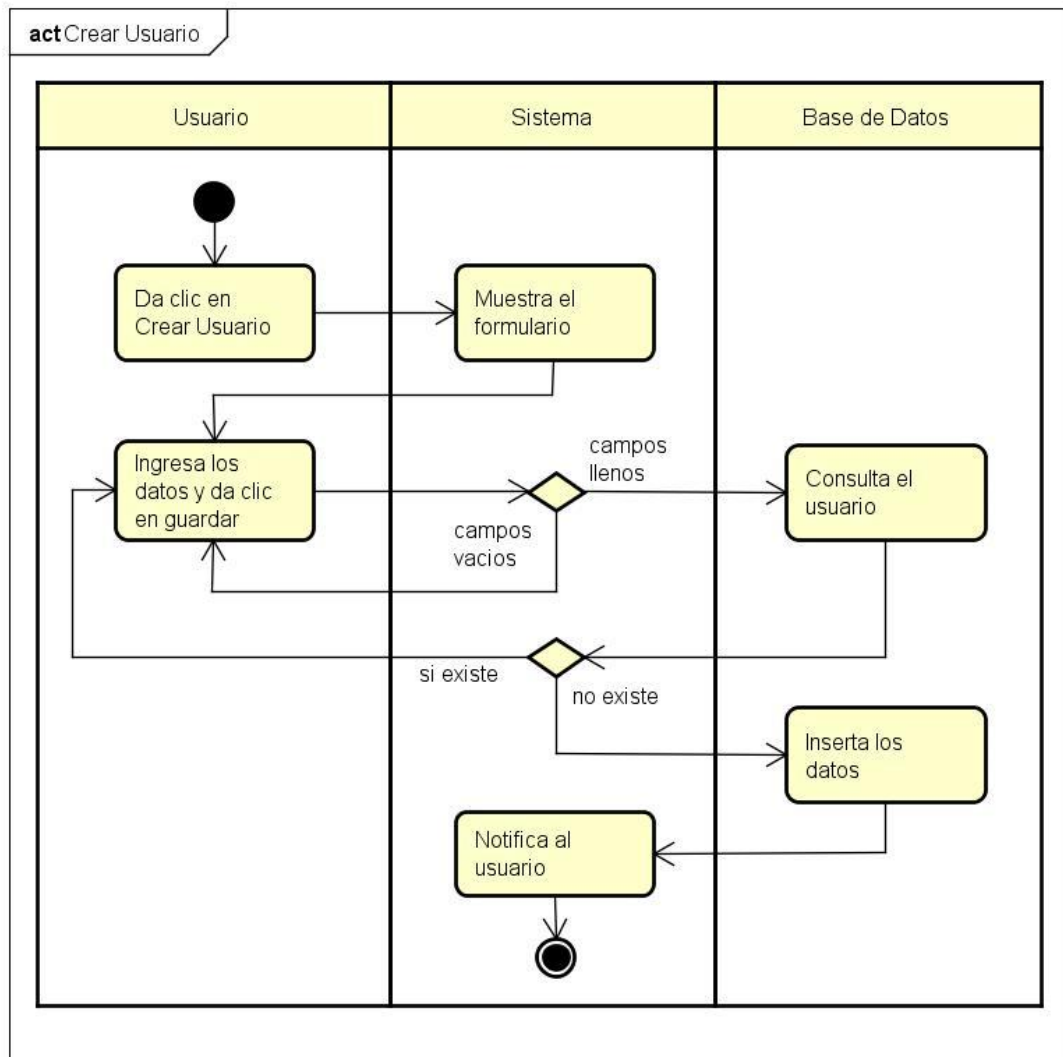


Imagen 6 Diagrama de actividades - Crear usuario

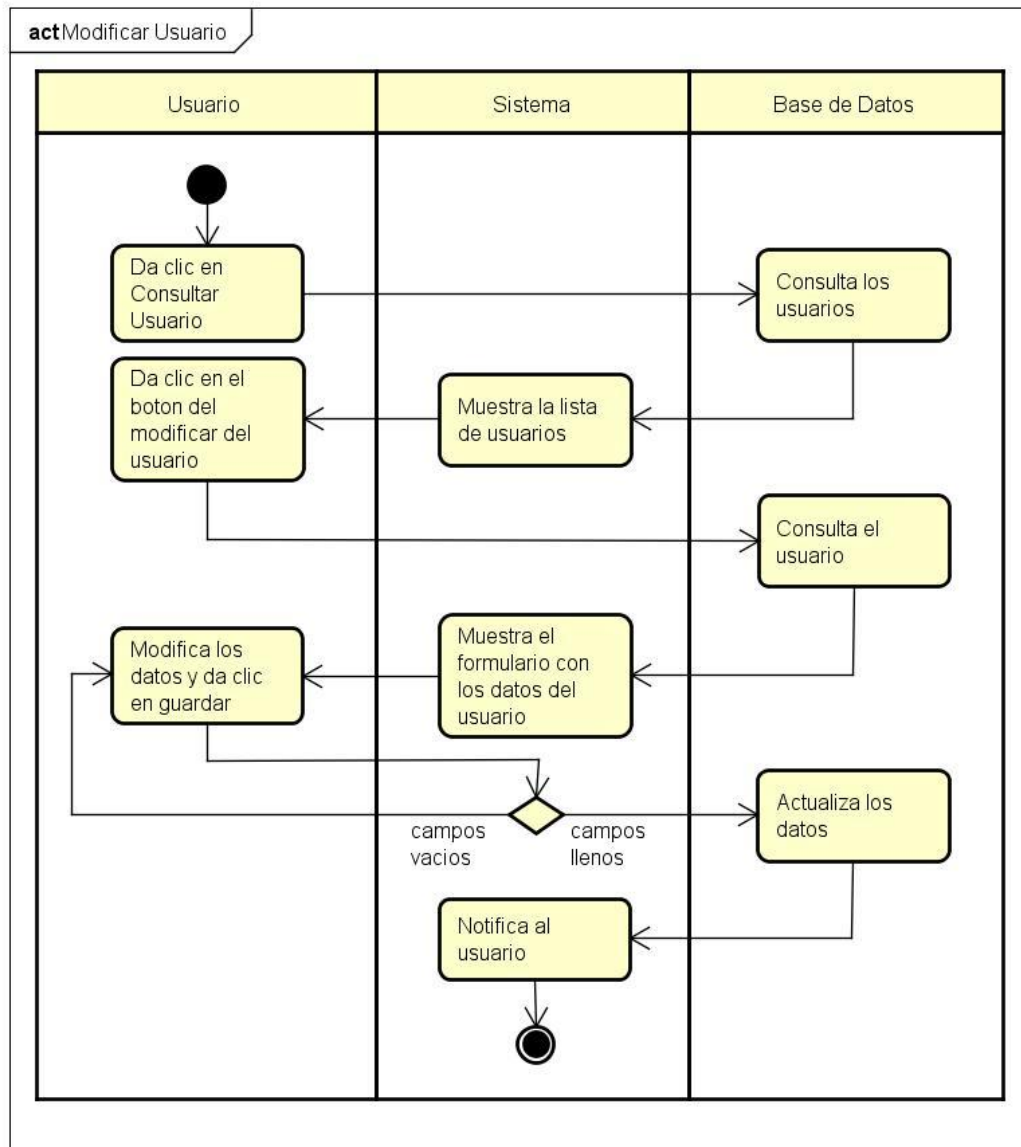


Imagen 7 Diagrama de actividades - Modificar usuario

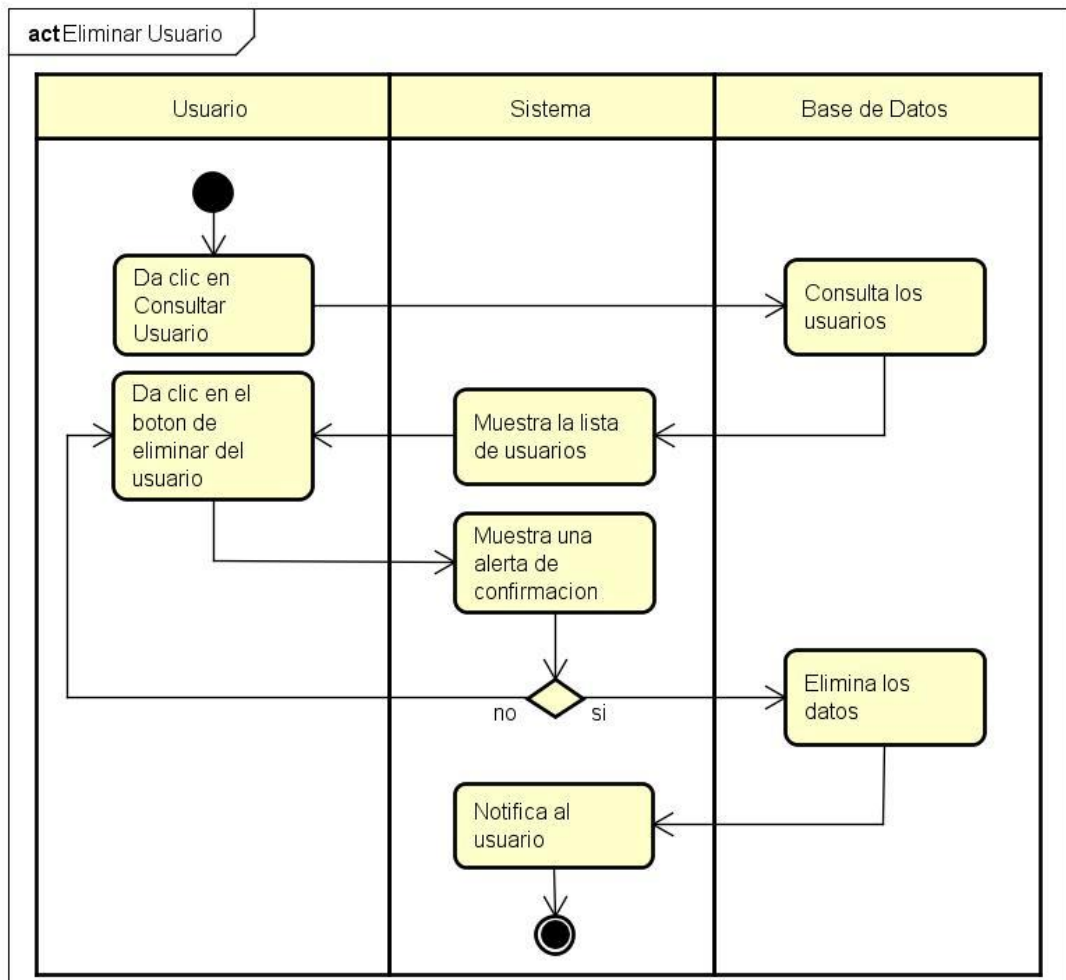


Imagen 8 Diagrama de actividades - Eliminar usuario

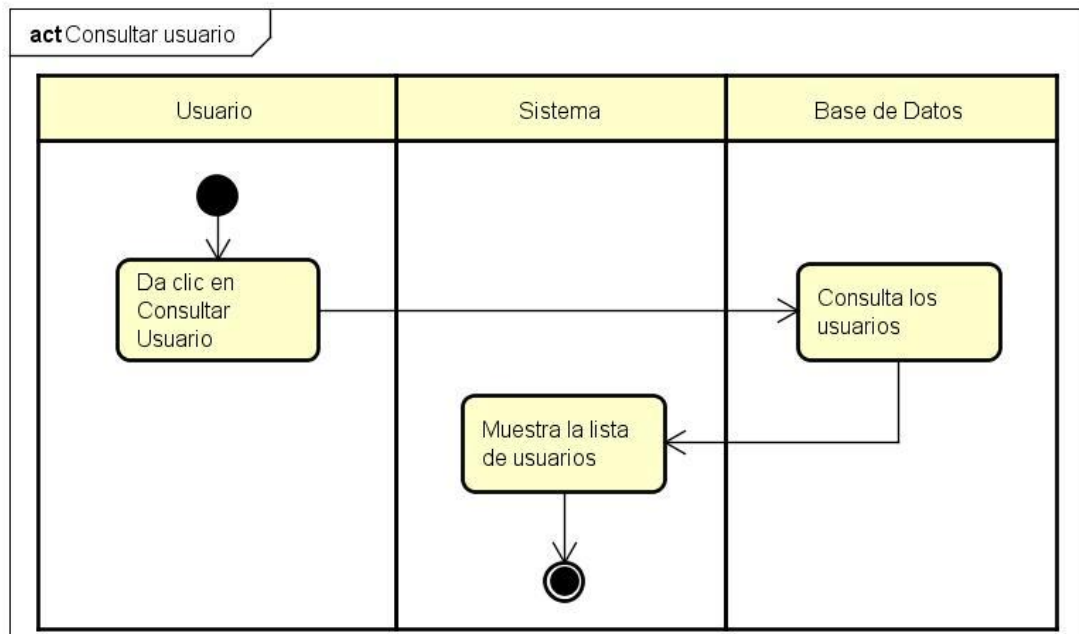


Imagen 9 Diagrama de actividades - Consultar usuario

Módulo Administrador de Dashboard

El módulo *Administración de Dashboard* permitirá la creación, actualización y eliminación de dashboard.

Restricciones

Solo se disponen de 10 plantillas para crear los dashboard.

Interacciones de Usuario

Se muestra a continuación mediante diagramas UML como interactúa el usuario con el modio del componente de visualización.

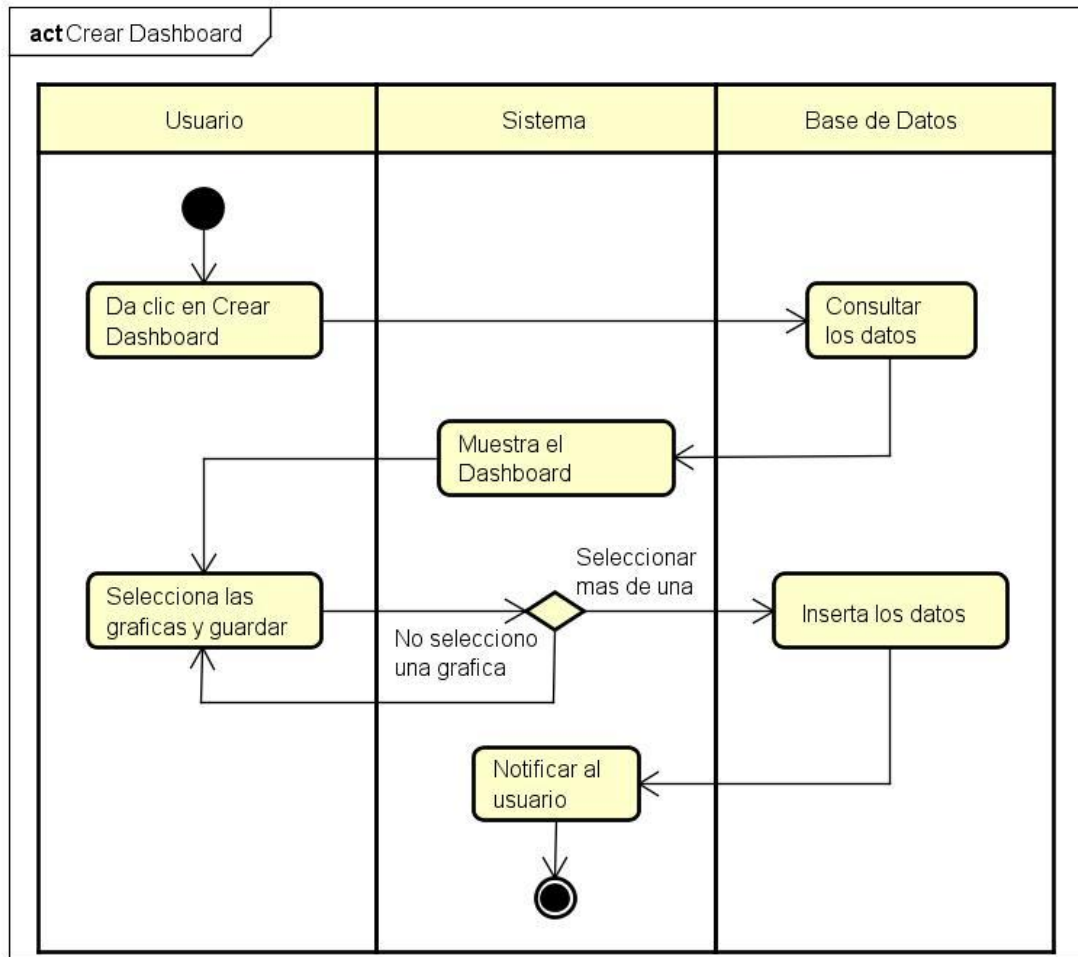


Imagen 10 Diagrama de actividades - Crear dashboard

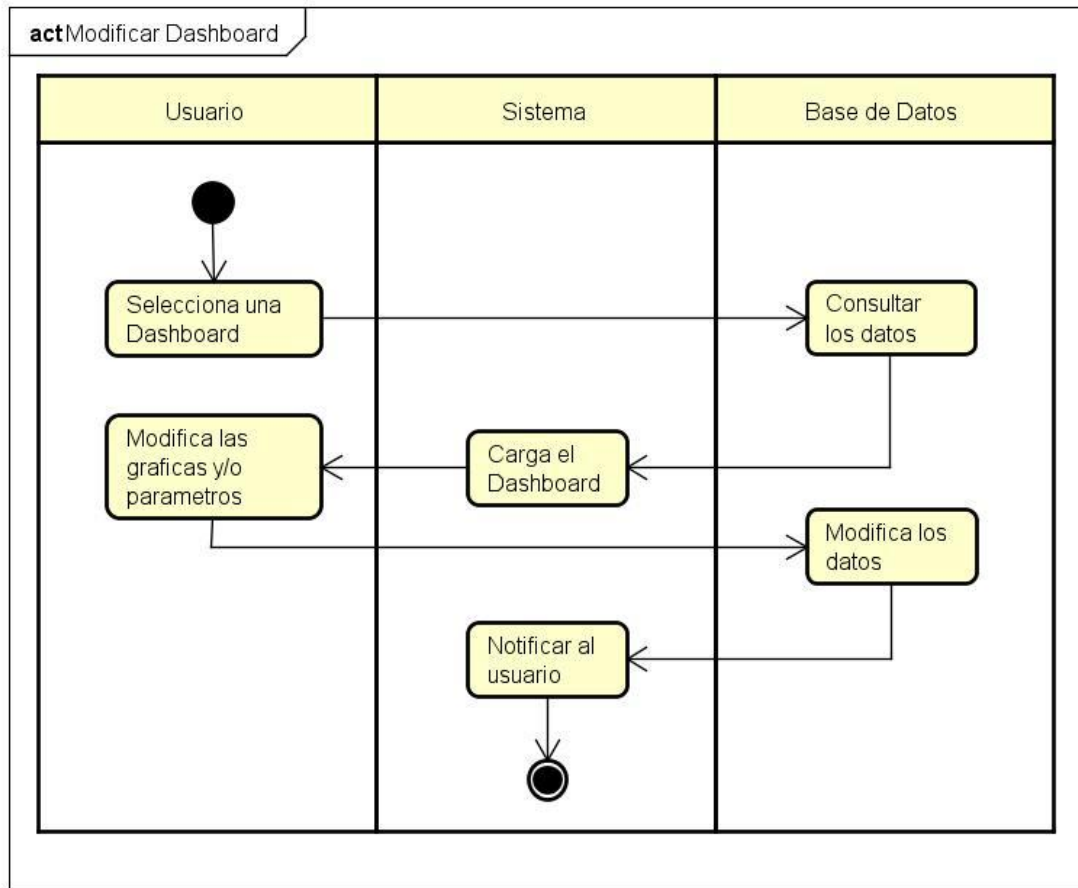


Imagen 11 Diagrama de actividades - Modificar dashboard

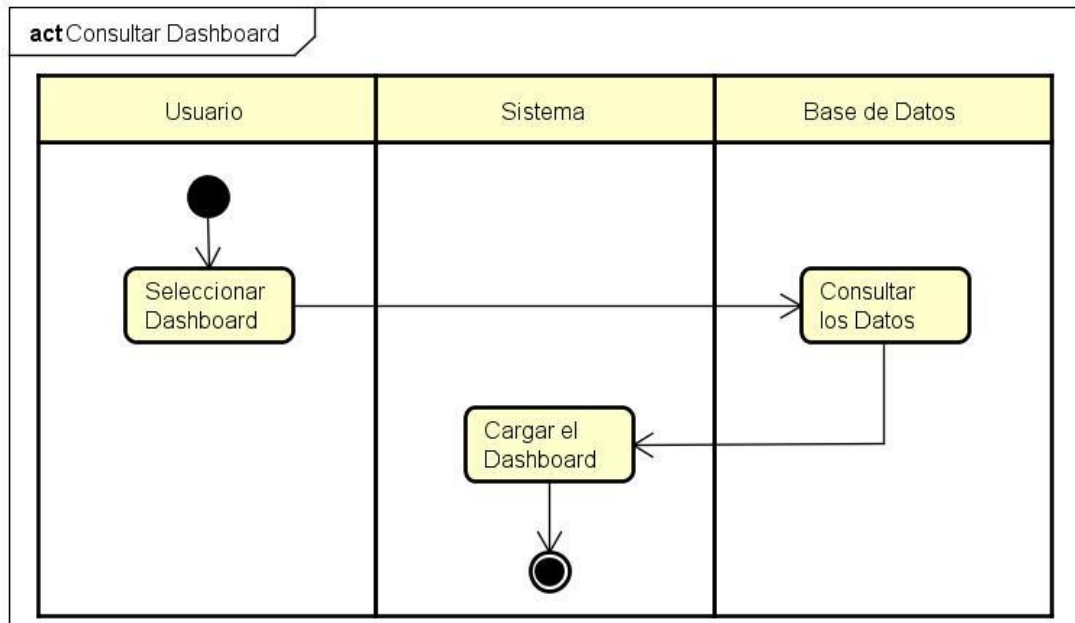


Imagen 12 Diagrama de actividades - Consultar dashboard

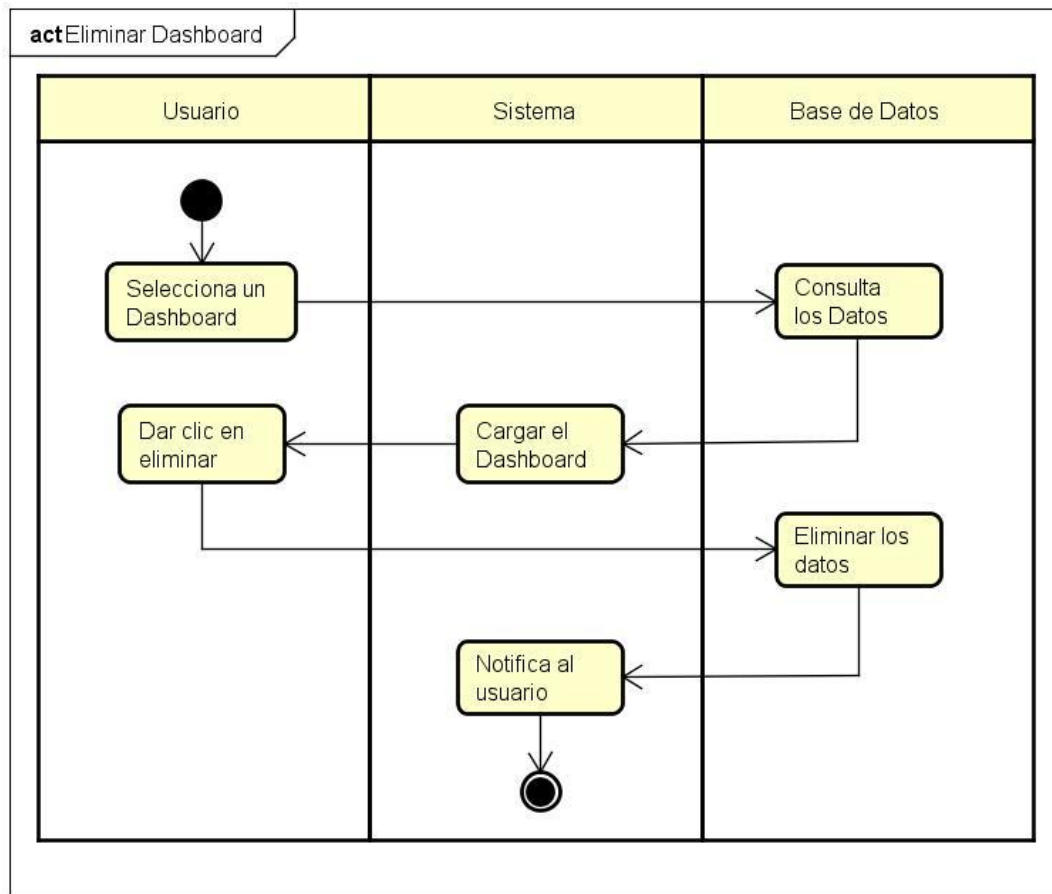


Imagen 13 Diagrama de actividades - Eliminar dashboard

Políticas

- La instalación del componente se realizará exclusivamente en los servidores propiedad de la Universidad Católica de Colombia.
- Es facultad exclusiva de los miembros del proyecto de realizar la instalación del componente en los servidores propiedad de la Universidad Católica de Colombia.
- Es atribución exclusiva de los miembros del proyecto dimensionar las necesidades de componente de visualización.

Glosario

- Componente de software: Es un módulo, paquete o servicio que encapsula un conjunto de funciones relacionadas para realizar una tarea.
- Dependencia: Las dependencias de software se utilizan para identificar relaciones entre elementos de software.[5]
- Mindfulness: es una manera de prestar atención de manera intencional a las experiencias que ocurren en un momento la cual es originaria de las prácticas

de meditación orientales.

- SDD: Documento de Diseño de Software (Software Design Document).
- UML: Lenguaje de Modelado Unificado (Unified Modeling Language).
- Visualización científica: Pretende a través de los datos generar un conjunto gráfico que muestre puntos clave y agrupados que permita entender relaciones o tendencias que llevan a conclusiones en su interpretación. Estos pueden ser modelados en 2D (barras, circular, etc.) o en 3D (superficies, objetos).

Bibliografía

- [1] IEEE Computer Society, “12207-2008 - Systems and software engineering”, *IEEE STANDARD*, 2008. [En línea]. Disponible en: <https://standards.ieee.org/findstds/standard/12207-2008.html>.
- [2] ALIDA VERGARA, “5 ventajas de usar Node.js”, *ELASTIC CLOUD*. [En línea]. Disponible en: <https://www.facilcloud.com/noticias/5-ventajas-de-usar-node-js/>.
- [3] Enríquez Toledo Alma; Maldonado Ayala Jesús; Nakamura Ortega Yunko; Nogueron Toledo Goretty, “MySQL”, pp. 1–2, 1981.
- [4] I. Object Management Group, “What is UML”, 2017. [En línea]. Disponible en: <http://www.uml.org/what-is-uml.htm>.
- [5] IBM, “Dependencias de software”, *IBM Knowledge Center*, 2009. [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SS2GNX_7.1.1/com.ibm.ti voli.tpm.scenario.doc/software/csrm_reqcap.html.

ANEXO 3



UNIVERSIDAD CATÓLICA
de Colombia

COMPONENTE WEB PARAMETRIZABLE PARA LA VISUALIZACIÓN DE DATOS CIENTÍFICOS

Cesar Yesid Pineda Patarroyo

Edward Jiménez Hernández

DOCUMENTO DE ARQUITECTURA DE SOFTWARE

DIEGO ALBERTO RINCÓN YÁÑEZ MSc

Contenido

Control de Cambios	84
Introducción	85
Propósito y Alcance	85
Definiciones, Acrónimos y Abreviaciones	85
Organización del Documento	85
Representación de Stakeholders	85
Definiciones de Puntos de Vista	86
Punto de Vista Gerencial	86
Punto de Vista Lógica de Negocio	86
Punto de Vista Técnico	86
Antecedentes de la Arquitectura	87
Antecedentes del Problema	87
Visión General del Sistema	87
Objetivos y Contexto	87
Vistas	88
Vistas Estructurales	88
Vista de Componentes	88
Vista de Despliegue	90
Referencias	90

Tabla de imágenes

Imagen 1 Diagrama del componente a alto nivel	86
Imagen 2 Diagrama de componentes	88
Imagen 3 Diagrama capa de utilitarios.....	88
Imagen 4 Diagrama capa de acceso a datos.....	89
Imagen 5 Diagrama capa de negocio	89
Imagen 6 Diagrama capa de presentacion	90
Imagen 7 Diagrama de despliegue	90

Control de Cambios

Fecha	Autor	Versión	Comentarios
10/14/2017	Cesar Yesid Pineda Edward Jiménez	1.0	Se inició el desarrollo del documento.

Introducción

Propósito y Alcance

El propósito de este documento es brindar una descripción comprensiva del componente web parametrizable para la visualización de datos científicos en su arquitectura, de cómo será sus relaciones y características por medio de vista seleccionadas.[1]

Además, mostrar los elementos por medio de vistas de diseño, tanto internos que forman el componente de visualización, como algunos aspectos importantes de los externos que interactúan con el componente.

Definiciones, Acrónimos y Abreviaciones

- Componente de software: Es un módulo, paquete o servicio que encapsula un conjunto de funciones relacionadas para realizar una tarea.
- Mindfulness: es una manera de prestar atención de manera intencional a las experiencias que ocurren en un momento la cual es originaria de las prácticas de meditación orientales.
- MVC: Modelo Vista Controlador (Model View Controller).
- SAD: Documento de Arquitectura de Software (Software Architecture Document).
- Visualización científica: Pretende a través de los datos generar un conjunto gráfico que muestre puntos clave y agrupados que permita entender relaciones o tendencias que llevan a conclusiones en su interpretación. Estos pueden ser modelados en 2D (barras, circular, etc.) o en 3D (superficies, objetos).

Organización del Documento

El presente documento de arquitectura se organiza en 3 secciones que se describen a continuación:

- Introducción: Muestra el propósito y alcance del documento, además de algunas definiciones.
- Antecedentes de la Arquitectura: Una visión a alto nivel de lo que busca resolver la arquitectura.
- Vistas: Muestra de distintas maneras el componente para que pueda ser entendido.

Representación de Stakeholders

Administrador de sistema: Es el usuario que creara los usuarios y los proyectos, además de tener las opciones generales de crear dashboard.

Investigador principal: Es el usuario que podrá crear dashboard y visualizarlos. Con el fin de realizar su análisis visual.

Investigador auxiliar: Solo podrá visualizar los dashboard para realizar el análisis visual.

Definiciones de Puntos de Vista

Punto de Vista Gerencial

Descripción

La visualización de datos es más fácil para el usuario poder entender y revisar los datos, ya que el componente es muy importante para el desarrollo del proyecto, ya que al final todos los proyectos de data science terminan en la visualización.

Punto de Vista Lógica de Negocio

Descripción

La visualización de los datos permite realizar análisis y encontrar algunos factores determinantes que pueden ayudar a la toma de decisiones empresariales o científicas.

Punto de Vista Técnico

Descripción

El componente se basará en el patrón de arquitectura MVC donde la vista controlará las acciones del usuario, el controlador manejará los eventos que se generen de la vista y el modelo manejar el acceso a los datos por parte del controlador.[2][3]

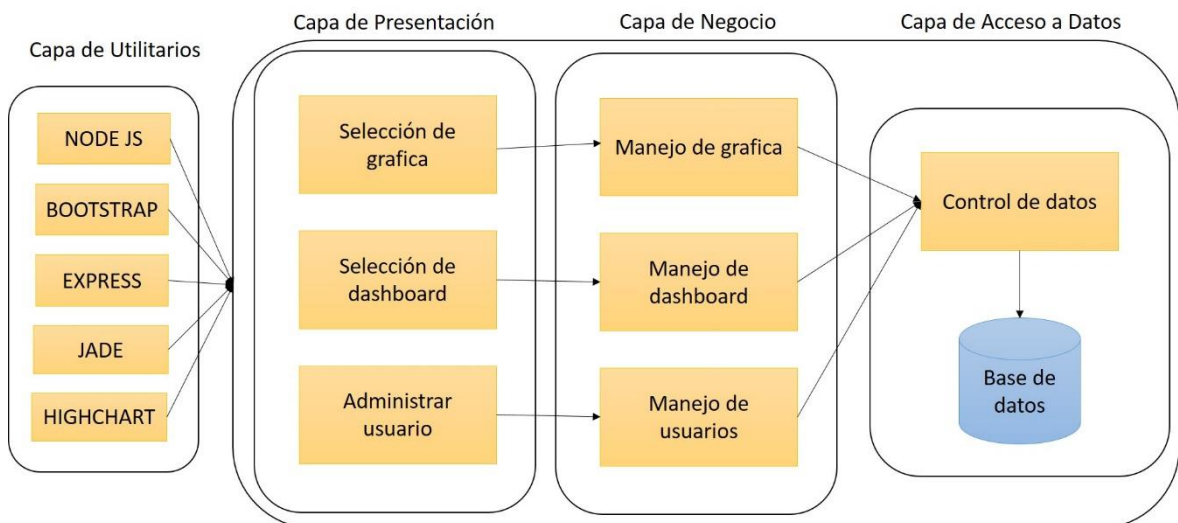


Imagen 14 Diagrama del componente a un alto nivel

Antecedentes de la Arquitectura

Antecedentes del Problema

En la actualidad se almacenan grandes volúmenes de datos los cuales siguen en aumento a través del tiempo y del cual es necesario usarlos para sacar un mayor provecho de ellos y que ciertas herramientas extraen estos datos y los almacenan en de una manera adecuada. Pero para poderlos analizar debemos visualizarlos, es por ello por lo que para el proyecto de Mindfulness es necesario un componente de visualización, el cual se busca también sirva para otros tipos de proyectos.

Visión General del Sistema

El componente de visualización será accedido por medio de autenticación en el sistema, que luego de realizar este proceso podrá construir dashboard con graficas de línea, torta y barras de acuerdo con unos parámetros ya establecidos. La graficas se alimentarían del set de datos almacenados en el proyecto de Mindfulness.

Objetivos y Contexto

El principal objetivo del componente es que pueda ser parametrizable para la visualización de datos científicos en primera instancia para redes sociales, pero que se pueda acoplar a cualquier proyecto siempre y cuando cumpla con los parámetros que solicite el componente de visualización.

Para la mejor adaptación del componente a otros proyectos se debe cumplir:

Integración: El componente debe permitir interactuar con los demás componentes externos del proyecto macro bajo ciertos parámetros fijos.

Seguridad: El componente debe controlar la autenticación de usuarios con el fin de restringir su uso.

Disponibilidad: El componente debe estar activo al momento de que se envíen peticiones por parte de los demás componentes del proyecto macro.

Mantenimiento: El componente debe permitir fácil adaptación con lo demás componentes y permitir su actualización para mejorar s funcionalidad.

Vista técnica: El modelo planteado se define en paquetes principales siguiendo la arquitectura seleccionada MVC, con lo que se busca mantener una mejor administración del componente de visualización.

Vistas

Vistas Estructurales

Vista de Componentes

El diagrama de componentes muestra los elementos principales del componente de visualización de datos científicos donde se divide en 3 partes que son en la lógica de la gráfica, la lógica del dashboard y la lógica del usuario.

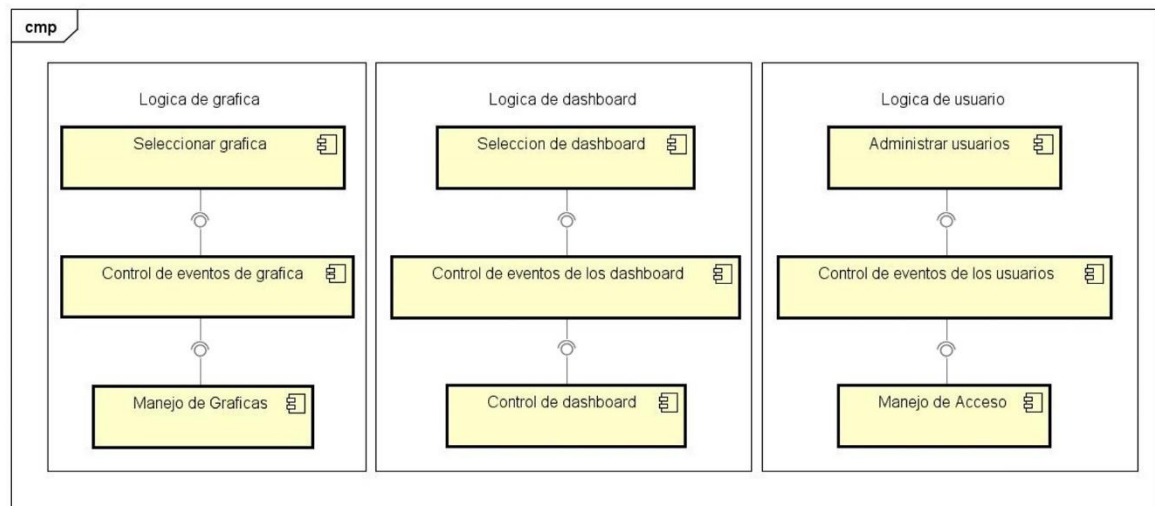


Imagen 15 Diagrama de componentes

Capa de Utilitarios

Esta capa muestra las diferentes librerías, lenguajes de programación y framework que permiten el desarrollo del componente de visualización.



Imagen 16 Diagrama capa de utilitarios

Capa de Acceso a Datos

La capa de acceso a datos contiene dos componentes, uno que es control de datos donde se controlara todas las consultas que se enviar a la base de datos y en la base de datos que almacenara la información del componente.

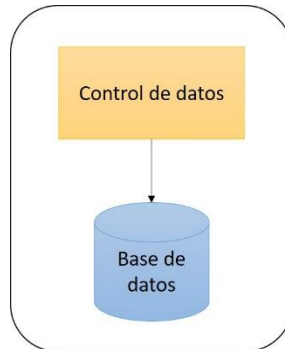


Imagen 17 Diagrama capa de acceso a datos

Capa de Negocio

En esta capa es donde se encuentra el mecanismo lógico del componente con las restricciones y su forma de procesar las solicitudes desde la capa de presentación y el cual consulta los datos desde la capa de acceso de datos.



Imagen 18 Diagrama capa de negocio

Capa de Presentación

En esta capa se presenta al usuario las diferentes opciones que podrá realizar en el componente y de acuerdo con ello, lo transmitirá a la capa de negocio.



Imagen 19 Diagrama capa de presentación

Vista de Despliegue

El diagrama de despliegue muestra por medio de los nodos, como es la ejecución de los componentes y como se distribuirán de manera correcta, pero en un principio, tanto el aplicativo como la base de datos se manejará de forma local.

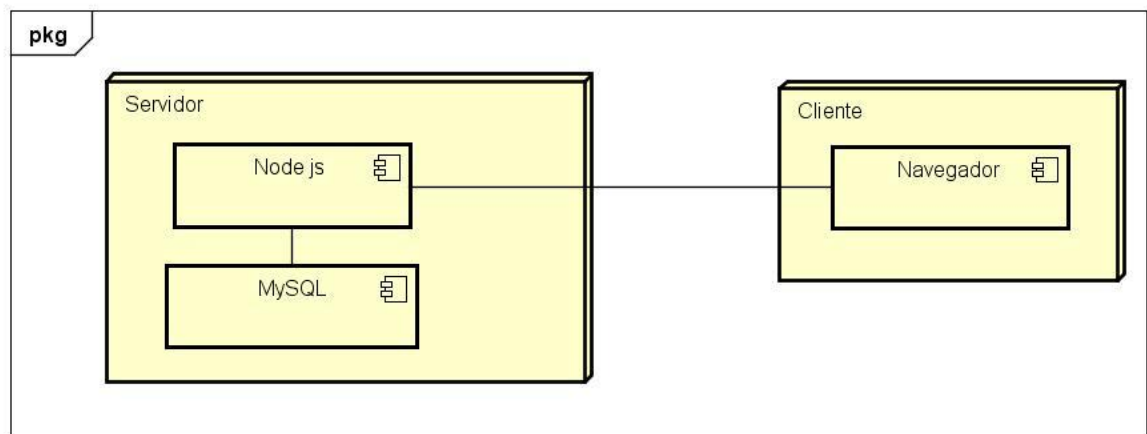


Imagen 20 Diagrama de despliegue

Referencias

- [1] IEEE Computer Society, "12207-2008 - Systems and software engineering", *IEEE STANDARD*, 2008. [En línea]. Disponible en: <https://standards.ieee.org/findstds/standard/12207-2008.html>.
- [2] Microsoft, "Información general sobre ASP.NET MVC", *Developer Network*,

2017. [En línea]. Disponible en: [https://msdn.microsoft.com/es-es/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/es-es/library/dd381412(v=vs.108).aspx).

- [3] IBM, "Patrón de diseño de modelo-vista-controlador", *IBM Knowledge Center*, 2017. [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SSZLC2_8.0.0/com.ibm.commerce.developer.doc/concepts/csdmvcdespat.htm.